Supports

OntoWeb
http://www.ontoweb.org/

ELSNET
http://www.elsnet.org/

Sponsors

Co-operating Organisations

The Workshop Programme

08:40 - 09:45 - Opening

08:45 - 09:10 - Melina Alexa, Bernd Kreissig, Martina Liepert, Klaus Reichenberger, Lothar Rostek, Karin Rautmann, Werner Scholze-Stubenrecht, Sabine Stoye. *The Duden Ontology: an Integrated Representation of Lexical and Ontological Information*

09:10 - 09:35 - Bernardo Magnini, Manuela Speranza. *Merging Global and Specialized Linguistic Ontologies*

09:35 - 10:00 - Dietmar Rösner, Manuela Kunze. *Exploiting Sublanguage and Domain Characteristics in a Bootstrapping Approach to Lexicon and Ontology Creation*

10:00 - 11:00 - Christiane Fellbaum. *Parallel Hierarchies in the Verb Lexicon*

11:00 - 11:20 - Coffee break

11:20 - 11:45 - Aldo Gangemi, Nicola Guarino, Alessandro Oltramari, Stefano Borgo. *Restructuring WordNet's Top-Level: The OntoClean based approach*

11:45 - 12:10 - Maarten Janssen. *EuroWordNet and Differentiae Specificae*

12:10 - 12:35 - James Pustejovsky, Anna Rumshisky, José Castaño. *Rerendering Semantic Ontologies: Automatic Extensions to UMLS through Corpus Analytics*

12:35 - 14:00 - Lunch

14:00 - 14:25 - Roberto Navigli, Paola Velardi. *Automatic Adaptation of WordNet to Domains*

14:25 - 14:50 - Wim Peters. *Self-enriching Properties of Wordnet: Relationships between Word Senses*

14:50 - 15:15 - Sandiway Fong. *On the Ontological Basis for Logical Metonomy: Telic Roles and WordNet*

15:15 - 15:40 - Anthony R. Davis, Leslie Barrett. *Relations among Roles*

15:40 - 16:40 - Yorick Wilks. *To be announced*

16:40 - 17:00 - Coffee break

17:00 - 18:00 - Discussion: *Distinctions between Lexical and Ontological Knowledge*

Workshop Organisers

Kiril Simov
Linguistic Modelling Laboratory, CLPP, Bulgarian Academy of Sciences, Bulgaria
and
OntoText Lab. Sirma AI Ltd, Bulgaria
E-mail: kivs@bgcict.acad.bg

Nicola Guarino
National Research Council, LADSEB-CNR, Italy
Email: Nicola.Guarino@ladseb.pd.cnr.it

Wim Peters
NLP group, Department of Computer Science, University of Sheffield, England
Email: W.Peters@dcs.shef.ac.uk

Workshop Programme Committee

Table of Contents

Author Index

The Duden Ontology:

An Integrated Representation of Lexical and Ontological Information

Melina Alexa[1], Bernd Kreissig[1], Martina Liepert[1*], Klaus Reichenberger[2], Lothar Rostek[3], Karin Rautmann[1], Werner Scholze-Stubenrecht[1], Sabine Stoye[2]

[1] Bibliographisches Institut und F.A. Brockhaus AG (BIFAB), Mannheim, Germany
{Melina.Alexa, Bernd.Kreissig, Martina.Liepert, Karin.Rautmann, Werner.Scholze-Stubenrecht}@bifab.de
[2] intelligent views, Darmstadt, Germany
{k.reichenberger, s.stoye}@i-views.de
[3] FhG-IPSI, Darmstadt, Germany
rostek@ipsi.fhg.de

Abstract

We report on a data model developed for the representation of lexical knowledge for the Duden ontology. The model is the result of a cooperation between the publishing house Duden and the software company *intelligent views*. Our general aim is to create an asset pool in which all the information present in the Duden dictionaries is integrated in order to support reusability for different print and electronic products, provide solutions for language technology applications as well as support the efficient maintenance of the Duden dictionary data.

1. Introduction

In this paper we describe the data model developed for the representation of lexical knowledge for the Duden ontology. Duden is a well-known publisher of language reference products in both print and electronic form as well as products for language technology for the German language. It belongs to the publishing house Bibliographisches Institut und F.A. Brockhaus AG (BIFAB). The model described here is the result of a cooperation project between Duden and the software company *intelligent views*, which is a spin-off company of the Fraunhofer Integrated Publication and Information Systems Institute (IPSI).

Our general aim is to create a rich computational resource in which all the information present in the Duden dictionaries is integrated in order to support

- the reusability for both print and electronic products,
- the development of language technology applications as well as
- the efficient maintenance of the Duden dictionary data, for example the ten volume Duden dictionary (Duden, 1999) or the Duden spelling dictionary (Duden, 2000).

Two further considerations have been important in developing this model:

- it should be flexible enough to adjust to new emerging requirements with regards to both the dictionary structure itself as well as the production of different titles and different types of dictionaries, and
- it should at a later stage allow the representation of encyclopedic information.

Note that a significant requirement has been that the Duden print dictionaries can be produced from the constructed computational resource at least as efficiently as is currently the case.

Furthermore, an important prerequisite has influenced the modeling of the data a great deal: the computational resource to be created should not only be useful for the production of print and electronic (both on- and off-line) dictionaries. It should also be useful for solving problems such as lexical and semantic ambiguity and reference resolution for knowledge intensive and real natural language applications such as, for example, a question answering system for German, for which broad-coverage of the morphological, grammatical and semantic information of the language is necessary.

1.1. Motivation

Although the majority of the Duden dictionary data are in SGML format, the markup of each dictionary is strongly print oriented rather than content oriented. For each of the SGML-based dictionaries there is a Document Type Definition (DTD) according to which the lexicographers maintain their data. Corrections or other modifications of existing lemmas and their properties as well as addition of new lemmas take place separately for each Duden title. This means that if, for example, a lexicographer modifies a lemma for the Duden dictionary *Duden – Fremdwörterbuch* (Duden, 2001a), the reference volume for the correct spelling of foreign words in German, each entry for the modified lemma in other Duden dictionaries, e.g. the Duden spelling dicitionary (2000) or Duden (2001b), has to be modified or updated manually. This is not only inefficient with regard to time but it is also prone to errors and inconsistencies. In contrast, the formal explicit representation of the Duden dictionary entries in a single knowledge base supports the administration and maintenance of dictionary data in an efficient, consistent and systematic manner.

[*] Since April 2002 at SFS, Universität Tübingen, Germany.

A further aspect concerns the additional possibilities offered by an explicit representation of all information relevant to each dictionary entry of the Duden data: depending on the quality of the data model it will be possible to generate different 'sub-lexicons' from a single data pool. These are, in principle, nothing more than different 'views' of the knowledge stored in the data pool. Examples of such sub-lexicons may be a list of all compounds in the Duden dictionaries, or a differentiated system of lexemes with their morphological (e.g. part of speech, gender), grammatical (e.g. subcategorization) and semantic (e.g. synonyms) information.

1.2. Related work

The work described in this paper relates to research on knowledge representation for lexical and semantic as well as for ontological information for the purposes of dictionary production and for natural language applications. It has to be emphasized, though, that it is our particular needs as publisher, our abilities and the tools supporting our work which guide the reported work in the first instance and not theoretical considerations. For this reason our main focus is not to construct *the* most expressive model for the representation of lexical and semantic representation but rather the construction of a large scale resource to be used for the efficient production of our dictionaries and for NLP applications.

Unlike Wordnet (Fellbaum, 1998), EuroWordNet (Vossen, 1998) and GermaNet (Hamp & Heldweg, 1997; Kunze, 2000) the Duden Ontology integrates extensive morphosyntactic properties of denotations with ontological information about their senses (see section 2). With regard to morphosyntactic information, this is represented in an extensive manner in the Duden Ontology, whereas WordNet and WordNet-like systems use elementary part-of-speech information and sub-categorization frames.

In contrast to the project WiW - Wissen über Wörter (Müller-Landmann, 2000; 2001), instead of a relational model we have opted for an object-oriented approach, which is advantageous for factorizing common information and supports inheritance of relations and attributes. A further point which distinguishes our work from the WiW-project is that we make use of the existing dictionary assets of Duden and therefore do not start from scratch. This allows us to build a comprehensive resource within a relatively short time and even more importantly to evaluate the expressiveness and suitability of the implemented model for our needs.

There are similarities between our approach and that of the Mikrokosmos project (Mahesh & Nirenburg, 1995): We too make a clear distinction between the representation of language-specific and language-neutral information. In our terminology language specific information is represented by *term* objects, whereas *concept* objects are used for representing language-neutral information (see section 2.1). One of the differences between the two projects is that the Duden Ontology integrates both kinds of information within a single resource, whereas the Mikrokosmos project uses two apparently separate databases, one for the lexicon and one for the ontology, for storing denotations and denotation-neutral concepts.

There are parallels of our work with the TransLexis conceptual schema (Bläser, 1995) with the distinction between lemma, homograph and sense. TransLexis is based on a relational model and has been driven by requirements for multilingual terminology management.

Currently, the Duden Ontology does not include an automatic classifier for classifying defined concepts on the basis of formal concept definitions, as for example the GALEN ontology and its related technology does (Rogers et al., 2001, Rector et al. 1998). With the exception of simple inference mechanisms, such as inheritance or relation path definition, the Duden Ontology does not feature a full-fledged inference engine.

2. Data model for the Duden Ontology

The Duden data model is based on a concept-oriented representation which offers the possibility of defining semantic relations between the concepts. In addition, it provides the hook for an integration of encyclopaedic data as well as for the representation of factual knowledge at a later phase.

To this end, the vocabulary of the Duden volumes is classified in a rigid manner according to a generic hierarchy relation. This is similar to WordNet where the synsets play the role of the concepts. In order to provide the hook for representing facts an explicit distinction between individuals and concepts (word senses) is necessary, which results in the creation of an ontology. According to our definition there are two essential features of an ontology:

- a classification of concepts according to a rigid generic hierarchy relation (SUBCONCEPT_OF relation) and
- the distinction between individuals and concepts, whereby an individual is related to a concept by means of an INSTANCE_OF relation.

Individuals in our data model are representations of concrete persons, geographical places, organizations, institutions, events etc. For example, 'Immanuel Kant', 'EU', 'Gran Canaria', 'Olympic Games 2004 in Athens' are all denotations of individuals.

2.1. Lemma-Term-Concept: roles of words in the language game

An ontology offers a formal method to structure sets of individuals with a set of individuals being an extension of a concept. Concepts are related to other concepts by means of a rigid hierarchy relation. This supports the factorizing of common information (see section 2.2.1) to more abstract levels.

Our idea is to represent the words of a language formally as *individuals*, called *lemmas* within our model. We consider morphosyntactic and word usage classes, e.g. information about the part-of-speech class of a word, its subcategorization frame, pragmatic usage, etc., formally as *concepts* and use them to group and classify the lemmas. This results in a further ontology, a kind of 'morphosyntactic ontology' about the 'world of words', which may be considered as a kind of further dimension of the first ontology described above, representing word senses and real world objects.

We bridge the two ontologies by using a denotation relation for connecting lemmas to one or more senses.

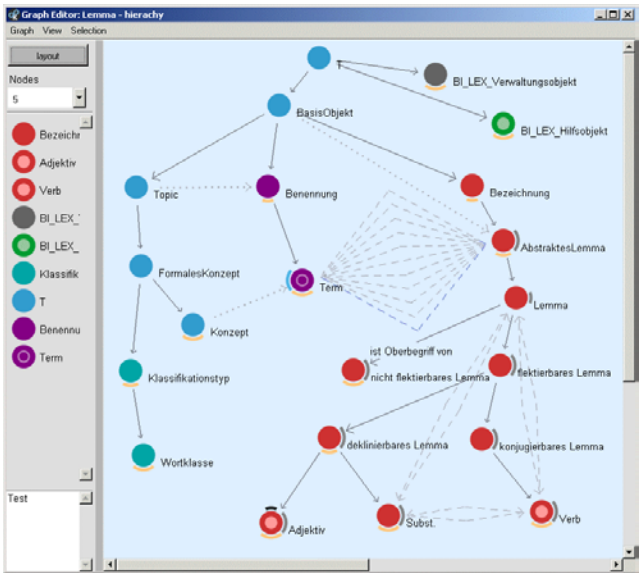


Figure 1: Top level of the Duden Ontology

Each sense of a lemma can be considered a role that this lemma plays in the language game, whereby each role played is represented by a single object, which we call *term*. In general, a lemma has more than one sense and thus a single lemma has more than one term assigned to it. Each sense of a lemma is represented by a single concept object.

On the other hand, a concept can be related to more than one term and thus to more than one lemma. This establishes the synonymy relation: Two lemmas are synonyms, if one of their corresponding terms points to the same concept.

We illustrate this in Figure 1: the top level concepts of the Duden Ontology are shown with the concept "Topic" being the root of the first ontology and "Bezeichnung" (denotation) being the root for the morphosyntactic ontology. The gap between the lemmas and concepts is bridged by means of a specific object class, i.e. "Term". All common information to all three object types is factorized at the "BasisObjekt".

2.2. Granularity gains

2.2.1. Factorizing of common information

One of our goals is to support the lexicographer in avoiding redundancy as this is one of the most important means for efficient maintenance, data consistency and multiple usage of the data. The means to avoid redundancy is the factorizing of common information: all information common to all objects should be stored in some more general object; when more general information is needed by the more specific object, this can be inherited (during runtime) from the more abstract one. Note that redundancy free storage does not hinder a redundant presentation of the data. The latter is not only useful for the lexicographer, but it is also advantageous for electronic products for which space restriction is not as rigid as in print products.

Obviously, it is not always possible to achieve a completely redundancy free data representation. Redundancy may, however, increase error-proneness in lexicography work. It has to be noted though that if redundant storage is required as a means for improving system performance, redundancy should be maintained by the system itself and be completely hidden from the user. The question of redundant storage is therefore "simply" a matter of the concrete implementation and not relevant to the model.

In our data model, the lemma is where the word-related information common to all its terms is factorized. A concept factorizes the meaning-related information common to all its synonymous lemmas. A term, though, may overwrite factorized information inherited by its corresponding lemma. In this way, we represent grammar and usage exceptions of particular lemmas, e.g. that a lemma in a particular sense may have no plural form.

2.2.2. Fine-grained relations

By representing terms as separate objects we gain granularity for the relations. In particular, we can link usage examples and citations for the dictionary entries to terms and not just to lemmas. By doing this, we disambiguate the meaning of the lemma in the usage example. Since we import data from the Duden dictionaries, the usage examples are already assigned to the particular meanings of a dictionary entry (for details see section 4.2.). With such information formally represented, one may get all usage examples of a concept simply by the union of all usage examples of all its terms.

In a similar manner, the representation of the decomposition of compound nouns on a term level and not only on a lemma level brings gains in granularity. This is advantageous when using such a resource for parsing or information retrieval tasks as the components of compounds are already disambiguated.

2.3. Concrete example

We explain the above model by means of an example from the Duden dictionary. The word "Bar" has three separate entries in the ten volume Duden dictionary (Duden, 1999):

[1]Bar, die; -, -s [engl. bar, urspr. = Schranke, die Gastraum u. Schankraum trennt < afrz. barre, Barre]: **1. a)** *intimes [Nacht]lokal, für das der erhöhte Schanktisch mit den dazugehörigen hohen Hockern charakteristisch ist:* eine B. besuchen, aufsuchen; in einer B. sitzen; **b)** *barähnliche Räumlichkeit in einem Hotel o. Ä.* **2.** *hoher Schanktisch mit Barhockern:* an der B. sitzen; Monsieur de Carrière lud mich ein, mich zu ihnen an die B. zu setzen (Ziegler, Labyrinth 258).

[2]Bar, das; -s, -s <aber: 3 Bar> [zu griech. báros = Schwere, Gewicht]: *Maßeinheit des [Luft]drucks;* Zeichen: bar (in der Met. nur: b).

[3]Bar, der; -[e]s, -e [H. u.]: *regelmäßig gebautes, mehrstrophiges Lied des Meistergesangs.*

There are three lemmas for "Bar" in the sense of (1) pub or bar, (2) measurement unit for (air) pressure and (3) a special form of song. The first entry, **[1]Bar,** has three senses (pub, hotel bar and counter) whereas **[2]Bar** and **[3]Bar** each have only one sense. Although all three lemmas are nouns, each lemma belongs to a different gender and declination class shown in the entry with the article and the genitive and plural form suffixes, e.g. "**[1]Bar**, die; -, -s" is feminine and forms the plural with a final 's'.

For each of the five senses there exists a separate term and a corresponding (separate) concept. Each sense definition, e.g. "*intimes [Nacht]lokal, ...*" for 1(a), is stored at the concept level. The usage examples and citations, e.g. „an der B. sitzen" (*English translation*: sitting at the bar) and „Monsieur de Carrière lud mich ein, mich zu ihnen an die B. zu setzen (Ziegler, Labyrinth 258)." (*English translation*: Monsieur de Carrière invited me, to join them at the bar (Ziegler, Labyrinth 258)), are connected to the term **[1]Bar (2)**.

Only the lemma, **[2]Bar** is synonymous to the lemma "**[2]bar**" as well as to the meteorological use of the sign "b". If we wish to extract all usage examples for say the concept "night bar" only those examples of the lemma "Bar" belonging to the term **[1]Bar (1a)** will be extracted. All other usage examples belong to terms, whose concepts are either hyponyms of the concept "night bar" or the concept "night bar" itself.

3. Tools and implementation

3.1. Ontology as a knowledge network

The data model is implemented with the *intelligent views* software system K-Infinity, which offers broad support for object-oriented knowledge modeling as well as for the creation, maintenance and use of a knowledge network. The software distinguishes between concepts and individuals and allows for the definition of relations and attributes both of which are inherited via the concept hierarchies.

The way we define ontology in our model fits well with the definition of a knowledge network in K-Infinity. The cornerstone of a knowledge network is a collection of concepts that structure information and allow the user to view it. The concepts are organized into hierarchies where each concept is related to its super- and subconcepts. This forms the basis for inheriting defined attributes and relations from more general to more specific concepts.

Concepts, individuals, attributes and relations are central to the construction of the knowledge network. A means for handling multiple inheritance are the so-called extensions or roles, the terms, which we use to represent the different senses of a lemma.

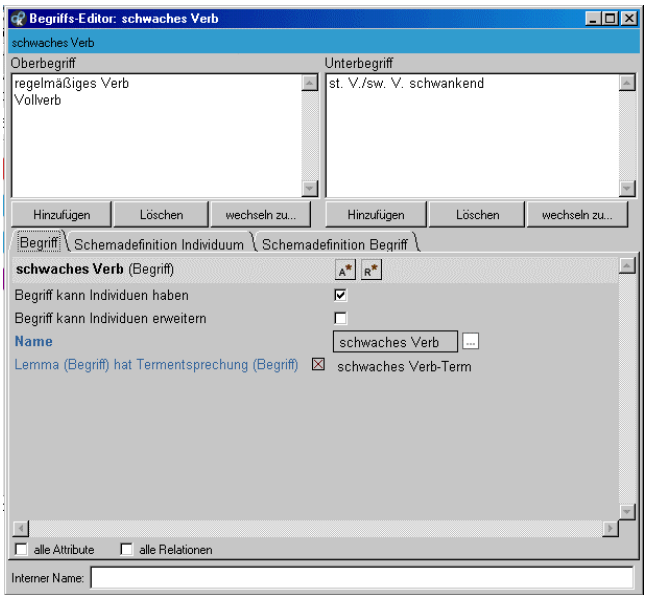


Figure 2: Concept Editor

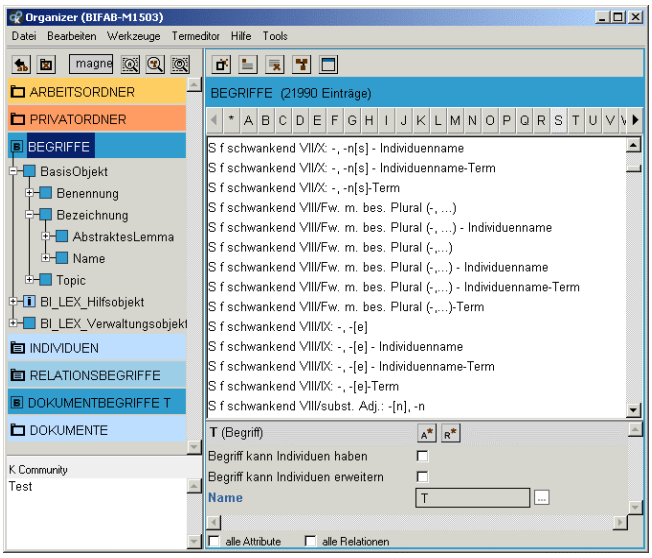


Figure 3: K-Organizer

3.2. K-Infinity Tools

The Knowledge Builder is K-Infinity's main component. It allows knowledge engineers and lexicographers to create, delete, rename and edit both objects and relations, as well as to relate objects to each other according to defined relations. This can be done in two different workspaces:

- The Graph Editor (shown in Figure 1) provides a graphical view of the network of objects and the relations between them. The network may be expanded according to the defined model. The Graph Editor supports the monitoring of the data by means of implemented consistency rules. One of the Editor's basic functions is an interactive network layout algorithm for the exploration of the knowledge network.
- The Concept Editor (see Figure 2) allows the user to focus on one object and its semantic links to neighboring objects. It is a supplement to the Graph Editor in that it allows the user to survey links and their attributes in detail, and to modify them if necessary.

Along with the tools for editing the knowledge network, there is the K-Organizer which supports administration, navigation, search and query formulation. The K-Organizer (Fig. 3) can be used to classify and group objects, either manually or by using existing object properties: for example, to organize all objects created before a certain date or all superconcepts with more than 10 subconcepts into a single folder.

Given the work context of the particular project, namely dictionary maintenance, an additional tool has been developed as a special extension for viewing and editing network objects from the perspective of a dictionary entry, called Term Editor. The Term Editor displays a lemma together with its associated terms and concepts in a single window in a comprehensive and compact way.

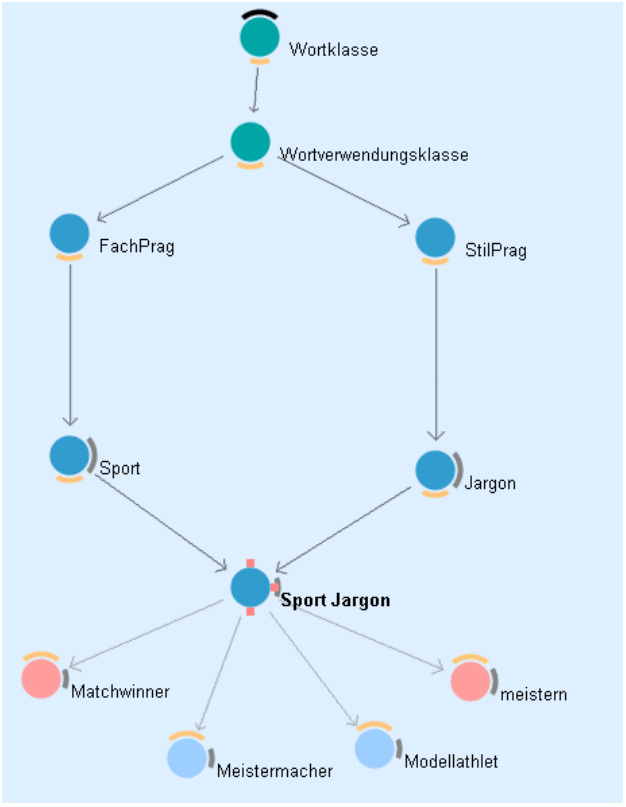


Figure 4: Example of pragmatic classes

3.3. Defined classes

There is a set of ca. 290 defined grammar classes, e.g. "noun which has a plural form", "masculine noun with declination type X", etc., ordered in a polyhierarchy. From these there are 160 classes which are assigned to lemmas; all the other classes are used to complement the poly-hierarchy as a means for flexible navigation and access.

Moreover, there are ca 1000 pragmatic classes, which are also ordered in a polyhierarchy, of which ca 250 are "basic pragmatic classes". The rest are combinations of pragmatic classes, such as for example, the class "Sport Jargon" shown in Fig. 4, which is a subclass of both "sport" and "jargon" classes. The class "jargon" is a subclass of "style" (StilPrag in Fig 4) whereas the super-class of "sport" is the pragmatic class "domain" (FachPrag). All in all there are at the moment over 200 relations defined in the model.

The defined grammar classes represent various aspects of the morphosyntactic nature of words. Starting from the general distinction of non-inflected and inflected word classes we divide the latter into conjugatable and declinable classes such as pronoun, article, adjective and noun and proceed to organize them extensively, which is necessary due to the rich morphology of German.

The noun hierarchy, shown in Figure 5, includes some abstract classes such as "noun by gender", "noun by type of declension", "noun with plural", "noun without plural", "noun derived from adjective", to classify the concrete noun classes such as the noun class the word "Aubergine" belongs to, namely, "feminine noun, declension type IX".

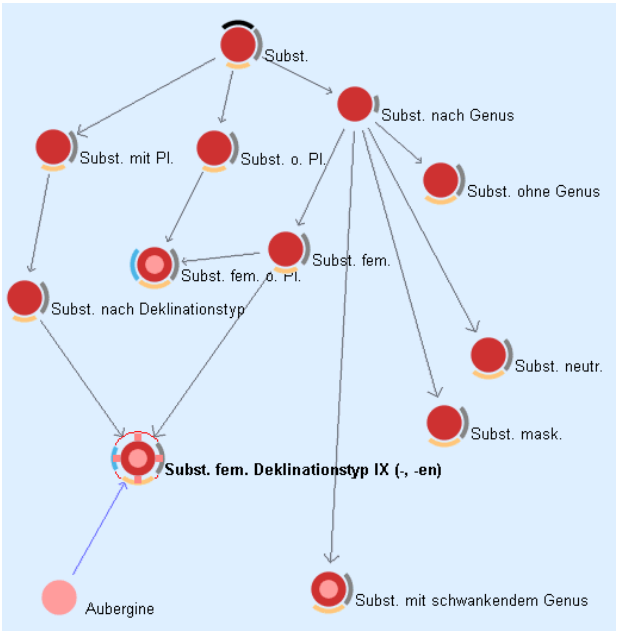


Figure 5: Example of noun classes

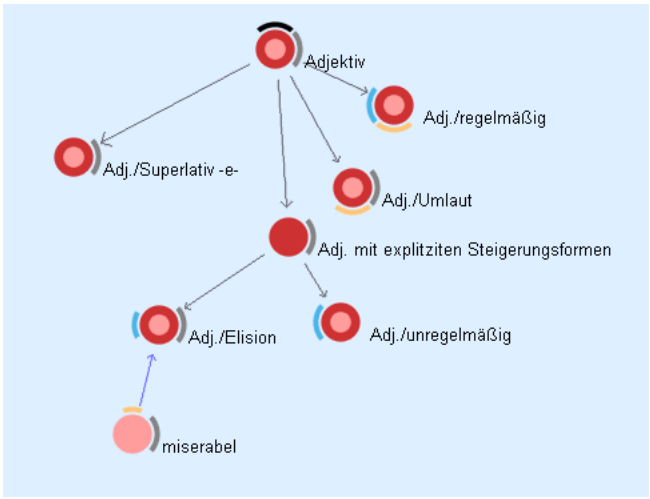


Figure 6: Example of adjective classes

As an additional example of the polyhierarchies consider the structure of the adjective classes (see Figure 6): In addition to the regular adjectives, we have defined subclasses for those with an explicit comparative form, with Umlaut and for those forming the superlative with "-e-". In the figure, the lemma "miserabel" is shown classified as an adjective belonging to the adjective subclass with an irregular comparative form, because of the elision of its -e-.

4. Import

To populate the Duden Ontology we first imported the data from the ten volume Duden dictionary (Duden, 1999), which contains ca. 200,000 lemmas, followed by the import of the entries of the Duden spelling dictionary (2000) with over 110,000 lemmas. Although there is a significant amount of overlap between the two dicitionaries, the former contains not only far more definitions than the latter, but also more grammatical,

etymological and pragmatic information. Importing and merging of further volumes are planned for the future.

The result of the complete import of the above data is a huge object network representing the information of over 200,000 entries from different dictionaries, whereby the entries themselves are decomposed into interlinked objects.

4.1. SGML dictionary data

As already mentioned, for each Duden dictionary, e.g. Duden (1999) or Duden (2000), there exists an SGML DTD. The basic structure of the dictionary articles is similar, however: Each dictionary article has a start and an end tag and each article element is divided into two parts, the head and the body. The head contains mainly information relevant to the lemma object of our data model and the body contains more detailed information concerning the senses of a lemma. The elements for phonetic, grammatical, etymological and pragmatic information are included in the head element. The body contains the substructure of the article and within this part there are elements containing definitions, examples, explanations, proverbs, idioms and idiomatic phrases. This straightforward structure is often interrupted by so called "meta-tags" which may appear anywhere within the above elements and contain some kind of text fragments. Naturally, this adds to the complexity of the import task.

There is, of course, no explicit tagging for terms and concepts, which is why a mapping from the existing mark up to the object types of our model is necessary. Because of the differences between the DTD(s) and our model it is not possible to write a simple context-free look-up table for mapping the DTD tags into the modeled object types. The content model of some elements is an iteration of a sequence of elements with optional parts, as shown in the example below for the element `defphr` (definition phrases):

```
<!ELEMENT defphr  - -
((ph?,gr?,prag?,(def|erk),erg?)?,bsp?,uew?,
rw?,spw?,iw?,(kurzf+  |  kurzw  |  abk+  |
zeich+)?)+ >
```

We map each iteration to a term, but since there is no explicit tag around this sequence of elements, the parsing process needs to exploit the contexts of the sequence in order to assign the information to the appropriate term.

4.2. Mapping

4.2.1. Creation of lemmas

Each dictionary entry is mapped to a lemma object. Typically, the homograph entries are indicated in the printed dictionary by a superscripted digit, which is also explicitly marked up as an attribute value in the article element. In this case we create different lemma objects with the same name, but with a different homograph-ID. The orthographic variants, e.g. "Photo" and "Foto", are marked up explicitly in the data. Separate lemma objects, which are related to the main lemma, are created for such variants.

Idioms and proverbs form specific lemma types which are automatically created during import.

4.2.2. Creation of terms and concepts

The different senses of an entry are structured in the dictionary by numbers or letters. We map each sense to a term and for each definition element we create an additional concept object. The usage and citation examples are assigned to the term object.

Grammatical or pragmatic information, which typically holds for the lemma, is modified in the sense description. Such modifications are stored in the corresponding term and overwrite the grammatical or pragmatic information inherited by the lemma.

The examples and definition phrases of the dictionary entries are often condensed for space reasons, e.g. the lemma appears in an abbreviated form. For instance, the entry for "Bar" in section 2.3 contains the phrase "an der B. sitzen" the complete form of which is "an der Bar sitzen". We expand such abbreviated forms during import and store the full form. Moreover – if necessary – we can generate the condensed form for export purposes.

4.2.3. Cross-references

During import we take care that no information necessary for the export of the data for the production of the dictionaries, such as the cross-references, is lost. The dictionary data contain explicit SGML elements for cross-referencing. We use the attribute values for the target article number and the subsection (the sense) in order to link the source and the target at the term level. We further check whether the subsection for the target lemma exists and whether the content of the cross-reference element can match the target lemma. In this way, we introduce an additional control for checking the correctness of cross-references, which is obviously advantageous for the quality of the constructed pool.

Due to the fact that the SGML data were originally created by an automatic conversion several thousands of the 80,000 cross-references solely refer to a subsection and have no reference to the article-ID. To resolve the missing cross-references we lemmatise the content of the cross reference elements and generate a list of target candidates, which is proofread by the lexicographers.

4.3. Enriching

Our aim is to populate the network with semantic relations, such as synonymy, hyperonymy, PART_OF or INSTANCE_OF relations. The SGML data contain no explicit mark up for such relations and a fully automated acquisition of semantic relations is not possible. We thus depend on maximal exploitation of our dictionary data in order to acquire semi-automatically semantic knowledge of this kind. For instance, the structure of the definition texts – which are stored at the concept level – is sometimes indicative for a synonymy relation holding between a given dictionary entry and its definition. As an example consider the dictionary entry "Yellow Press" in Duden (1999):

Yel||low Press ['jɛloʊ 'prɛs], die; - - (auch:) **Yel||low|press,** die; - [engl. yellow press, eigtl. = gelbe Presse] (Jargon): *Regenbogenpresse:* Längst ist die Witwe, von deren Auftritten einst die Y. P. profitierte, ruhiger geworden (FR 2. 1. 99, 9).

The word "Regenbogenpresse" (literary translation: "rainbow press") is marked up as definition text of the

term “Yellow Press”. We establish a synonymy relation between the two terms “Regenbogenpresse” and “Yellow Press” and their corresponding lemmas by assigning the same concept object to both terms.

We further plan to exploit the definition texts in combination with the cross-references to acquire hyperonymy and INSTANCE_OF relations.

A further method for extraction of hyperonyms is to automatically analyse compound words with the aim of extracting the heads of the compounds as these are in most cases the hyperonyms of the compounds.[1] For example, by analysing the compound “Volkstanz” (folk dance) we can infer that it is a hyponym of the word “Tanz” (dance).

For the representation of the morphological decomposition we define two relations and an attribute: *hat_Bestimmungswort* (has_modifier), *hat_Grundwort* (has_head) and the attribute *hat_Fuge* (has_join_morpheme). These relations are defined for both terms and lemmas. This is necessary since we cannot acquire all information we need in a single step. Rather we proceed iteratively to achieve a decomposition at the term level. In a first step all compound words of the dictionary are automatically morphologically analysed with the morphological analysis tool MPRO (Maas, 1996) to generate their components. As the decomposition of compounds is not always unambiguous, we disambiguate the analysis output by rejecting those compound analyses which have at least one component which is not a dictionary lemma. To illustrate this, there are two possible decompositions of the word “Medizinaldirektorin” (medical director) when automatically analysed:

medizinal – direktorin (medical – director)
medizin – aldi – rektorin (medicine – Aldi – rector)

The second analysis is nonsensical: Aldi is the name of a well-known German supermarket chain. The second analysis is thus rejected on the basis that there is no lemma for the the name Aldi. This strategy, however, does not always work, for example, consider the automatic analysis of the word “Marineuniform”:

marine – uniform (navy – uniform)
marine – uni – form (navy – university – form)

Again, the second decomposition is nonsensical, but in this case all three components are proper dictionary lemmas. The rule for selecting the correct decomposition is here a different one: the candidates for the right decomposition are those with the minimal number of components.

This way we fill in the lemma relations for the components of compounds[2]. If the lemmas which are components of a compound have only one sense, we have also achieved a decomposition at the term level. This is only possible, however, for a small number of compounds. Further investigation is required to determine a method to support the decomposition of compounds at the term level.

5. Conclusions and future work

In constructing the Duden Ontology our aim is not to build a general ontology of the world, but rather to create a computational resource which both supports efficient dictionary production and aids real world NLP applications. The creation of the Duden Ontology has been driven by our products and needs as well as by the abilities within the context of our work and the tools chosen.

This approach is guided by practical needs and has practical advantages for the lexicography work: by means of such an approach it is possible to maintain the dictionary data in a homogenous manner within a single data pool, something which was not previously possible for the Duden data.

With regard to the data model presented here, we believe that this kind of integrated model of semantic and grammatical information helps to avoid redundancy in storage and to maintain data without losing the ability to filter different sets of data and to generate various views of them with different granularity. The implementation of the data model is such that it allows modifications and further extensions, such as for example the definition of further semantic relations.

The next steps of our work concern the enrichment of the ontology with subcategorization information as well as with further semantic information. In particular, we plan to exploit the definition texts in combination with the cross-references to acquire hyperonymy and INSTANCE_OF relations.

For the future we plan to model further semantic relations to embed factual knowledge and encyclopedic information.

6. Acknowledgements

Special thanks go to Annette Klosa, Elke Siemon and Jan Schümmer for their valuable contribution to the project.

The work reported in this paper has been in part supported by the BMBF (German Ministry for Education and Research) grant 08C5885 for the research project „Lexikonbasierte Wissenserschließung: Natürlich-sprachige Suche und 3D-Wissensnavigation“.

7. References

Bläser. B. (1995). TransLexis: An Integrated Environment for Lexicon and Terminology Management. In P. Steffens (Ed.): *Machine Translation and the Lexicon, Third Internationals EAMT Workshop, Heidelberg, Germany, April 26-28, 1993, Proceedings*. Heidelberg: Springer Verlag, pp. 159-173.

[1] Note that ca 50% of the dictionary entries are compounds, which is attributable to the productivity of compounding in German.

[2] It is interesting to add that compound analysis at the lemma level is also important to determine the grammatical class for the compound word. Due to space reasons the single grammatical information coded for compound words in e.g. the ten volume Duden dictionary (1999) is gender. Whereas this is not problematic for a user of a dictionary, for automatic processing the missing information about the grammatical class of the compound is necessary. The grammatical class of the compound is determined by the class of the compound head.

Duden (1999). Duden - Das Große Wörterbuch der deutschen Sprache in 10 Bände. Mannheim: Dudenverlag, 3rd Edition.

Duden (2000). Band 1 – Die deutsche Rechtschreibung. Mannheim: Dudenverlag, 22nd Edition.

Duden (2001a). Duden Band 5 – Das Fremdwörterbuch. Mannheim: Dudenverlag, 7th Edition.

Duden (2001b). Duden Band 10 – Das Bedeutungswörterbuch. Mannheim: Dudenverlag, 2nd Edition.

Fellbaum, Ch. (Ed.) (1998). *Wordnet: An Electronic Lexical Database.* , Cambridge, MA: The MIT Press.

Hamp, B. & H. Feldweg (1997). GermaNet a Lexical-Semantic Net for German. In *Proceedings of the ACL/EACL-97 Workshop on Automatic Information Extraction and Building Lexical Semantic Resources for NLP applications*, Madrid, pp. 9-15.

Kunze, C. (2000). Extension and Use of GermaNet, a Lexical-Semantic Database. In *Proceedings of LREC 2000 Workshop on Lexicon: Semantic and Multilingual Issues* , Athens.

Mahesh, K. & S. Nirenburg, 1995. A situated ontology for practical NLP. *Proceedings of IJCAI '95 Workshop on Basic Ontologies Issues in Knowledge Sharing*. Montreal.

Maas, H-D. (1996). MPRO - Ein System zur Analyse und Synthese deutscher Wörter. In Roland Hausser (Ed.): *Linguistische Verifikation, Dokumentation zur ersten Morpholympics.* Tübingen: Max Niemeyer Verlag.

Müller-Landmann, S. 2001. Wissen über Wörter. Die Mikrostruktur als DTD. Ein Beispiel. In H. Lobin (Ed.), *Proceedings der GLDV-Frühjahrstagung*, Universität Gießen, 2001, pp. 31-40.

Müller-Landmann, S. 2000. Design eines Internet-Lexikons zwischen Recherche und Rezeption. In U. Heid, S. Evert, E. Lehmann & C. Rohrer (Eds.): *Proceedings of the Ninth EURALEX International Congress,* Universität Stuttgart, Vol. I, pp. 97-105.

Rector, A.L., P.E. Zanstra, W.D. Solomon, J.E. Rogers, R. Baud, W. Ceusters, A.M.W. Claassen, J. Kirby, J. Rodrigues, A. Rossi Mori, E.J. Van der Haring & J. Wagner (1998). Reconciling user' needs and formal requirements: issues in developing a reusable ontology for medicine. *IEEE Transactions on Information Technology in Biomedicine, 2* (4), pp. 229-241.

Rogers, J., A. Roberts, D. Solomon, E. van der Haring, Ch. Wroe, P. Zanstra & A. Rector (2001). GALEN ten years on: Tasks and supporting tools. In Patel, V., R. Roger & R. Haux (Eds.) (2001): *MEDINFO 2001. Proceedings of the 10th World Congress on Medical Informatics*. Amsterdam: IOS, pp. 256-260.

Vossen, P. (Ed.) (1998). *EuroWordNet: A Multilingual Database with Lexical Semantic Networks* (reprinted from Computers and the Humanities, 32:2-3, 1998). Dordrecht: Kluwer Academic Publishers.

Relations among Roles

Anthony R. Davis*, Leslie Barrett†

* StreamSage, Inc.
1818 N St. NW, Washington, DC 20036, USA
davis@streamsage.com

†TransClick, Inc.
250 W. 57 St., New York, NY 10019, USA
leslie@transclick.com

Abstract

In this paper we discuss the possible types of relationships between participant roles in related situation types. We first discuss principles that might determine which roles are present in one type of situation, given the roles present in a related type of situation. While no simple general rules seem to exist, there are useful rules for particular cases. In addition, we discuss how relationships between roles themselves parallel relations between other elements in ontologies. Apart from the subrole relation, we consider relations analogous to meronymy and antonymy, which are rare in the domain of roles, and a complementarity relation between roles, which is fairly common.

1. Introduction

How are participant roles in one type of situation related to the roles in another? What implications do the relations between roles in different situation types have for the relations between other elements of ontologies? We will focus here on two topics that bear on these issues. First, we discuss which principles might determine, given the set of roles appropriate for one situation type, which subset of those roles are appropriate for a second, related situation type. Second, we examine the extent to which relationships between roles parallel those we find between other kinds of elements in ontologies.

2. Goals and assumptions

The way we have described the two topics above presupposes certain characteristics of an ontology (or related resource such as WordNet). We first briefly present these assumptions here, and pose the more detailed questions that we will address in this paper.

2.1. Participant roles, situation types, and hierarchies

We view participant roles for present purposes as relations between an entity and a situation.[1] Thus we will often refer to the entity as a participant in the situation.[2]. We also assume that for each role there can be type restrictions on the kinds of entities and situations that are appropriate arguments for that role. A *perceiver* role, for instance, is restricted to sentient entities in perception situations. This in turn rests on the assumption that situations and entities can be grouped into types, a strategy that has proven fruitful as a central organizing principle in many ontologies (e.g., the Cyc ontology, the SENSUS ontology, Mikrokosmos, the ontology developed in Sowa (2000), and numerous more specialized ontologies). Here we assume that types of entities and situations are hierarchically arranged, with multiple inheritance permitted (multiple inheritance is pervasive in the Cyc ontology but rare in WordNet).[3]

2.2. Subroles

In part because roles have type restrictions on the entities and situations that can serve as their arguments, it is reasonable to talk of subroles. One advantage of structuring roles in this way is that we can provide for arbitrarily specific roles for situation types anywhere in the situation-type hierarchy, while maintaining very general roles, which prove useful, for example, in stating the linguistic regularities in linking from semantic roles to syntactic arguments of predicators. The type restrictions on a subrole's arguments must be at least as restrictive as those on its super-roles. In addition, it is natural to assume that a participant playing a role in a situation also plays all of its super-roles:

(1) $R \subset R'$ implies $\forall x,e$: $R(x,e) \rightarrow R'(x,e)$

This entails a homomorphism under subsumption from the hierarchy of situation types to the hierarchy of roles, and from the hierarchy of entity types to the hierarchy of roles. The reverse implication—that all roles R and R' for which the condition on the right hand side of (1) holds are in the subrole-/super-role relation—is less obvious. This is an issue we briefly touch on below.

2.3. Role projectability between situation types

In section 3, we examine the problem of role projectability; that is, what principles and structures in an ontology determine, given the set of roles appropriate for one situation type, the set of roles of a related situation type. For example, are there any general statements we can make about the roles in subsituations, given the roles

[1] We use the term *situation* to speak of events and states. An event type is merely a situation type whose instances are events.

[2] We will not delve into the question of whether the entity actually must exist, or if it does, must temporally and spatially overlap the situation.

[3] Examples from the Cyc ontology are from the OpenCyc release of April, 2002, which can be examined or downloaded at www.opencyc.org. Version 1.7 of WordNet can be obtained from www.cogsci.princeton.edu/~wn/.

in a situation? This is particularly important in cases where it is debatable whether to analyze one situation type as a subtype of another or the second as a subsituation of the first (for instance, is *eating a meal* a subtype of *eating*, or is it better analyzed as containing a subevent of eating, along with other subevents such as serving oneself, and if the latter is the preferred analysis, how do the roles of the eating subevent project to the roles of eating a meal?). Another set of cases involves groups of similar situations, such as a group of *walking* events, for which we might wish to project some roles but not others. Finally, we also consider the interaction of role projectability and multiple inheritance.

We can classify projectability issues along three dimensions, as shown in the following table:

	situations	**entities**
type-level	sub/super-types of situations	sub/super-types of entities
individual level	sub-situations/ super-situations	sub-entities/ super-entities

Thus we can examine, for example, whether roles appropriate for a particular type of situation are appropriate for any subtypes of it, or we can examine whether a role played by a particular entity in a situation is also a role played by entities of which it is a part. Type-level projection is concerned with generalization and specialization relations between types (or hypernymy and hyponomy relations in lexical resources like WordNet), while individual-level projection is concerned with mereological (or meronymic) relations between individual situations and entities. One straightforward case is that of roles projecting from situation types to their subtypes, which is entailed by (1). We will not examine all of the possible options for projecting roles here (some of them are highly implausible in any case), but the table helps to situate the issues we examine in sections 3 and 4.

2.4. Parallelism in relations between roles and relations in other elements of ontologies

Often independently of concerns about the relations between concepts, scholars in linguistics and philosophy have been concerned with determining and classifying the roles of participants in situations. When situations and entities are arranged in type hierarchies, it is natural to inquire whether participant roles can be similarly arranged (see, among others, Parker-Rhodes (1978), Ostler (1979), Somers (1987), Lehmann (1997), and Sowa (2000)). In addition to subtype-supertype relations, however, we also find other types of relationships frequently modeled in ontologies. This leads to our second objective, which is to compare the structures of the participant role hierarchy to the other two. To what extent does the role hierarchy parallel the others, and which relationships commonly posited among situation and object types are applicable to roles as well? This will be discussed in section 4.

3. Some cases of role projectability

In this section we consider three cases of role projectability between situations and subsituations. The first concerns the case of a situation that can be regarded as composed of a group of situations of some common type. We suggest that roles of the group situation can be systematically related to those in the subsituations; the latter are subroles of the former. We next examine a more general case motivated by Lehmann's (1998) discussion of situations and roles, in which multiple inheritance in the situation-type hierarchy is pervasive. We argue that freely allowing this kind of multiple inheritance creates complications for the role system and should probably be constrained more than Lehmann envisions, or recast as a form of embedding the parent situation types as subsituations in another type rather than as multiple inheritance. Finally, we note the case of related telic and atelic situation types, which seems to require projection of roles from situations to subsituations, rather than inheritance from situation types to subtypes.

3.1. Groups of events of a common type

One frequent case of situations and subsituations is that of a group of situations of a given types, treated as a group, which itself can be regarded as a situation. This kind of operation is frequently represented in ontologies; Cyc's *GroupFn* is one example.[4] What can we conclude about the roles in the group situation, given the roles in its elements? One possibility is that they are identical. But this seems problematic. Suppose that the role *R* is defined for the situation type of the group's elements, and that in the group event *g*, the participant playing *R* is the mereological sum of all the participants playing role *R* in each of the elements of *g*. We will write this as *R(y,g)*, where *y* is the sum of the individuals playing this role in each of the elements. This is simply a case of the *cumulativity* or *summativity* property of roles (Krifka, 1992, 1998). While for some roles this is a reasonable move, it vitiates the definition of others. It may not cause any difficulties, for example, to regard a group of children running around a playground as the collective *agent* in a group *running* event. However, the *path* role in such an event is a discontinuous set of trajectories, while for a single child running, and for motion of a single body in a continuous time interval generally, the trajectory is continuous. This property of paths is important to maintain; Krifka's (1998) analysis of telicity in motion event relies on it, for example. Another example concerns *source* and *goal* roles in groups of motion events. It is useful to have a rule that either the *source* and *goal* of a motion event are distinct locations, or that the *moving object* has not changed its position (if the motion is a complete revolution in a circle, for instance). But this rule will not apply to groups of motion events; two runners might exchange places, each ending up in the other's starting location. The *source* and *goal* would then be identical in the group event.

A more palatable alternative is to assume that for every role *R* such that *R(x,e)* for some *x* in each element *e* of *g*, there is a super-role of *R*, *R'*, such that *R'(g,y)*, where *g* and *y* are the mereological sums, as above. These super-roles can have some of the properties of the original roles but need not have all of them. For example, the super-role of the *path* role could have discontinuous trajectories as its

[2] This represents some kinds of group situations adequately, but not all. Situations involving joint action or intent, for example, are not always readily decomposed into subsituations of a closely related type.

value, and those of the *source* and *goal* roles would have weaker distinctness conditions. At the same time, nothing precludes using the original roles to describe a situation that can be regarded both as a group of subsituations and as a single situation of the same type as those subsituations; in this case the same participant (a group of entities) will play the role *R*, and hence *R'*. A potential drawback to this approach is that, if we adopt the definition of subrole in (1), we are then committed to treating the type of groups of situations of type *S* as a supertype of *S* (the two types could be identical in some cases, such as at the top of the situation-type hierarchy). However, we see no obvious problems with this move, although this condition is not typically found in ontologies.

3.2. Multiple inheritance and roles

When a situation type is a child of more than one parent type, there are two possible outcomes with regard to the roles in the parent types. One is that two roles from two parent types merge, so that a single participant in an instance of the child type plays both of these roles. From two parent situation types such as *eating in a restaurant* and *eating breakfast* we can construct a type inheriting from both, *eating breakfast in a restaurant*, in which the eater and eaten roles of both parent types are merged; that is, there is a single eater participant and a single eaten participant in a situation of *eating breakfast in a restaurant*. In this case, the roles in the child type must be subroles of the roles in the parent types. This is represented graphically in Figure 1.

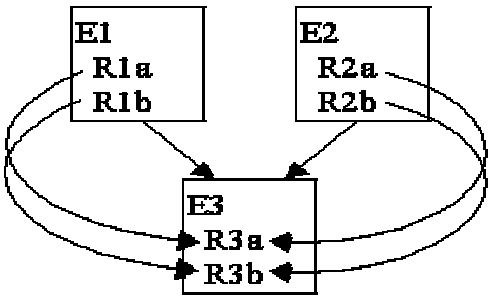


Figure 1. Merged roles in situation-type inheritance

A second possibility is that a role from one parent type does not merge with any other role, remaining distinct in the child type. In Cyc, for example, the type *CausingAnotherObjectsTranslationalMotion* is a subtype of *Movement-TranslationEvent*, which has the roles *objectMoving* and *trajectory*) and of *ActionOnObject*, with the roles *doneBy* and *objectActedOn*. The trajectory and *doneBy* roles remain distinct in the child type. As for the participant that is caused to move, it plays the roles of *objectActedOn* and *objectMoving* in an instance of *CausingAnotherObjectsTranslationalMotion*, but these two roles are not necessarily merged. There is no role reified within the Cyc system that inherits from these two roles. Instead, a rule states that the same participant plays both roles in situations of the type *CausingAnotherObjectsTranslationalMotion*. This second possibility is shown in Figure 2.

This has some implications for some of the ontological structures in Lehmann (1997). Lehmann exemplifies a situation-type hierarchy with increasingly complex types that inherit from multiple parents. For example, there are event types labeled "father gets harmed and angry child then gets revenge", a subtype of the situation types "father gets harmed" and "an angry child gets revenge".[5] Now if roles are inherited from types to their subtypes, this implies that the child type has all the roles of its two parent types. If this kind of type construction is fully productive in the situation-type hierarchy, however, it leads to the uncomfortable conclusion that roles always project from subevents to the events they are part of, since each conjunct can be considered a subevent. Consider the example of taking a trip in a car. We define event types of unlocking a car and driving a car in our hierarchy; the former type has a key as an instrument. Now we define the type of taking a trip in a car, inheriting from these two types of unlocking and then driving a car. By inheritance, this type also will have a key as an instrument, which is the undesirable situation we encountered above. This issue becomes particularly acute when there are two participants in the complex event type that are assigned the same role as a result of inheritance. Consider an event of taking dictation, where one person is reading aloud and another is copying down the words. The reader or writer in the parts of this event can both be considered agents, but we will certainly wish to distinguish these two roles in taking dictation. One solution here, of course, is to provide distinct, more specific roles, such as *reader* and *writer*. But this strategy is not always available; when two events of the same type are combined, the roles in the resulting type will be the same. As an example of this, picture a situation where two people compare versions of a text by having one read aloud and then the other, or a "call and response" situation where one person echoes another's words.

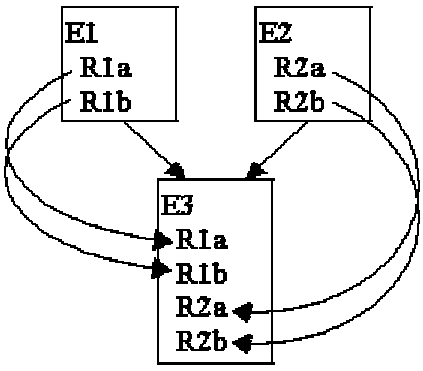


Figure 2. Distinct roles in situation-type inheritance

We could circumvent these problems in several ways. One is to postulate distinct roles for each situation type. Thus a key would play a role in the complex event type just mentioned, but it would not be the same role that it plays in the simpler subevent *unlocking a car*. This

[5] We disregard here the issue of how the temporal order of these two events in the subtype is specified. There must be some mechanism for doing so, however, since the reverse temporal order would describe a very different type of complex event.

allows us to be fully productive in creating complex situation types, at a cost of complicating our system of roles considerably. The number of roles is obviously indefinitely large, as the potential for creating successively more complex event types is unlimited, and there remains a problem of determining when two roles are necessarily filled by the same participant. For instance, a subtype of *unlocking a bicycle* is *unlocking a bicycle that is locked with a Kryptonite lock*. We can classify an instance of such an event in either fashion. This subtype has a distinct role for the key, but we want to equate the roles in the two event types, rather than worrying about whether there are two distinct keys. This representation, in which there are 4 roles, but only two participants, is shown in figure 3.

The large number of roles, and their uniqueness to individual situation types under this option, might become more palatable if we adopt a feature-based analysis of roles, along the lines of Somers (1987), Ostler (1979), Parker-Rhodes (1978), or Sowa (2000). From a linguistic standpoint, for instance, something like such features would be needed to account for regularities in the mapping from roles to syntactic arguments of verbs and nominalizations (see Dowty (1991) and Wechsler (1995) for similar accounts that can cast in a feature-based model). But in some sense we have merely shifted the problem from projectability of roles to projectability of features. If the features of a key as an instrument in unlocking a car are projected to its role in driving, why is it so odd to say that "we drove to the store with the key"?

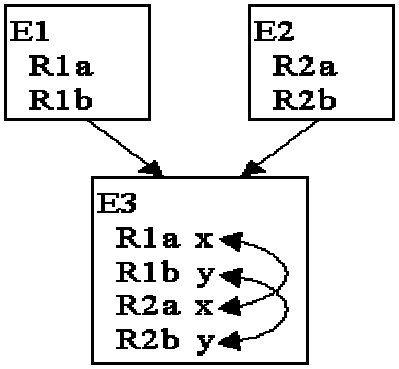


Figure 3. Distinct roles, shared by identical participants in subtype

Another option would be to structure the set of roles more richly, so that both sets of roles are inherited in a complex event, maintained as two separate structures (with additional roles potentially added as well). This option is in the spirit of feature structure representations, in which structures can be embedded recursively (Lehmann may allude to something similar when he refers to "structural specification"). A representation of this kind, in which the role-sets of the parent events are embedded within new role features in the child event, is shown in figure 4. The roles *R3a* and *R3b* within *E3* are filled by subevents; they might be relations such as *cause* and *effect*, for example. This allows roles to be inherited, albeit in a non-uniform way, which depends on how the parent situation types are combined in the child type. Furthermore, it is necessary to specify when a single participant fills roles in each part of the situation. In the type "father gets harmed and angry child then gets revenge on perpetrator", the same individual (the perpetrator) plays a role in both subevents.

Yet another approach would be to restrict the situation-type hierarchy to a set of types for which role inheritance makes sense. The trouble with this is that it seems too restrictive for many purposes. We sometimes do wish to refer to "composite" event types, like commuting to work on a bicycle, moving from one city to another, or holding a presidential election. But some kind of compromise position may be possible. We might maintain the kind of role inheritance that appears useful by designating one parent type as the "principal type", whose roles are inherited. For commuting by bicycle, the principal parent might be something like riding a bicycle, and the roles of the bicycle, the rider, the origin, and the destination would be inherited. Other, "minor" events involved in commuting, like locking and unlocking the bicycle, would not be involved in role inheritance. A subgraph of the hierarchy of situation types, filtered by "main event" or "principal type", links might be homomorphic to the role hierarchy. This approach seems reasonable for many of the situation types that we would be likely to reify in an ontology. It may apply less well to elaborate and complex events with many participants, such as political elections, which have many specialized roles, and would not necessarily inherit many of them from their parents representing their subevents. In some of these complex event types, the notion of a "main event" might not make much sense.

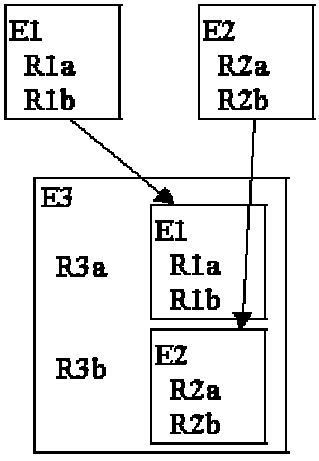


Figure 4. Embedded role-sets in subtype

From a linguistic standpoint, multiple inheritance in the situation-type hierarchy interacts with issues of linking; that is, the syntactic realization of predicates and their arguments. Subject selection is one good example; verbs denoting commercial transactions refer to situations in which there are two *agents*, as do causative verbs in the many languages that allow causativization of verbs denoting agentive situations. In each of these cases an accurate account of subject selection must appeal to more than the agentive status of a participant, since more than one participant plays an agentive role (see (Dowty 1991), (Wechsler 1995).and the Framenet system developed by

Filllmore and others for some approaches to this problem). Designating one of the subevents as the "main" or "salient" event for linguistic purposes, as in Framenet, accords well with the foregoing suggestion, although linguistic evidence is only a rough guide in these matters.

3.3. The inheritance of properties

One final issue regarding roles and subroles concerns how strictly we wish to enforce inheritance of properties. The OntoClean proposals of Guarino and Welty (2002), for example, place high importance on transmitting various properties dependably in inheritance. For instance, they argue against a pervasive characteristic of Cyc, that individual object types commonly inherit from the stuff types of which the objects are composed (thus *Ocean* is a subtype of *Water* in Cyc). When we examine the comparable situation in the realm of situations, we are led to the conclusion that telic situation types, such as eating an apple or painting a wall, should not be regarded as subtypes of atelic types such as eating or painting. The latter types are cumulative: two eating events may be combined and treated as a single eating event, but two events of eating an apple cannot be regarded as a single larger event of eating an apple.[6] This suggests that, parallel to the object and stuff types, telic and atelic event types should not be in a subtype-supertype relationship. If so, then telic event types will not inherit the roles of corresponding atelic event types. Instead, we could adopt a projectability rule that states: if *e* is an event of telic event type *T*, and *T* is "composed" of events of atelic type *A* (just as oceans are composed of water), then *e* also has those roles. In some cases, there may be no roles specified for events of type *T*, independently of type *A*. In others, such as many telic movement event types, additional roles are present, including *source* and *goal* roles.

In this case, then, we are led to a conclusion that is roughly the reverse of what we advocated in the case of groups of situations. For group situations, a consideration of roles for the group and for the subsituations comprising it led us to suggest that the group situation type is a supertype of the type of the elements. For the case of atelic and telic situation types, which might initially appear to be in a supertype-subtype relation, a re-examination of this assumption leads us to posit projection of roles from (atelic) subsituations to (telic) situations.

In sum, we see that there are unlikely to be simple general principles regulating the projection of roles between situations and their sub- or super-situations, although there do appear to be some useful, more specific principles covering some cases of interest.

4. Parallelism between roles and other elements in ontologies

In this section we explore what parallelisms may exist between the hierarchy of participant roles and other types of ontologies. Besides supertype-subtype relations, mereological relations are crucial in ontologies and in lexical resources like WordNet. Lexical resources also frequently employ an antonymy relation between words, though it is less clear that this is coherent ontological relation and ontologies emphasize this much less. In this section we will investigate to what extent these other relations can be applied to roles. In doing so, we will continue to mention issues of role projectability, this time with respect to entities and their parts.

4.1. Specialization/generalization (hyponymy/hypernymy)

Concept specialization is represented in WordNet with hyponymy links, and in Cyc with the predicate *genls* (and some extensions of it for relations). These apply both to entity types (or nouns in WordNet) and situation types (or verbs in WordNet, which then refers to this relation as "troponymy"). The comparable relationship for roles is simply the subrole relation; if one role is a subrole of another, then any participant that plays the first role in a situation necessarily also plays the second. This is the chief organizing relation for the hierarchy of roles, as it is for object and situation types.

However, we would like to remark here on one more linguistically relevant issue, since much of this same machinery is brought to bear on computational lexicons, including WordNet. Because the mapping from semantic roles to syntactic arguments is not completely semantically determined and displays some arbitrary variation, we cannot assume that hyponyms of a verb will exhibit the same mapping as that of the verb itself. In some cases, for example, an argument is incorporated in the verb (e.g., "spread butter on the bread" vs. "butter the bread", "put the money in the pocket" vs. the "pocket the money"). In others, the mapping is simply different (e.g., "eat oysters" vs. "dine/gorge on oysters"). This means that syntactic patterns are not necessarily reliable indicators of participant roles, and although hyponymy usually does imply inheritance of participant roles, corresponding roles may not occupy corresponding syntactic positions.

4.2. Partial roles (meronymy/holonymy)

Meronymy/holonymy, the lexical part/whole relation, and other mereological relations in ontologies, appear to be more complex, with several discernable subtypes. For example, Winston, Chaffin, and Hermann (1988) differentiate seven types of meronym: component-object (branch/tree), member-collection (tree/forest), portion-mass (slice/cake), stuff-object (aluminum/airplane), phase-process (adolescence/growing up), feature-activity (paying/shopping), and place-area (Baltimore/Maryland). Iris, Litowitz, and Evens (1988) acknowledge only four, however: functional part (wheel/bicycle), segment (slice/loaf), member (sheep/flock) and subset (meat/food), which is really specialization rather than meronymy. Likewise Cyc distinguishes numerous part/whole relations, including ingredients, physical and abstract parts, and subevents. The WordNet hierarchies employ just a single meronym link type, used only in the noun hierarchy. Meronymy applies just as usefully, however, to situation types (or verbs in WordNet), as we have been assuming throughout this paper. The type of meronymy called "phase-process" by Winston, Chaffin and Hermann

[6] Note that one and the same event can be regarded as both atelic or telic; eating an apple is certainly also eating. The telicity distinction is at the situation-type level, not at the individual level.

(1988) relates pairs of nouns and gerunds such as *adolescence/growing_up*. Feature-activity meronymy relates pairs of gerunds such as *paying/buying* or *steering/driving*. In short, events can be said to have component parts just as objects have them. The analogy to meronymy in the domain of participant roles is much less obvious than the specialization parallel, however.

We can begin by offering a definitions of "partial roles", as a mereololgical counterpart to the definition of subroles in (1):

(2) *R'* is a partial role of *R* iff:
R(x,e) → ∃*x,e*: *x'* is a part of *x* and *e'* is a subsituation of *e* and *R'(x',e')*

Unlike physical part and subsituation relationships, which are ubiquitous and obviously crucial to ontologies, there are relatively few instances of roles in this relationship that we are aware of, beyond the trivial case where *R* = *R'*, *x* = *x'*. and *e* = *e'*. Two cases are exemplified in the following sentences, where the participant denoted by the object of 'with' or 'by' is a part of another participant. Thus the "instrument" role is a partial role of *agent* in a. and the "body part" role is a partial role of the *grabbed* participant (or *theme*, or *affected object*) in b:

(3) a. I bumped the vase with my elbow.
b. I grabbed the iguana by the tail.

A third case of partial roles involves the *moving object* in movement events. In such events the parts of the object also move during at least some subintervals of the event, so the role *moving object* is partial to itself in a non-trivial way. In a parallel fashion, some roles in states are non-trivially self-partial. If someone owns a car for a year, that person owns the engine for the first six months, and if a beam supports a roof for a year, it is plausible to infer that a section of the beam supports a section of the roof for any period within that year.

Despite these cases, it appears that this type of part/whole relationship between roles is rare, and not particularly useful in inference. Possibly this is due to the relational character of roles, mediating between situations and their participants. We will now consider a more widespread phenomenon, the projection of role from participant entities to larger entities of which those participants are parts.

4.3. Projection of roles from entities to super-entities

We now examine the question of which roles can be projected from parts to wholes and vice versa; that is, if an object plays a role in a situation do larger objects of which it is a part and smaller objects that are parts of it also play that role in the situation? It should be clear that when this is the case, the role in question violates Krifka's uniqueness of objects property (Krifka 1992, 1998). Two kinds of roles for which this does seem to be true are roles of *source* and *goal* in motion events. For example, the following inferences seem valid:

(4) I flew from Baltimore to Boston. therefore,
I flew from Maryland to Boston. and
I flew from Baltimore to Massachusetts.

This inference has limits, in that the super-region cannot include both the origin and the destination of the trip, however, so the following are aberrant:

(5) #I flew (from the U.S.) (to the U.S.) and,
#I flew from the U.S. to Boston. and,
#I flew from Baltimore to the U.S.

The *path* role, in contrast, can be projected down to parts of the trajectory, but not to larger paths:

(6) Kim hiked (all of) the John Muir Trail. therefore,
Kim hiked the Tahoe-Yosemite Trail.

As Krifka (1992, 1998) has pointed out, we can make similar inferences from parts to wholes in the case of roles that involve contact or perception, as the following examples illustrate:

(7) John touched the door handle. therefore,
John touched the door.

(8) Kim rammed Sandy's bumper. therefore,
Kim rammed Sandy's car.

(9) The jar contacts the countertop. therefore,
The jar contacts the counter.

Note also that in situations involving both motion and contact, the contact inference is allowed even if the motion is not:

(10) I shook a link of the chain. therefore,
I touched the chain (even if I didn't shake it).

As for roles involving perception, the same pattern seems to apply, though the inference seems less solid:

(11) Fred saw the elephant's trunk. therefore,
Fred saw the elephant.

(12) Alice smelled the roasted chicken. therefore,
Alice smelled the meal.

As the story about the blind men and the elephant suggests, however, there is some uneasiness about such inferences. Perception differs from contact in this respect.

Finally, there are situation types in which one participant stands in a relationship of superiority to another, denoted by verbs such as 'exceed', 'surpass', 'dwarf', and verbs prefixed with 'out-'. In these cases, it arises virtually a matter of definition that the superior participant's role projects to objects of which it is a part, and the inferior one's role to its parts. This is exemplified in the following sentences:

(13) Nitrous oxide levels exceeded the Federal standards. therefore,
Smog levels exceeded the Federal standards.

(14) Bach outlived Vivaldi. therefore,
The Bach family outlived Vivaldi.

(15) Russia dwarfs Korea. therefore,
Russia dwarfs North Korea.

There are many roles for which projection to parts or wholes does not follow, except in some metaphorical or metonymic sense, including most roles involving agency, motion, and affectedness. In sum, "spatial" roles (including those that are appropriate for situation types whose linguistic realization is metaphorically based on spatial relationships) exhibit some projection properties that should prove useful in inference. But there is no direct parallel among roles to part/whole relationships of the type that apply ubiquitously to entities and situations.

4.4. Antonymy/opposition

Another relation in WordNet, more explicitly lexical, is antonymy, although as Miller (1998) points out, it is not a fundamental an organizing relation between nouns. True antonymy is present in the verb hierarchy, as well as among adjectives. Change-of-state verbs, for example, have antonymous counterparts quite similar to nouns, although the verb pairs don't normally share parents (e.g., 'lengthen'/'shorten' and 'strengthen'/'weaken'). Relations of opposition occur as well, where there is no common superordinate or entailed verb unique to the pair (e.g., 'give'/'take', 'buy'/'sell').

Antonymy is closely tied to lexical properties and not a coherent ontological relationship, but some aspects of it can be singled out and represented as conceptual relationships. For example, reversative actions (*zipping* and *unzipping*, *loading* and *unloading*, *arriving* and *leaving*, *creating* and *destroying*) exemplify a fairly coherent notion of opposition that bears on participant roles. We cannot say that the event types in each pair have the same roles; for example, *loading* and *arriving* both have a *goal* role, but may lack a *source*, while *unloading* and *leaving* are the opposite. But it is probably fair to say that each role of an event type has a counterpart in the corresponding reversative event type. The same may hold true for other sorts of opposites (e.g., *helping* and *hindering*, *benefiting* and *suffering*, *believing* and *doubting*), though in many of these cases we are more likely to say that the role's counterpart is itself. It seems less meaningful to posit a counterpart relationship between roles in some other types of situations sometimes thought of as "opposites" (being awake or asleep, liking and disliking, and many others), let alone antonyms in the domain of properties and objects.

4.5. Complementary roles

Another relation between two roles that seems worthwhile is what we term *complementarity*. For some situation types, we know that when one role is present, another role must be also. We then say that this second role is complementary to the first. Complementarity may unidirectional or bi-directional, but most of our examples will involve the latter case. Some examples of such roles are *buyer* and *seller*, *buyer* and *payment*, *moving object* and *path*, *driver* and *vehicle*, and *perceiver* and *perceived*. One application of a complementarity relation in inference should be fairly clear; it allows us to postulate the existence of a participant filling a role when the participant playing the complement role is known to be present. However, this sort of inference is probably equally simply performed with reference to situation types, as long as they specify which roles are necessary and appropriate. The complementarity relation bears some resemblance to meronymy and to the "partial role" relations; it could even be considered a type of partial role relation applied to situations, disregarding the requirement in (2) that participants playing each role be in a part-whole relationship. Complementarity certainly has counterparts in the entity and situation domains. The existence of a hole depends on the existence of a cavity wall, and the two transfer subevents of a commercial transaction seem complementary in much the same way that the roles are.

Roles that are complementary and that, in a given situation type, are entailed to be filled by the same participant, may violate the reverse of the implication in (1). That is, if *R'* is a complementary role of *R*, and a situation type is constrained so that *R(x,e)* and *R'(x,e)* for any situation *e* of that type, then a bi-directional interpretation of (1) would treat *R* as a subrole of *R'*.[7] There may be legitimate grounds, however, to distinguish two participant roles in such situations. For example, someone who is talked into performing an action is both a *addressee* and a *performer* (of the action). It is possible in such cases to create a role specific to that type inheriting from the two roles *R* and *R'*, but it does not always seem desirable to do so. We leave this question open.

5. Conclusions

We have seen that an examination of the relations among roles can be fruitful in illuminating other aspects of ontologies and lexical resources. Considering the question of role projectability has shown that permitting multiple inheritance to operate without constraint in the situation-type hierarchy is problematic, and that other mechanisms do not cause the same difficulties for inheritance of roles from situation types to their subtypes. We have also seen how role inheritance interacts with two particular cases of situations and subsituations: a group of like situations and telic situations composed of atelic subsituations. In these two cases, role projectability reveals interesting relationships among situation types in ontologies.

Roles parallel situation and entity types in constituting a hierarchy, but, perhaps because of their inherently relational nature, the parallelism beyond that is limited. While we can formulate coherent definitions of relationships between roles that parallel the mereological relations that are so pervasive among situations and entities, their usefulness is less apparent. In contrast, the complementarity relation between roles is widespread and its utility in inference clear.[8]

6. References

Chaffin, R., Hermann, D.J. and Winston, M.E., 1988. An empirical taxonomy of part-whole relations: Effects of the part-whole relation type on relation identification. *Language and Cognitive* Proceses, 31, pp. 17-48.

[7] If the two roles are complementary to each other, then each would be a subrole of the other, hence they would be the same role.

[8] We would like to thanks the referees for comments which have improved this paper.

Dowty, D., 1989. On the Semantic Content of the Notion 'Thematic Role'. In G. Chierchia, B. Partee, and R. Turner (eds.), *Properties, Types, and Meaning*. Dordrecht: Reidel.

Dowty, D., 1991. Thematic Proto-Roles and Argument Selection. *Language* 67:3, pp. 547-619.

Fellbaum, C. (ed.) 1998. *Wordnet: An Electronic Lexical Database*. Cambridge, MA: MIT Press

Guarino, N., and Welty, C., 2002. Evaluating Ontological Decisions With OntoClean. *Communications of the ACM (CACM)*, 45:2, pp. 61-65.

Iris, M.A., Litowitz, B.E. and Evans, M., 1988. Problems of the part-whole relation. In M. Evans (ed.) *Relational models of the lexicon*, Cambridge: Cambridge University Press.

Krifka, M., 1992. Thematic Relations as Links between Nominal Reference and Temporal Constitution. Iin I. Sag and A. Szabolcsi (eds.), *Lexical Matters*. Stanford, CA: CSLI Publications.

Krifka, M., 1998. The Origins of Telicity. In S. Rothstein (ed.), *Events and Grammar*. Dordrecht: Kluwer Academic Publishers.

Ladusaw, W., and Dowty, D., 1988. Toward a Nongrammatical Account of Thematic Roles, In W. Wilkins (ed.), *Syntax and Semantics 21: Thematic Relations*. San Diego: Academic Press.

Lehmann, F., 1997. Big Posets of Participatings and Thematic Roles. In P. Eklund, G. Ellis, and G. Mann (eds.), *Conceptual Structures: Knowledge Representation as Interlingua*. Heidelberg: Springer.

Miller, G.A., 1998. Nouns in Wordnet. In C. Fellbuam (ed.), *Wordnet: An Electronic Lexical Database*. Cambridge, MA: MIT Press.

Ostler, N., 1979. *Case Linking: A Theory of Case and Verb Diathesis Applied to Classical Sanskrit*. Ph.D. dissertation, MIT.

Parker-Rhodes, A., 1978. *Inferential Semantics*. Atlantic Highlands, NJ: Harvester Press.

Pustejovsky, J., 1995. *The Generative Lexicon: A theory of computational semantics.* Cambridge, MA: MIT Press.

Somers, H., 1987. *Valency and Case in Computational Linguistics*. Edinburgh: Edinburgh University Press.

Sowa, J., 2000. *Knowledge Representation: Logical, Philosophical, and Computational Foundations*. Pacific Grove, CA:.Brooks Cole Publishing Co.

Wechsler, S., 1995. *The Semantic Basis of Argument Structure*. Stanford, CA: CSLI Publications.

Restructuring WordNet's Top-Level: The *OntoClean* approach

Alessandro Oltramari[(1)], Aldo Gangemi[(2)], Nicola Guarino[(1)], Claudio Masolo[(1)]
[(1)]LADSEB-CNR*, Padova, Italy: {Nicola.Guarino, Alessandro.Oltramari, Claudio.Masolo}@ladseb.pd.cnr.it
[(2)]ISTC-CNR, Rome, Italy: gangemi@ip.rm.cnr.it

Abstract

In this paper we propose an analysis and a rearrangement of *WordNet's* top-level taxonomy of nouns. We briefly review WordNet and identify its main semantic limitations, in the light of the ontology evaluation principles lying at the core of the *OntoClean* methodology. Then we briefly present a first version of the *OntoClean Top* (OCT) ontology, and show how WordNet can be aligned with it. The result is a "cleaned-up" WordNet, which is meant to be conceptually more rigorous, cognitively transparent, and efficiently exploitable in several applications.

1 Introduction

The number of applications where WordNet is being used more as an ontology than just as a lexical resource seems to be growing more and more. To be used as an ontology, however, some of WordNet's lexical links need to be interpreted according to some formal semantics, which tells us something about "the world" and not (just) about the language. One of such links is the hyponym/hypernym relation, which corresponds in many cases to the usual subsumption (or IS-A) relation between concepts. An early attempt at exploring the semantic and ontological problems lying behind this correspondence is described in (Guarino, N., 1998).

In the recent years, we developed a methodology for testing the ontological adequacy of taxonomic links called OntoClean (Guarino, N. & Welty, C., 2002; Guarino, N. & Welty, C., 2002), which was used as a tool for a first systematic analysis of WordNet's upper level taxonomy of nouns (Gangemi, A. *et al.*, 2001). The first version of OntoClean was based on an ontology of properties (unary *universals*), characterized by means of meta-properties. We are now extending OntoClean with an ontology of *particulars* called OCT (OntoClean Top ontology), which is presented here in some detail, although still in an informal way. The OCT will be the first module of a minimal library of *foundational ontology* that we shall develop within the *WonderWeb*[1] project.

This paper is structured as follows. In the next section we present an extension of our FOIS paper (Gangemi, A. *et al.*, 2001), concerning some ontological inadequacies of WordNet's taxonomy of nouns. Then we introduce the most recent version of our OntoClean Top ontology, and discuss the preliminary results of an alignment work aimed at improving WordNet's overall ontological (and cognitive) adequacy, and facilitate its effective deployment in practical applications.

2 WordNet's Preliminary Analysis

2.1 Experiment Setting

We applied our methodological principles and techniques to the noun synsets taxonomy of WordNet 1.6.To perform our investigation, we had to adopt some preliminary assumptions in order to convert WordNet's databases[2] into a workable knowledge base. At the beginning, we assumed that the hyponymy relation could be simply mapped onto the subsumption relation, and that the synset notion could be mapped into the notion of concept. Both subsumption and concept have the usual description logics semantics (Woods, W. A. & Schmolze, J. G., 1992). In order to work with named concepts, we normalized the way synsets are referred to lexemes in WordNet, thus obtaining one distinct name for each synset: if a synset had a unique noun phrase, this was used as concept name; if that noun phrase was polysemous, the concept name was numbered (e.g. window_1). If a synset had more than one synonymous noun phrase, the concept name linked them together with a dummy character (e.g. Equine$Equid).

Firstly, we created a Loom[3] knowledge base, containing, for each named concept, its direct super-concept(s), some annotations describing the quasi-synonyms, the gloss and the synset topic partition, and its original numeric identifier in WordNet; for example

```
(defconcept Horse$Equus_Caballus
    :is-primitive Equine$Equid
    :annotations ((topic animals)
    (WORD |horse|)
    (WORD |Equus caballus|)
    (DOCUMENTATION "solid-hoofed herbivorous quadruped domesticated since prehistoric times"))
    :identifier |101875414|)
```

noun entries	116364
equivalence classes: synonyms, spelling variants, quasi-synonyms	50337
noun synsets (with a gloss and an identifier for each one)	66027
nouns	95135
monosemous nouns	82568
polysemous nouns	12567
one-word nouns	70108
noun phrases	25027

Table1: Elements processed in the Loom WordNet kb

The elements processed in the Loom WordNet knowledge

*In the process of moving to ISTC-CNR, Rome, Italy.

[1] http://wonderweb.semanticweb.org/

[2] We used the Prolog WordNet database, the Grind database, and some others from the official distribution.

[3] Loom is a knowledge representation system that implements a quite expressive description logic (MacGregor, R. M., 1991).

base are reported in Table 1. We report in Figure 2 an overview of WordNet's noun top-level as translated in our Loom knowledge base. The nine Unique Beginners are shown in boldface.[4]

2.2 Main problems found

Once the Loom WordNet was created, we systematically applied the OntoClean methodology to the upper taxonomy of noun senses. Let us discuss now the main ontological drawbacks we found after applying this cleaning process.

2.2.1 Confusion between concepts and individuals

The first critical point was the confusion between concepts and individuals. For instance, if we look at the hyponyms of the Unique Beginner Event, we'll find the synset Fall - an individual - whose gloss is “the lapse of mankind into sinfulness because of the sin of Adam and Eve”, together with conceptual hyponyms such as Social_Event, and Miracle.[5] Under Territorial_Dominion we find Macao and Palestine together with Trust_Territory. The latter synset, defined as "a dependent country, administered by a country under the supervision of United Nations", denotes a general kind of country, rather than a specific country as those preceding it. If we go deeper in the taxonomy, we find many other examples of this sort. For instance, the hyponyms of Composer are a mixture of concepts and instances: there are classes corresponding to different special fields, such as Contrapuntist, or Songwriter, and examples of famous musicians of the past, such as Bach, and Beethoven.

Under Martial_Art, whose top hypernym is Act, we find Karate, and Kung Fu, but these synsets do not stand for concepts, they represent individuals, namely particular examples of martial arts.

If we look through Organization, under the branch whose root is Group, we find conceptual hyponyms such as Company, Alliance, Federation, Committee, together with instances like Irish_Republican_Army, Red Cross, Tammany Society[6], and so on.

We face here a general problem: the concept/individual confusion is nothing but the product of an “expressivity lack”. In fact, if there was an INSTANCE-OF relation, we could distinguish between a concept-to-concept relation (subsumption) and an individual-to-concept one (instantiation).

2.2.2 Confusion between object-level and meta-level: the case of Abstraction

The synset Abstraction_1 seems to include both object-level concepts, such as Set, Time, and Space, and meta-level concepts such as Attribute and Relation. From the corresponding gloss, an abstraction “is a general concept formed by extracting common features from specific examples”. An abstraction seems therefore intended as a psychological process of generalization, in accordance to Locke's position ((Lowe, E. J., 1998), p.211). This meaning seems to fit the latter group of terms (Attribute, Relation, and possibly some hyponyms of Quantity), but not to the former. Moreover, it is quite natural to consider attributes and relations as meta-level concepts, while set, time, and space, seem to belong to the object domain.

2.2.3 OntoClean constraints violations

A core aspect of OntoClean is the analysis of subsumption constraints induced by the identity, rigidity, and unity meta-properties. In our analysis, we only found rigidity violations. We suspect that there are two reasons why we didn’t observe other kinds of violation: on one hand, we limited our analysis to the upper levels, where the criteria of identity and unity are very general; on the other hand, WordNet tends, notoriously, to multiply senses, so the chances of conflict are relatively limited.

The most common violation we have registered is bound to the distinction between roles and types. A role cannot subsume a type. Let's see an important clarifying example.

In its first sense, Person (which we consider as a type) is subsumed by two different concepts, Organism and Causal_Agent. Organism can be conceived as a type, while Causal_Agent as a formal role. The first subsumption relationship is correct, while the second one shows a rigidity violation. We propose therefore to drop it.

Someone could argue that every person is necessarily a causal agent, since ‘agentivity’ (capability of performing actions) is an essential property of human beings. Causal_Agent should therefore be intended as a synonym of ‘intentional agent’, and considered as rigid. But, in this case, it would have only hyponyms denoting things that are (essentially) causal agents, including animals, spiritual beings, the personified Fate, and so on. Unfortunately, this is not what happens in WordNet: Agent, one of Causal_Agent hyponyms, is defined as: "an active and efficient cause; capable of producing a certain effect; (the research uncovered new disease agents)". Causal_Agent subsumes roles such as Germicide, Vasoconstrictor, Antifungal. Instances of these concepts are not causal agents essentially. This means that considering Causal_Agent as rigid would introduce further inconsistencies.

These considerations allow us to add a pragmatic guideline to our methodology: when deciding about the formal meta-property to attach to a certain concept, it is useful to look at all its children.

2.2.4 Heterogeneous levels of generality

Going down the lower layers of WordNet's top level, we register a certain ‘heterogeneity’ in their intuitive level of generality. For example, among the hyponyms of Entity there are types such as Physical_Object, and roles such as Subject. The latter is defined as “something (a person or object or scene) selected by an artist or photographer for graphic representation”, and has no hyponyms (indeed, almost any entity can be an instance of Subject, but none is necessarily a subject)[7].

For Animal (subsumed by Life_Form) this heterogeneity becomes clearer. Together with classes such as Chordate, Larva, Fictional_Animal, etc., we find out more specific concepts, such as Work_Animal, Domestic_Animal,

[4] Note that the sense numeration reported in our Loom kb is different from the WordNet's original one. Nevertheless, the reader will easily recognize the synsets we are referring to.

[5] In the text body, we usually do not report all the synonyms of a synset (or their numeration), but only the most meaningful ones.

[6] “A political organization in New York city (late 1800’s early 1900’s) seeking political control by corruption and bossism”.

[7] We can draw similar observations for relation_1 and set_5 with respect to abstraction_1, etc.

Mate_3, Captive, Prey, etc. We are induced to consider the formers as types, while the latters as roles.
Although problematic on the side of ontological distinctions among event-classes, the hyponyms of Phenomenon_1 represent another meaningful example of heterogeneity. At the same taxonomic level there are "reasonably" general synsets like Natural_Phenomenon and Process together with a specific concept like Consequence, which could be modeled as anti-rigid (every event can be a consequence of the occurring of a previous event, but we could assume that this is not the essential characteristic of the event itself[8]).
In short, intuitively some synsets sound too specific when compared to their siblings. Look at them from the formal point of view we are developing, we can pinpoint their "different generality" by means of the distinction between types and roles.

3 The OntoClean Top Ontology

Before presenting our (still preliminary!) OCT ontology, a couple of clarifications may be useful. First of all, we do *not* intend this as a candidate for a "universal" standard ontology. Rather, we support the vision of a *library* of *foundational ontologies*, reflecting different commitments and purposes. In our opinion, the most important challenge today is not so much the agreement on a monolithic set of ontological categories, but rather the careful isolation of the fundamental ontological options and their formal relationships. If general ontologies reflecting different commitments and purposes are described in terms of these formal notions, then we can hope they will form a library of "foundational" ontologies accessible in a modular way, keeping the necessity of largely shared ontological commitments to the very minimum, and making the rationales and alternatives underlying the different ontological choices as explicit as possible. This is one of the goals of the *WonderWeb* project, where the OCT ontology will be linked to other foundational ontologies.
A second clarification concerns the general attitude underlying our ontological choices. The OCT ontology has a clear *cognitive bias*, in the sense that we aim at capturing the ontological categories lying behind natural language and human commonsense. Hence, we do not claim that our categories have "deep" metaphysical implications related to the intimate nature of the world: rather, they are thought of as "conceptual containers" useful to describe ontologies as cognitive artifacts ultimately depending on human perception, cultural imprints and social conventions. So, especially with respect to natural language, our attitude is more "descriptive" than "revisionary" (Strawson, P. F., 1959; Loux, M. J., 1998).
Finally, we have to point out that the ontology presented here is an ontology of *particulars*. Properties and relations are therefore not part of its domain. Some proposals for a ontology of properties have been made in (Guarino, N. & Welty, C., 2000). We are not aware of any systematic work on the ontology of relations.

[8] For instance, the extinction of dinosaurs could have be the consequence of the impact of an asteroid on the Earth, or of a sudden glaciation, or of a mortal epidemic – scientists are not sure about this – but in terms of ontology of events, it is a conclusive event, at most an annihilation event, and there is no need (and here no possibility) to model it as a consequence.

3.1 General notions

Before introducing the OCT categories, let us first introduce the general notions we shall use to characterize them. Some of these notions (like rigidity and unity) have already been defined in previous papers (respectively, (Guarino, N. & Welty, C., 2002) and (Gangemi, A. *et al.*, 2001)), and will not be discussed here. So we shall limit ourselves to the basic distinction between *enduring and perduring* entities, and the varieties of dependence relationships involving particulars.[9] We shall keep the discussion to an informal, introductory level; a rich axiomatization will be presented in a forthcoming paper.

3.1.1 Enduring and perduring entities

A fundamental distinction we assume in the OCT ontology is that between *enduring* and *perduring* entities. This is almost identical, as we shall see, to the distinction between so-called *continuants* and *occurrents* (Simons, P., 1987), which is still being strongly debated both in the philosophical literature (Varzi, A., 2000) and within ontology standardization initiatives[10]. Again, we must stress that this distinction is motivated by our cognitive bias: we do not commit to the fact that both these kinds of entity "really exist", and we are indeed sympathetic with the recent proposal made by Peter Simons, that enduring entities can be seen as equivalence classes of perduring entities, as the result of some kind of abstraction mechanism (Simons, P., 2000).
But let us see what this distinction is about. The difference between enduring and perduring entities (which we shall also call *endurants* and *perdurants*) is related to their behavior in time. Endurants are always *wholly* present (i.e., all their proper parts are present) at any time they are present. Perdurants, on the other hand, just extend in time by accumulating different temporal parts, so that, at any time they are present, they are only *partially* present, in the sense that some of their proper parts (e.g., their previous phases) may be not present. For instance, the piece of paper you are reading now is wholly present, while some temporal parts of your reading are not present any more. Philosophers say that endurants are entities that *are in time*, while lacking however temporal parts (so to speak, all their parts travel with them in time). Perdurants, on the other hand, are entities that *happen in time*, and can have temporal parts (all their parts are fixed in time).
This different behavior affects the notion of change in time. Endurants can "genuinely" change in time, in the sense that the very same whole endurant can have incompatible properties at different times; perdurants cannot change in this sense, since none of their parts keeps its identity in time. To see this, suppose that an endurant has a property at a time *t*, and a different, incompatible property at time *t':* in both cases we refer to the whole object, without picking up any particular part. On the other hand, when we say that a perdurant has a property at *t*, and an incompatible property at *t'*, there are always two different parts exhibiting the two properties.
We have already mentioned that endurants and perdurants can be taken as synonyms of the more common terms

[9] In the OntoClean taxonomy evaluation methodology only dependence between properties is used.
[10] See for instance the extensive debate about the "3D" vs. the "4D" approach at www.suo.org.

continuants and *occurrents*. We prefer however the adopted terminology, because the continuants/occurrents distinction is sometimes considered only within so-called *concrete* entities, while, as we shall see, we take it as spanning the whole domain of particulars, including abstracts that we shall consider as endurants. Finally, we shall take *occurrence*, and not *occurrent*, as synonym of *perdurant*, since it seems natural to use *occurrent* to denote a type (a *universal*), whose instances are occurrences (*particulars*).

The endurants/perdurants distinction evidences the general necessity of temporally indexing the relationships within endurants. This means that, in general, it is necessary to know *when* a specific endurant bears a certain relation to other endurants. Consider for instance the classical example of Tibbles the cat (Simons, P., 1987): Tail is part of Tibbles before the cut but not after it, i.e. we have to "temporalize" the part relation: *P*(Tail, Tibbles, *before*(cut)) and ¬*P*(Tail, Tibbles, *after*(cut)).

With respect to a temporalized relation *R*, we can distinguish *R-constant* endurants from *R-variable* endurants. An endurant *e* is called *R-constant* iff, when $R(x_1, \dots, x_n, e, t)$ holds for a temporal interval *t*, then $R(x_1, \dots, x_n, e, t')$ also holds whenever *e* is present at *t'*.

We can also strengthen this definition introducing the modal notion of an *R-invariant* endurant. An endurant *e* is called *R-invariant* iff, if it is possible that $R(x_1, \dots, x_n, e, t)$ then necessarily $R(x_1, \dots, x_n, e, t)$ holds whenever *e* is present at *t'*.

For the purpose of characterizing the OCT categories, the property of being constant (or invariant) with respect to the parthood relation (*mereologically constant* (*invariant*)) has a special relevance. For example, we usually take ordinary material objects as mereologically variable, because during their life they can lose or gain parts. On the other hand, amounts of matter are taken as mereologically invariant (all their parts are *essential part*), and so on.

3.1.2 Dependence

Let us now introduce informally some useful definitions based on the notion of dependence, adapted from (Thomasson, A. L., 1999). We focus here on *ontological dependence* (holding primarily between particulars, and only by extension between properties), to be distinguished from *notional dependence*, which only holds between properties).

A particular *x* is *specifically constantly dependent* (SCD) on another particular *y* iff, at any time *t*, *x* can't be present at *t* unless *y* is also present at *t*. For example, a person might be specifically constantly dependent on its brain.

A particular *x* is *generically constantly dependent* (GCD) on a property ϕ iff, at any time *t*, *x* can't be present at *t*, unless a certain instance *y* of ϕ is also present at *t*. For example, a person might be generically constantly dependent on having a heart.

1.2 The OntoClean Top Categories

The most general kinds of particulars assumed in the OntoClean Top ontology are described in Figure 1. They are assumed to be mutually disjoint, and covering the whole domain of particulars. They are also considered as *rigid* properties, according to the OntoClean methodology that stresses the importance of focusing on these properties first.

Quality
Quality Region
Aggregate
- Amount of matter
- Arbitrary collection

Object
- Physical Object
  - Body
  - Ordinary object
- Mental Object

Feature
- Relevant part
- Place

Occurrence
- State
- Process
- Accomplishment

Abstract

Figure 1: Onto Clean Top Categories.

1.2.1 Qualities and quality regions

'Quality' is often used as a synonymous of 'property', but this is not the case in the OCT ontology: qualities are particulars, properties are universals. According to our view, every entity comes with certain qualities, which exist exactly as long as the entity exists. These qualities belong to different *quality types* (like color, size, smell, etc.), and are characteristic (*inhere to*) specific individuals: no two particulars can have the same quality. So we distinguish between a quality (e.g., the color of a specific rose), and its "value" (e.g., a particular shade of red). The latter is called *quale*, and describes the "extension" (or "classification") of an individual quality with respect to a certain *conceptual space* (called here *quality space*) (Gärdenfors, P., 2000), So when we say that two roses have the same color their two colors are classified in the same way wrt the color space (they have the same *color quale*), but still they have two numerically distinct qualities.

The reason of this distinction between qualities and qualia, which is inspired to the theory of tropes (with some differences that can't be discussed here[11]), is mainly due to the fact that natural language – in certain constructs – seems often to make a similar distinction. For instance, when we say "the color of the rose turned from red to brown in one week" or "the room's temperature is increasing" we are not speaking of a certain shade of red, or a specific thermodynamic status, but of something else that changes its properties in time while keeping its identity. This is why we assume that qualities are endurants.

On the other hand, when we say that "red is opposite to green" or "red is close to brown" we are not speaking of qualities, but rather of regions within quality spaces. The specific shade of red of our rose – its color quale – is therefore an atom in the color space.[12]

[11] An important difference is that standard tropes theories explain a qualitative change in terms of a substitution of tropes (an old trope disappears and a new one is created). We assume instead that qualities are a sort of "enduring tropes".

[12] The possibility of talking of qualia as particulars rather than reified properties is another advantage of our approach.

Each quality type has an associated quality space with a specific structure. For example, lengths are usually associated to a metric linear space, and colors to a topological 2D space. The structure of these spaces reflects our perceptual and cognitive bias.
Under this approach, we can explain the relation existing between 'red' intended as an adjective (as in "this rose is red") and 'red' intended as a noun (as in "red is a color"): the rose is red because its color is located in the red region within the color space (more exactly, its color quale is a part of that region).
As a final remark, we note that qualities are assumed to be as specifically constantly dependent on the entities they *inhere to*.

1.2.1.1 Location

In the OCT ontology, space and time are considered as quality types like color, weight, etc. The spatial (temporal) individual quality of an entity is called *spatial* (*temporal*) *location*, while its quale is called *spatial* (*temporal*) *region* and it belongs to the associated quality space (respectively geometric space and temporal space). For example, the spatial location of a physical object is just one of its individual qualities: it belongs to the quality type *space*, and its quale is a region in the geometric space. Similarly for the temporal location of an occurence. This allows an homogeneous approach that remains neutral about the properties of the geometric/temporal space adopted (for instance, one may assume a circular time).
Notice that quality regions can have qualities themselves (for instance, the spatial location of a certain object can have a shape), in particular we assume that all quality regions are temporally located, and that their temporal qualia coincide with the temporal universe, i.e. quality regions are always present.

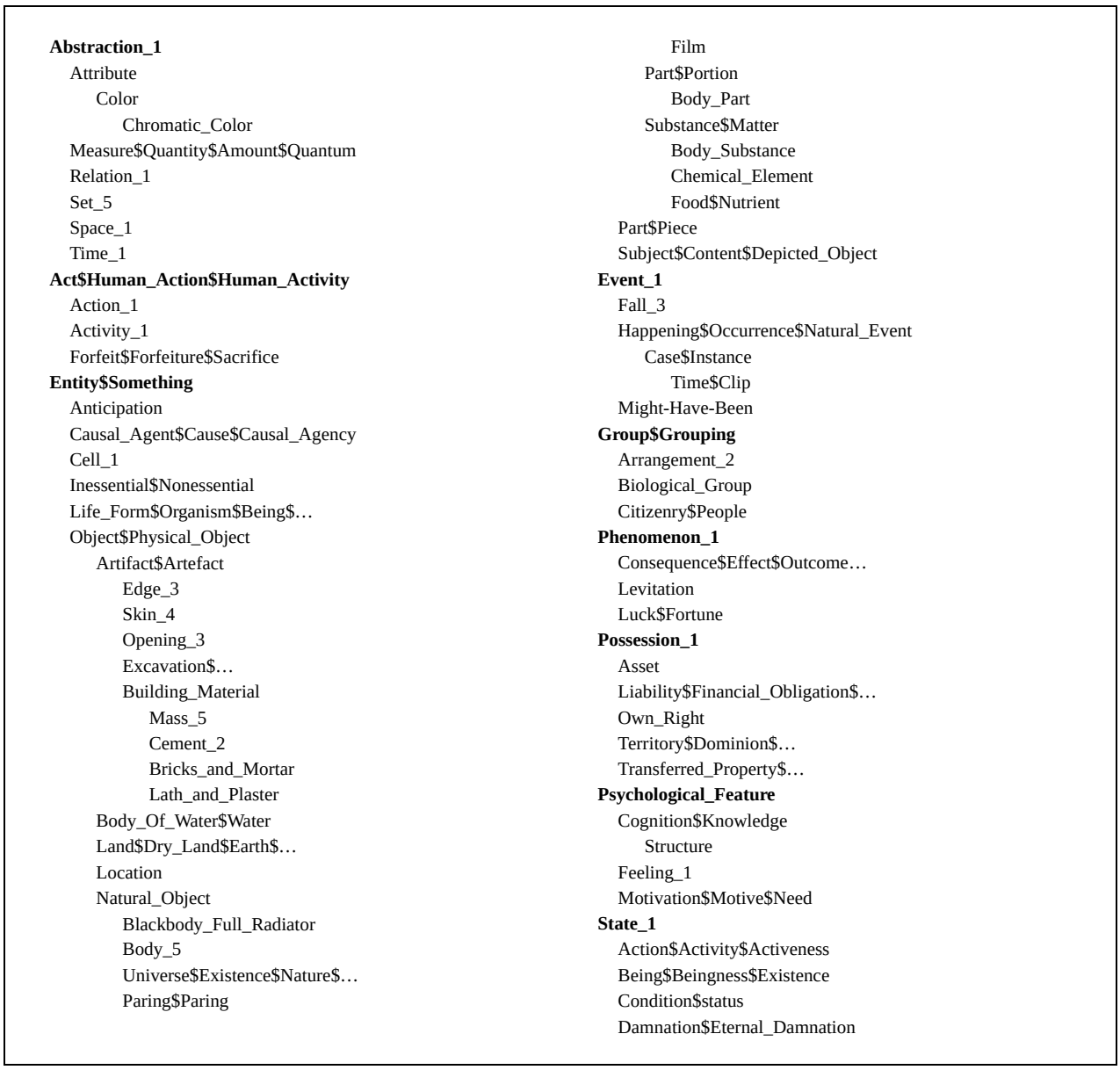


Figure 2: WordNet's top Level

1.2.2 Aggregates

The common trait of aggregates is that they are endurants and none of them is an essential whole. We consider two kinds of aggregates: *Amounts of matter* and *Arbitrary collections*. The former are mereologically invariant, in the sense that they change their identity when they change some parts. The latter are defined as "mere mereological sums" of essential wholes which are not themselves essential wholes (like the sum of a person's nose and a computer keyboard). They are essentially mereologically *pseudo-constant*, in the sense that they change their identity when a member (i.e. a special part of a collection, see (Gangemi, A. *et al.*, 2001)) is changed, while a change in the non essential parts of a member is allowed. We may have called arbitrary collections *groups*, or perhaps *sets*; but we prefer to use *set* for abstract entities, and *group* for something having an intrinsic unity.

1.2.3 Objects

The main characteristic of objects is that all of them are endurants and essential wholes. They have no *common* unity criterion, however, as different subtypes of objects may have different unity criteria. Often objects (indeed, all endurants) are considered ontologically independent from occurrences (discussed below). But, if we admit that every object has a life, it is hard to exclude a mutual ontological dependence between the two. Nevertheless, we can use the notion of dependence to distinguish between objects that are not specifically constantly dependent on other objects and have a spatial location (*physical objects*) and objects that are generically constantly dependent on *persons* (that are also objects) and do not have a spatial location (*mental objects*). Among physical objects, we further distinguish between *bodies* and *ordinary objects*. Bodies are mereologically invariant, and then they are material objects in the sense of physics.[13]. Ordinary objects (and mental objects even more) have a more cognitive nature, as they are admitted to change some of their parts while keeping their identity: they can have therefore *temporary parts*. Among mental objects, we could distinguish between purely *subjective mental objects*, i.e. objects depending on a singular person (like an intention, or a competence), and *intersubjective mental objects*, i.e. objects depending on a community of persons (like a project, a legal norm, a moral value, an aesthetic notion).

1.2.4 Features

Typical examples of features are "parasitic entities" such as holes, bumps, surfaces, or stains, which are generically constantly dependent on physical objects[14] (their *hosts*). All features are essential wholes, but no common unity criterion may exist for all of them. However, typical features have a topological unity, as they are *singular* entities. Features may be *relevant parts* of their host, like a bump or an edge, or *places* like a hole in a piece of cheese, the underneath of a table, the front of a house, which are not parts of their host.

[13] Notice that differently from the amounts of matter they are essential whole.

[14] We may think that features are specifically constantly dependent on their host, but an example like "a whirlpool" is very critical in this sense. Notice that we are not considering as features entities that are dependent on mental-objects.

1.2.5 Occurrences

Occurrences are synonymous of perdurants. They comprise what are variously called events, processes, happenings, and states. Occurrences can have temporal parts or spatial parts. For instance, the first movement of (the execution of) a symphony is a temporal part of it. On the other side, the play performed by the left side of the orchestra is a spatial part. In both cases, these parts are occurrences themselves. Clearly objects can't be parts of occurrences, rather they *participate* to them.

Within occurrences, we consider two main ontological dimensions of distinction: homeomery and relationality. The first dimension has been introduced by Parsons, Cresswell, and Mourelatos (see (Casati, R. & Varzi, A., 1996)): intuitively, we say that an occurrence is homeomeric iff all its temporal parts can be described *in the same way* used for the whole occurrence: for instance, every temporal part of "my sitting here" for an hour is still a "sitting here of mine". But if we consider "Messner's ascent to Everest" (intended in the complete sense), no parts of it are a "Messner's ascent to Everest". To formalize this notion, we need to refer to a certain property that holds for all the temporal parts of a certain occurrence *o*. We individuate this property by considering the most specific *occurrent* of *o*, i.e. the most specific occurrence type *o* is instance of. Then we can say that *o* is homeomeric iff all its temporal parts are instances of the same most specific occurrent.

The second dimension takes inspiration mainly from (Smith, B., 1982). An occurrence is said *non-relational* when only one object participates to it, while it is *relational* when it has two or more objects as participants. Occurrences involving qualities varying in time (i.e., which can change their qualia in time) are prototypical examples of non-relational occurences: the change of color of a rose has only one object as a participant (there may be other participants, such as the rose's color, but this is a quality and not an object).

In our proposal, homeomery seems to be enough to account for the distinctions proposed in the literature (especially (Mourelatos, A., 1996)) among *states*, *processes*, and *accomplishments*. It is easy to see that states are homeomeric occurrences (e.g., "the air smelling of jasmine"), while *accomplishments* are non-homeomeric (e.g. "the sunset"). Processes can be characterized as *weakly non-homeomeric*, in the sense that *some* temporal parts of them are instances of the same most specific *occurrent*, and some are not. For instance, in the case of "running", if you consider that instantaneous temporal part of your running through the park in which your right foot touches the ground while your left foot does not (think about photo-finish in a race), this sub-event is no more a "running". Together, processes and accomplishments are often described as *dynamic events*, just because of an (apparent) change of some of their properties across their different temporal parts.

In any case, we can further divide each of these categories into relational and non-relational occurrences.

1.2.6 Abstracts

Like mental-object and their qualities, abstracts are enduring entities that do not have a spatial location (indeed they do not have any "physical quality"). Differently from mental-object and their qualities, abstracts are independent from objects (and in particular from persons). Exam-

ples of abstracts are *sets, symbols, propositions, structures*, and *physical laws.*

4 Mapping WordNet into the OCT ontology

Let us consider now the results of integrating the WordNet top concepts into our top-level. According to the OntoClean methodology, we have concentrated first on the so-called *backbone taxonomy*, which only includes the rigid properties. Formal and material roles have been therefore excluded from this preliminary work.

Comparing WordNet's unique beginners with our ontological categories, it becomes evident that some notions are very heterogeneous: for example, Entity looks like a "catch-all" class containing concepts hardly classifiable elsewhere, like Anticipation, Imaginary_Place, Inessential, etc. Such synsets have only a few children and these have been already excluded in our analysis.

The results of our integration work are sketched in Table 2. Our categories are reported in the first column; the second column shows the WordNet synsets that are *covered* by such categories (i.e., they are either equivalent to or included by them); the third column shows some hyponyms of these synsets that were rejected according to our methodology. Finally, the last column shows further hyponyms that have been appended under our categories, coming from different places in WordNet. The problems encountered for each category are discussed below.

4.1 Aggregates, Objects, and Features

Entity is a very confused synset. As sketched in the table, a lot of its hyponyms have to be "rejected": in fact there are roles (Causal_Agent, Subject_4), unclear synsets (Location[15]) and so on. This Unique Beginner maps partly to our Aggregate and partly to our Object category. Some hyponyms of Physical_Object are mapped to our new top concept Feature.

By removing roles like Arrangement and Straggle, Group$grouping becomes a partition of the Ordinary Object category. In fact, hyponyms like Collection, Social_Group, Biological_Group, and so on, are nothing but plural objects, supporting a clear unity criterion.

Possession_1 is a role, and it includes both roles and types. In our opinion, the synsets marked as types (Asset, Liability, etc.) should be moved towards lower levels of the ontology, since their meanings seem to deal more with a specific domain - the economic one - than with a set of general concepts (except some concepts that can be mapped to Mental Object, such as Own_Right). This means that the remainder branch is also to be eliminated from the top level, because of its overall anti-rigidity (the peculiarity of roles).

4.2 Abstracts and Qualities

ABSTRACTION_1 is the most heterogeneous Unique Beginner: it contains abstracts such as Set_5, mental objects such as Chromatic_Color (an example of quality space[16]), qualities (mostly from the synset Attribute) and a hybrid concept (Relation_1) that contains mental objects, concrete entities (as Substance_4[17]), and even meta-level categories (see §2.2.2). Each child synset has been mapped appropriately.

Psychological_feature contains both mental objects (Cognition[18]) and events (Feeling_1). We consider Motivation as a material role, so to be added to lower levels of the taxonomy of mental objects.

The classification of qualities deals mainly with adjectives. This paper focuses on the WordNet database of nouns; nevertheless our treatment of qualities foreshadows a semantic organization of the database of adjectives too, which is a current desiderata in the WordNet community (see (Fellbaum, C., 1998), p. 66).

4.3 Occurences

Event_1, Phenomenon_1, State_1 and Act_1 are the Unique Beginners of those branches of WordNet denoting events. WordNet does not support the distinction between relational and non-relational occurrences, so first of all, in order to restructure this partition of the top level, we need to separate the hyponyms of the above-mentioned four synsets by means of our defined first dimension. We see, for example, that State_1 maps in part to non-relational state (condition$status, cognitive_state, existence, death_4, degree, skillfulness…), in part to relational state (medium_4, relationship_1 and relationship_2, disorder, order, hostility, conflict…). We register a similar behavior for the children of Process (a subclass of Phenomenon_1): decrement_2, increment and shaping could be seen as kinds of process involving a single main participant, while chelation, economic_process, execution and some hyponyms of Natural_Process (a direct hyponym of Process) seem to denote relational occurrences. Under Act_1 we find in general events of two kinds: processes (see activity_1 and its hyponyms) and accomplishments (see the homonymous synset under action_1). For sake of simplicity, we consider the hyponyms of Act_1 as being both relational and non-relational, depending on the context in which they are used. Event_1 has a too much generic composition in order to be partitioned clearly in terms of our approach (see, for instance, the beginning of §2.2.1): to a great extent, however, its hyponyms could be added to lower levels of the taxonomy of occurrences.

5 Conclusions

The final results of our integration effort are sketched in Figure 3. Our results show that a serious taxonomy rearrangement is needed. The blind application of OntoClean's taxonomy evaluation methodology provides a first guideline, but stronger ontological commitments seem to be unavoidable in order to get a "disciplined" taxonomy. In our opinion, strong (and explicit) ontological distinctions do also reduce the risk of classification mistakes in the ontology development process, and simplify the update and maintenance process.

Our research is still in progress: we hope we have paved

[15] Referring to Location, we find roles (There, Here, Home, Base, Whereabouts), instances (Earth), and geometric concepts like Line, Point, etc.).

[16] By looking to the corresponding hyponyms, it becomes clear that this synset could also be viewed as denoting a quality (by means of this we decide to append it both under Quality and Quality Region top concepts).

[17] "The stuff of which an object consists".

[18] "The psychological result of perception, and learning and reasoning".

the way for future work and possible cooperation.

6 Acknowledgements

We would like to thank Stefano Borgo and Luc Schneider for the fruitful discussions and comments on the earlier version of this paper. This work was jointly supported by the Eureka Project IKF (E!2235, Information and Knowledge Fusion), the IST Project 2001-33052 *WonderWeb* (Ontology Infrastructure for the Semantic Web) and the National project TICCA (Cognitive Technologies for Communication and Cooperation with Artificial Agents).

7 References

Casati, R. & Varzi, A. (eds.) (1996). Events. Aldershots, USA, Dartmouth.

Fellbaum, C. (ed.) (1998). WordNet - An Electronic Lexical Database. , MIT Press.

Gangemi, A. *et al.* (2001). Understanding top-level ontological distinctions: In Proceedings of IJCAI-01 Workshop on Ontologies and Information Sharing (26-33). Seattle, USA, AAAI Press.

Gangemi, A. *et al.* (2001). Conceptual Analysis of Lexical Taxonomies: The Case of WordNet Top-Level. In C. Welty & S. Barry (Eds.), Formal Ontology in Information Systems. Proceedings of FOIS2001 (285-296). , ACM Press.

Gärdenfors, P. (2000). Conceptual Spaces: the Geometry of Thought. Cambridge, Massachussetts, MIT Press.

Guarino, N. (1998). Some Ontological Principles for Designing Upper Level Lexical Resources. In A. Rubio *et al.* (Eds.), Proceedings of First International Conference on Language Resources and Evaluation (527-534). Granada, Spain, ELRA - European Language Resources Association.

Guarino, N. & Welty, C. (2000). A Formal Ontology of Properties. In R. Dieng & O. Corby (Eds.), Knowledge Engineering and Knowledge Management: Methods, Models and Tools. 12th International Conference, EKAW2000 (97-112). France, Springer Verlag.

Guarino, N. & Welty, C. (2002). Evaluating Ontological Decisions with OntoClean. Communications of the ACM, 45(2), (61-65).

Guarino, N. & Welty, C. (2002). Identity and subsumption. In R. Green *et al.* (Eds.), The Semantics of Relationships: an Interdisciplinary Perspective . , Kluwer (in press).

Loux, M. J. (1998). Metaphysics, a Contemporary Introduction. , Routledge.

Lowe, E. J. (1998). The possibility of metaphysics. Oxford, Clarendon Press.

MacGregor, R. M. (1991). Using a Description Classifier to Enhance Deductive Inference: In Proceedings of Seventh IEEE Conference on AI Applications (141-147).

Mourelatos, A. (1996). Events, Processes, States. In R. Casati & A. Varzi (Eds.), Events (457-476). Aldershot, Dartmouth Publishing Company.

Simons, P. (1987). Parts: a Study in Ontology. Oxford, Clarendon Press.

Simons, P. (2000). How to Exist at a Time When You Have No Temporal Parts. The Monist, 83(3), (419-436).

Smith, B. (ed.) (1982). Parts and Moments: Studies in Logic and Formal Ontology. München, Philosophia Verlag.

Strawson, P. F. (1959). Individuals. An Essay in Descriptive Metaphysics. London and New York, Routledge.

Thomasson, A. L. (1999). Fiction and Metaphysics. Cambridge, Cambridge University Press.

Varzi, A (2000). Foreword to the special issue on temporal parts. The Monist, 83(3).

Woods, W. A. & Schmolze, J. G. (1992). The KL-ONE family. In F. W. Lehmann (Ed.) Semantic Networks in Artificial Intelligence (133-177). Oxford, Pergamon Press.

OCT Top Categories	**Covered Synsets**	**Rejected Hyponyms**	**Imported Hyponyms**
Quality	Attribute*	Trait, Ethos, Inheritance, …	
Temporal Location	Time_interval$interval*	Eternity, Greenwich_Mean_Time, Present, Past, Future	
Spatial Location	Position$place		
Color …	Chromatic_color		
Quality Region	Attribute*	Trait, Ethos, Inheritance, …	
Time Region	Time_1, Time_interval$interval*	Eternity, Greenwich_Mean_Time, Present, Past, Future	
Space Region	Space_1*	Subspace, …	
Color Region …	Chromatic_color		
Aggregate	Aggregate_2 (!)		
Amount of Matter	Substance$Matter*	Bedding_Material, Ballast, Atom, …	Mass_5, Cement_2, Substance, …
Arbitrary Collection			
Object	ENTITY$SOMETHING*	Anticipation, Causal_Agent, Imaginary_Place, Substance	
Physical Object			
Body	Natural_Object*	Dead_Body, Constellation, Stone, Nest, …	
Ordinary Object	Physical_Object*, Group*	Finding, Catch, Vagabond; Arrangement, Social_Group, …	
Mental Object	PSYCHOLOGICAL_FEATURE*	Feeling_1, Motivation_1	Own_Right (!), Social_Group
Feature			
Relevant Part	Part$portion*, Fragment	Substance_4	Edge_3, Skin_4, Paring$Parings, …
Place			Opening_3, Excavation$hole_in_the_Ground, …
Occurrence	STATE_1*, PHENOMENON_1*, ACT*	Utopia, Dystopia, Nature, Consequence, Stay_1, …	
State	STATE_1*	Utopia, Dystopia, Nature	
Non-relational	Condition$status, Cognitive$State, Existence, Death_4, Degree, …		
Relational	Medium_4, Relationship_1, Relationship_2, Order, Disorder, Hostility, Conflict, …		
Process	Process, Activity_1		
Non-relational	Decrement_2, Increment, Shaping		
Relational	Chelation, Execution, …		
Accomplishment	Accomplishment$achievement		
Non-relational			
Relational			
Abstract			Statement_1, Cognition, Arrangement_2,
Proposition	Proposition_1		
Set …	Set_5		

Table 2: Synsets marked with '*****' are heterogeneous (some of their children are to be moved elsewhere, some are roles, or some are instances); those marked with '(**!**)' have no hyponyms; those in upper case are WordNet Unique Beginners.

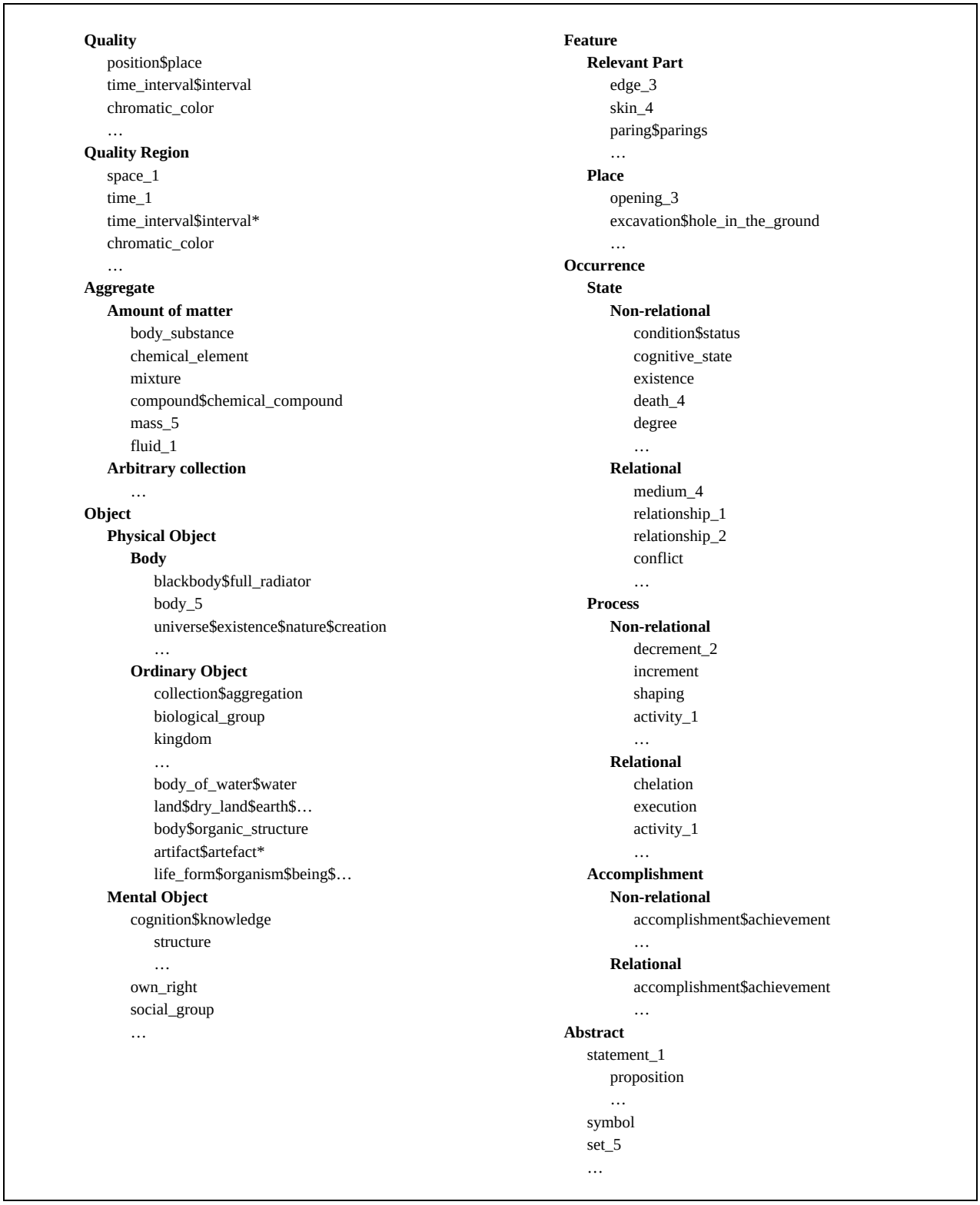


Figure 3: WordNet cleaned up: mapping WordNet into the OntoClean top-level.

Parallel Hierarchies in the Verb Lexicon

Christiane Fellbaum

Cognitive Science Laboratory, Department of Psychology
Princeton University, Princeton, NJ 08544, USA

Abstract

We discuss semantically heterogeneous manner-relations in the verb component of a lexical database. To make verb hierarchies more consistent while at the same time including instances of links among verbs that are based on expectancy instead of logical necessity, we propose to augment the lexical database with a parallel relation among hierarchically organized verbs. Possibilities for identifying instances of para-troponymy in corpora are outlined and the advantages of an enriched lexical database for NLP are briefly discussed.

1. Introduction and Background

It has been pointed out that the noun hierarchies in WordNet are built on heterogeneous subsumption relations (Gangemi et al., 2001; Gangemi et al., 2002; Guarino and Welty, 2001). The most common violation of the subsumption relation is the failure to distinguish Types and Roles (Guarino and Welty, 2002). Thus, WordNet lists as subordinates of the synset *dog, domestic_dog, Canis_familiaris* such synsets as *poodle, poodle_dog, Newfoundland*, and *corgi, Welsh_corgi* along with synsets like *cur, mongrel, mutt, lapdog, hunting_dog,* and *working_dog*. (Gangemi et al., 2001; Gangemi et al., 2002) propose eliminating from WordNet violations of strict subsumption (Type) relations and moving Roles like *student* to lower levels of the taxonomy.

Some of WordNet's verb hierarchies exhibit heterogeneous kinds of subordinates that seem intuitively similar to the Type/Role distinction among the nouns. For example, among the manner-subordinates of *clean*, we find *steam-clean* along with *brush, sweep*, and *wipe*. One of our goals here is to examine the heterogeneous manner-of relations in WordNet's verb component. Referring to work in progress, (Gangemi et al., 2002) briefly outline a clean ontology of events, categorizing them on the basis of criteria such as aspect and intentionality. Their examples are all complex events, such as *conducting a symphony* and *running a 100-meter race*. The number and nature of the event's participants as well as its spatial and temporal parts provide criteria for the ontological status of the events.

WordNet's verb entries are for the most part simple lexical items and do not include the kinds of complex events cited in (Gangemi et al., 2002). To the extent that WordNet is an ontology, it is a strictly lexical ontology whose entries are limited to concepts that are lexicalized in English[1]. WordNet resembles a traditional dictionary or thesaurus in that it does not explicitly account for aspectual or argument-taking properties of verbs (though verbs that are hierarchically related frequently share the same valency and aspectual properties). Therefore, the criteria for a clean ontology of events outlined by (Gangemi et al., 2002) are not applicable, and, indeed, may be complementary to the present discussion. Our treatment of simple verbs must necessarily be less ambitious, though we hope, no less interesting.

Besides offering some theroretical reflections, this paper attempts to outline how the different manner relations among the verbs could be constructively exploited and how corresponding links might be added to WordNet. Distinguishing and introducing a second manner relation parallel to the existing one would not only ensure semantically consistent relations but also yield a richer and more tighly interconnected network with a greater potential for NLP applications.

2. Hierarchies in WordNet's Verb Lexicon

In WordNet (Fellbaum, 1998), a word's meaning is represented by its membership in a group of cognitively synonymous words (a synset), and labelled pointers among the synsets that stand for semantic relations such as hyponymy, meronymy, and opposition.

The semantic relation that organizes most of the verbs in WordNet is the manner relation, or troponymy (Fellbaum, 1998). This relation allows one to build hierarchical structures akin to those found in the noun lexicon. Similar to the hyponymy relation expressable by the formula "X is a kind of Y", the formula for troponymically related verbs is (1):

(1) to X is to Y in some manner/way

For example, *stammer*, *lisp*, and *whisper* are among the many manner subordinates of *speak*, as the statement "to stammmer/lisp/whisper is to speak in some manner" shows.

Thus, WordNet expresses (part of) the meaning of verb X in terms of the meaning of its superordinate, Y. And the meaning of verb Y is expressed, in part, as the sum of the meaning of its subordinates (troponyms), such as X.

The manner relation is highly polysemous, as (Fellbaum, 1998) notes. Depending on the semantic domain, the differentiae distinguishing the superordinate from the more specific subordinate may be dimensions like speed (*walk-run*), direction (*move-rise*), volume (*talk-scream*), or intensity (*persuade-brainwash*). Despite these differences, the formula given in (1) seems to fit thousands of English verb senses and could be used to construct WordNet's extensive net, which currently includes well over 13,000 verb synsets.

3. Heterogenous Troponymy Relations

Most verbs fit neatly into a given hierarchy and can be assigned to a clearly identifiable superordinate (following

[1] WordNet's verb component contains a few non-lexicalized nodes that are arguably occupied by lexical gaps. See (Fellbaum and Kegl, 1989) for discussion.

an initial stage of identifying and coding top-level concepts, WordNet was constructed bottom-up). But if one examines specific hierarchies, it becomes clear that the relation is not just polysemous along the dimensions referred to above, but semantically heterogeneous.[2]

For example, *exercise* has subordinates like *jog*, *swim*, and *bike*. But these are clearly also manners of *moving/travelling*[3]. Both the following statements are true:

(2) to jog/swim/bike is to exercise in some manner

(3) to jog/swim/bike is to move in some manner

But clearly, there is a difference. The relation between *jog*, *swim*, *bike* and *exercise* is defeasible: Not every jogging/swimming/biking event is necessarily an exercising event. By contrast, every jogging/swimming/biking event is necessarily a moving event:

(4) She jogged/swam/biked but did not exercise

(5) *She jogged/swam/biked but did not move

The concept *exercise* is definable only by means of subordinates like *swim*, *jog*, and *bike* that are shared with another subordinate, *move*. But *move* has many subordinates that are not shared with *exercise*, such as *fly* and *drive*.

The relation of *jog*, *swim* and *bike* to their superordinates *move* and *exercise* is similar to that between, e.g., *dog*, *cat*, and *goldfish* to *animal* on the one hand and to *pet* on the other hand:

(6) A dog/cat/goldfish is a kind of pet.

(7) A dog/cat/goldfish is a kind of animal.

(8) That's my dog/cat/goldfish, but it is not a pet.

(9) *That's my dog/cat/goldfish, but it is not an animal.

Just as one can recognize dogs, cats, and goldfish as animals, but not (necessarily) as pets (Guarino, 1999), so one can recognize instances of biking, swimming, jogging as moving events, but not (necessarily) as exercising events. Unlike moving, the exercise component of biking, swimming, and jogging does not supply an identity criterion and is notionally dependent. Applying the terminology of (Guarino and Welty, 2001; Guarino and Welty, 2002) for nouns to verbs, we could say that *moving* is a rigid property, and *exercising* is an anti-rigid property of a biking/swimming/jogging event. Thus, verbs like *exercise* are similar to role nouns like *pet*, and *move* is similar to type nouns like *animal*.

[2]Some of the examples discussed here are not in fact coded in the current version WordNet, 1.7.

[3]For the sake simplification, we omit other nodes that may intervene; e.g., *jog* is linked to *move* via *run*.

3.1. Consequences for a Lexical Database

(Gangemi et al., 2002) propose an important criteria for "cleaning up" an ontology like WordNet: An anti-feature cannot subsume a feature. Thus, anti-rigidity cannot subsume rigidity. (Gangemi et al., 2002) advocate eliminating all violations of this principle found among WordNet's nouns. This would cut out hierarchical links between synset pairs like *animal* and *fictitious_animal*, while leaving intact the relation between pairs like *animal* and *horse*.

3.2. Arguments for Including Heterogeneous Troponymy Relations

The verb component of WordNet contains (perhaps many) cases of heterogeneous subsumption relations, and these must be recognized and distinguished. But we argue for retaining the corresponding pointers and, in fact, for coding more instances. Our arguments are grounded largely in a pragmatic view of WordNet as an NLP tool, rather than as an ontology that is perfectly consistent with strict logical principles.

First, if links between verbs like *bike* and *exercise* were eliminated in favor of links such as between *bike* and *move, travel*, important and potentially valuable information would be lost. In some cases, the semantic relation between words that are not conforming to strict subsumption principles is more salient than between words that are properly linked. This point will be discussed further later on.

Second, lexical databases that are useful for NLP gain from a tight network of relations. Word sense disambiguation, anaphor resolution, and applications relying on measures of textual cohesion can benefit from links such as between *bike* and *exercise*.

Finally, a random search in the WordNet shows up a fair number of subsumption violations of the *jog/swim/bike* as a manner of *exercise* kind. They are not simple lexicographic errors, as demonstrated by the goodness of the formula *to jog/bike/swim is to exercise in some manner*. But at present, we don't know how common such relations are, nor whether they are distributed evenly throughout the lexicon. Eliminating them when found would preclude a systematic study of the range, variety, and distribution of these relations and a better understanding of the structure of the lexicon.

4. Representing Different Kinds of Verb Hyponymy

Various possibilities exists for representing links between *bike, swim, jog* and superordinates like *move* on the one hand and *exercise* on the other hand.

First, each verb could be linked to multiple parents by means of the same labelled "manner" pointer. However, this "tangled hierarchy" approach is clearly unsatisfactory, as it implies that every jogging/swimming/biking event is both an exercising and a moving event, when in fact only the latter is true.

The second possibility is to posit two distinct senses each for verbs like *swim*, *bike* and *jog*, each sense with a different superordinate, here *move* and *exercise*. Some traditional dictionaries take this route; for exampe, *jog* is

represented in the *American Heritage Dictionary* as having distinct *running* and *exercising* senses. But this solution has the undesirable effect of increasing polysemy. More seriously, positing two distinct senses misses the fact that is every instance of jogging-as-exercise is necessarily also an instance of moving.

A better way to capture the relevant semantic facts is to introduce two distinct kinds of super-/subordinate relation linking a single verb to two superordinates. In addition to strict hyponymy, there would be a parallel hyponymy relation with the appropriate properties.

4.1. Para-troponymy

(Cruse, 1986) proposes a relation dubbed *para-hyponymy* for organizing nouns like *dog* and *pet* hierarchically. Like regular hyponymy, para-hyponymy admits the formula *Xs and other Ys*, where X is the subordinate and Y the superordinate: Both *roses and other flowers* and *dogs and other pets* are good. This formula can easily be adopted for verbs, and fits both strict hyponymy and para-hyponymy:

(10) Biking/swimming/jogging and other manners of moving/travelling

(11) Biking/swimming/jogging and other manners of exercising

To distinguish strict hyponymy from para-hyponymy among nouns, (Cruse, 1986) cites the *but*-test:

(12) It's a dog, but it's not a pet

This test shows that the hyponymy relation between *pet* and *dog* is first, expected, and second, defeasible.

Para-hyponymy can easily be applied to concepts expressed by verbs. The pairs *walk* and *exercise*, *jog* and *exercise*, *bike* and *exercise* etc. are all good in the *but not* frame:

(13) It's a walking/jogging/biking event but it's not an exercising event.

To distinguish this relation in the verb lexicon from para-hyponymy among nouns, we will call it para-troponymy. Our proposal for WordNet or a similar lexical database designed for NLP applications then is to include among the verb relations both strict troponymy and para-troponymy.

Other examples of verbs related by para-troponymy are listed below[4]. *Brush, wipe, sweep* are para-troponyms of *clean* and troponyms of *rub*; by contrast, *steam-clean, dry-clean* are strict troponyms of *clean*. *Nod, wink, scowl, frown, pout* are para-troponyms of *gesture*, *communicate* and troponyms of *move [a specific bodypart]* (omitting several intervening nodes).

[4]The examples of para-troponyms that we have found so far intuitively suggest a similarity to the telicity of Role nouns in para-hyponymic hierarchies; para-troponyms refer to events with a specific purpose or goal, as noted in (Fellbaum, 2002)

5. Expectation

(Cruse, 1986) notes that para-hyponymy is defined not by logical necessity but by "expectation." While intuitively convincing, this notion immediately raises several questions. How can expectation be characterized? Can it be quantified? How can pairs of verbs related by para-troponymy identified in the lexicon? And how do we know whether, say, a verb token *jog* in a corpus refers to an exercising event or (merely) to a running event?

To begin with, expectation is often context-dependent rather than inherent in the concept. In some contexts, a given verb's interpretation as a para-troponym is more salient, whereas in other context, its reading as a strict troponym of another superordinate is more appropriate:

For example, *move* is more salient in (14), but *exercise* is more salient in (15):

(14) a. The boat capsized and we had to swim to the shore.

(14) b. My car is in the repair shop so I'll bike to work.

(14) c. It started to rain heavily so she ran into the library.

(15) He swims/bikes/runs 3 miles every morning before work.

Some contexts allow for an underspecified reading:

(16) He jogged to the store.

More specifically, the nature of the verb's argument projection may play a role in setting up the expectation and the appropriate reading in some cases. *Clear dishes from the table*, where the Locatum entity is the direct object, seems to favor the *remove* reading (the strict superordinate) rather than the *clean* reading (the para-superordinate); *clear the table of dishes*, with the Location entity in direct object position, appears to favor the *clean* interpretation.

Second, the degree of expectation may differ across verbs independently of specific contexts. For some verbs, the para-relation is stronger than the strict relation, and the reverse may be true for other verbs. For example, *jog* intuitively is more strongly associated with its para-superordinate *exercise* than with its logical superordinate *run, move*. This is reflected in the fact that some dictionaries have distinct running and exercising senses for *jog*, as noted earlier. Conversely, *walk* seems be more strongly associated with *move* that with *exercise*. *Walk* seems like a less canonical form of exercise than *jog*, and thus exhibits a weaker association with its para-hypernym and a correspondingly stronger link to its strict superordinate.

The relative frequency of one reading as compared to another presumably influences expectation. Just as, say, hawks as pets may be more conventional in certain cultures than in others, there are probably cultures where jogging is not done for exercise purposes but, say, for pursuing game in a hunt.

Of course, the higher frequency of one reading as compared to the other makes the former more expected and thus stronger. It would therefore be desirable to firm up intuitions about the relative strength or weakness of the (para)troponymy relation with the aid of corpus data.

Almost any verb that is a hyponym of *move* could be made a para-troponym of *exercise*, just as any animal can be called a pet. If one wants to code para-relations in the database, it is important to avoid flooding it with links that reflect readings with very low expectancy. Here, too, corpus data would be useful to identify genuine from spurious para-links.

6. Para-tronomymy in the Lexicon

This paper has cited only a handful of examples of para-troponymy. At this point, we don't know how prevalent this relation is in the lexicon, or how many cases of concepts that exist merely by virtue of contingent subordinates are lexicalized in English. To find them, we need characteristic syntactic frames and a tool to search a corpus for appropriate occurrences of such patterns[5]. This section merely offers some thoughts and suggestions for future work.

We saw that para-troponyms pass the tests adapted from the one for para-hyponymy; in this respect, para-troponyms are indistinguishable from strict troponyms:

(17) X-ing and other manners/ways/methods of Y-ing.

(18) To X is to Y in some way/manner.

Using Google to search the Web for the string *and other manners/ways of*, we turned up quite a few examples of para-troponymy and para-hyponymy, as well as some cases of regular troponymy and noun hyponymy, in addition to cases of verbs co-occurring with nominalizations. Here are some cases of para-troponymy:

(19) Flirtation, courting and other manners of getting the attention of the opposite sex is certainly a form of manipulation ... www.mothersmagic.net/Goddess/maiden.html

(20) Befriending, listening and other ways of helping.... www.britishcouncil.org/sudan/science/ - 17k

(21) volunteering and other ways to help www.fcs-sf.org/page5.html

(22) Home Cooking and other ways to save Money. www.geocities.com/ dvsclothing/cooking.html

(23) Walking and other exercise use many muscles. www.lungusa.org/diseases/exercise.html

(24) activities that repeatedly flex the knee (ie, jumping, squatting, running and other exercise). orthoinfo.aaos.org/fact/thr_report.cfm? Thread_ID=252&topcategory=Knee

(25) Swimming, running, biking, walking and other exercise that are at a time length of over 20 minutes.. www.pmssolutions.com/Hiddentruth.html

To limit the search to para-troponyms, we searched for instances where the expected relation is negated, as in the pattern in (26):

(26) It's X-ing but not Y-ing
(e.g., it's swimming but not exercising)

We found:

(27) ...and then spraying the action with a little WD-40 is not cleaning. It is a slow methodical destruction of a considerable investment. Like everything ... www.doubleought.com/cleaning.html

(28) No, this is not "cleaning for the cleaning lady", it's picking up so that the cleaning lady can clean ... www.bitchypoo.com/2000/May/11.html - 7k

Similarly, on can search for cases where the para-hyponymy is asserted, possibly over a negative presupposition, as in the pattern in (29):

(29) This X-ing is Y-ing
(e.g., This swimming is exercising)

A web search turned up examples like these:

(30) Shotblasting is a way of cleaning or preparing surfaces for recoating, using an abrasive material forced through a jet nozzle... www.westshotblasting.co.uk/

(31) ... shake hands, using the right hand, and explain that this is a way of greeting one another. Pair up children and allow them to practice shaking hands. www.atozkidsstuff.com/math.html

(32) Tipping-leaving a gratuity-is a way of thanking people for their service. www.istudentcity.com/stages/3mannerstipping.asp

Another possibility is to examine co-occurrences of verbs in contexts for cases of (defeated) para-troponymy, without using any specific patterns. The following are actual examples:

(33) really get the job done. If the goal is to have clean sidewalks, they're going to have to be swept and bagged, not just blown. www.heartlight.org/two_minute/2min_971015.html

(34) will be swept by City crews. Residential streets are now swept once a month, while downtown streets are cleaned three times a week... www.ci.walnut-creek.ca.us/street

(35) These sociologists think that interrupting is a way of exercising power. They say, "Here we are dealing with a class of speakers ... www.glc.k12.ga.us/qstd-int/ancill/guidance/schoices/sc-f20.htm

We hope to develop more sophisticated and efficient ways for finding para-relations in the lexicon in the near future and to test their usefulness in applications.

[5]Resnik, Fellbaum, and Olsen are currently developing a tool to search the Web for specific syntactic patterns.

7. Summary and Conclusions

We have argued for retaining instances of paratroponymy in a lexical database like WordNet. Furthermore, we advocate collecting and adding naturally attested cases of this relation. Semantic relations that are not based on logical necessity but on expectations grounded in pragmatics or world knowledge are an interesting area for research in lexical semantics. Enriching a lexical database with para-relations can not only shed light on the organization of the lexicon, but may yield benefits for NLP applications relying on this database.

8. Acknowledgment

We thank Alessandro Oltramari for commenting on an earlier version of this paper. This work was supported by grant number IIS-ITR 0112429 from the National Science Foundation to the author.

9. References

Alan Cruse. 1986. *Lexical Semantics*. Cambridge University Press.

Christiane Fellbaum. 1998. *WordNet*. MIT Press.

Christiane Fellbaum. 2002. On the Semantics of Troponymy. In R. Green, S. Myang and C. Bean, editors, *Relations*, Dordrecht. Kluwer.

Christiane Fellbaum and Judy Kegl. 1989. Taxonomic Structure and Object Deletion in the English Verbal System. In: K. deJong and Y. No, editors, *Proceedings of the Sixth Eastern States Conference on Linguistics*, Columbus, OH: Ohio State University.

Nicola Guarino. 1998. Some Ontological principles for Designing Upper Level Lexical Resources. *First International Conference of Language Resources and Evaluation*, Granada, Spain.

Nicola Guarino. 1999. The Role of Identity Conditions in Ontology Design. *Proceedings of the IJCAI Workshop on Ontologies and Problem-Solving Methods*, 1–7, Stockholm.

Nicola Guarino and Chris Welty. 2002. Identity and Subsumption. *LADSEP-CNR Internal Report*, Padua, Italy.

Nicola Guarino and Chris Welty. 2002. Evaluating Ontological Decisions with Ontoclean. *Communications of the ACM*, 45.2:61-65.

Aldo Gangemi, Nicola Guarino, and Alessandro Oltramari. 2001. Conceptual Analysis of Lexical Taxonomies: The Case of WordNet Top Level. *Proceedings of FOIS*, Ogunquit, Maine.

Aldo Gangemi, Nicola Guarino, Alessandro Oltomari, and Stefano Borgo. 2002. Cleaning up WordNet's Top Level. In U. N. Singh, editor, *Proceedings of the First Global WordNet Conference*, Mysore, India. CIIL.

On the Ontological Basis for Logical Metonomy

Telic Roles and WORDNET

Sandiway Fong

NEC Research Institute
4 Independence Way
Princeton NJ
sandiway@research.nj.nec.com

Abstract

The analysis of examples of Logical Metonomy, where an event-taking verb is combined a non-eventive object, intuitively involves the recovery or insertion of a missing verb generally known as a Telic Role. For example, for *Mary enjoyed the meal*, an appropriate might be *eat*, i.e. *Mary enjoyed eating the meal*. The question for lexical semantics is where do telic roles reside and how are they accessed? In this paper, we investigate the use of WORDNET, a widely used semantic network, both as an appropriate repository and also as an organization suitable for the recovery or assignment of telic roles.

1. Introduction

The interaction of aspectual verbs such as *begin* or *finish* with simple, non-eventive noun phrases (NPs) has been used to motivate an account of *logical metonymy* in which telic (purpose/function) and agentive (creation) roles are distinguished components of the lexicon, see (Pustejovsky, 1995). Others, e.g. (Lascarides and Copestake, 1995) and (Verspoor, 1997), have highlighted the role of context and convention. Consider (1).

(1) a. John began the novel (*reading*/*writing*)
b. The author began the unfinished novel back in 1962 (*writing*)

(1a) can mean *John began reading the novel*, accessing the functional sense or telic role of *novel*, or *John began writing the novel*, accessing the specific means of creation or agentive role of *novel*. The telic/agentive role ambiguity seen in (1a) can be made less apparent in context, either within the same sentence, as in (1b) above, or through discourse or semantic inference, as in (11) and (12), to be discussed below. Note that there are important constraints, e.g. with respect to boundedness and aspect, on the possible NPs that can appear with *begin*. See (Verspoor, 1997) and the references cited therein for discussion of the relevant factors.

Other verbs such as the subject-experiencer psych verb *enjoy*, or verbs such as *refuse*, exclude the agentive role.[1] For example, contrast (2a) with (1a).

(2) a. Mary enjoyed the novel (*reading*)
b. Timmy refused the meal (*to eat*)

In (2b), *refuse* can access the telic role for *meal*, namely *to eat*. However, there is room for ambiguity here; (2b) is also compatible with the interpretation *Timmy refused* to accept *the meal*, cf. (3) below.

(3) Timmy refused the present (*accept*)

In (3), arguably the telic role of *present*, meaning *gift*, is *accept*. However, the same account cannot be posited for *meal*; its basic function (if one exists) is to be consumed or eaten; thus creating a problem for enumeration in lexical representation. In other cases, such as (4), there is no (felicitous) telic role at all.

(4) a. !John enjoyed the rock
b. !!John enjoyed the door

A physical object like *rock* has no obvious function. Yet (4a) can be marginally interpreted in the context that some (physical) aspect of the object gave *John* pleasure, e.g. its appearance as in *John enjoyed looking at the rock*. Or we can appeal to other perceptual properties, e.g. the tactile sense as in *the blind man enjoyed* touching *the rock*. To take one more example, consider (5):

(5) Mary enjoyed the garden

The prototypical definition of a garden as a pleasing arrangement of plants and other natural (or non-natural) objects admits not only the (putative) telic role *to see* but also a range of other possibilities, illustrated in (6).

(6) a. Mary enjoyed *seeing* the garden
b. Mary enjoyed *inspecting* the garden
c. Mary enjoyed *visiting* the garden
d. Mary enjoyed *strolling* through the garden
e. Mary enjoyed *rollerblading* in the garden
f. Mary enjoyed *sitting* in the garden
g. Mary enjoyed *dozing* in the garden

The ease of defeasibility of telic roles and the productivity of plausible alternatives is striking. In general, the recovery of appropriate contextual function falls outside the domain of local or specific lexical knowledge. It belongs more appropriately to systems that carry out reasoning and inference about the real world.

In fact, the recovery of contextual function is more ideally suited to ontological networks, which encode general semantic relations between abstract and concrete concepts

[1] *Enjoy* can take *write* explicitly, as in *Mary enjoyed writing the novel*. But this is not an instance of what Pustejovsky terms "coercion".

in the real world. This paper explores the application of such a network, WORDNET, to this problem. In particular, we will make use of the *isa*, or hypernymy, relation on, assuming (as required) the existence of certain commonsense, or real-world, properties of higher-level concepts, to account for a range of data.[2]

2. Hypernymy

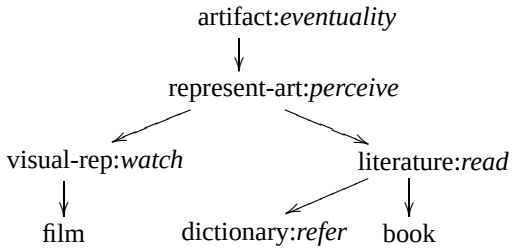
The idea that hypernymy may inform interpretation in logical metonymy has already been hinted at, or tacitly assumed, in several places in the literature. For example, this is apparent from the summary of logical metonymy in the BNC corpus, (Verspoor, 1997), excerpted in (7):

(7) eat FOOD/MEAL
drink LIQUID
tell STORY
play MUSIC
read/write WRITTEN_OBJECT
take MEDICINE/TREATMENT

(The capitalized terms in (7) denote semantically relevant concepts.)

(Lascarides and Copestake, 1995) assume the following telic roles for artifacts:

(8)



Finally, (Asher and Pustejovsky, forthcoming) assert the following complex types ($\otimes$ a type constructor):

(9) a. $p \otimes see$ and $p \otimes hear$ to encode the fact that objects with extension are typically visible, and objects involving sound are typically audible, respectively.
b. all artifacts inherit a general dependent type that gives their cause.
c. *wine*: liquid $\otimes_T$ drink ($\otimes_T$ introduces the telic role)
d. *class*: people $\otimes_T$ teach

In this paper, we explicitly test the hypothesis using the somewhat coarse-grained *isa*-relation available in WORDNET.[3] In conjunction with two principles, specificity and locality, defined with respect to hypernymy, we explain why telic/agentive roles are available for some cases but not for others. If this is the case, the locus of variation should be in ontological not lexical structure (as suggested by lexical entries such as the following):

(10) *novel*(y): telic:λx.*read*(x,y) agentive:λx.*write*(x,y)

In fact, in generative grammar, the lexicon is generally taken to be a repository of exceptions, see (Chomsky, 1965) citing Bloomfield. In this framework, non-idiosyncratic properties are factored out into grammar or further afield. Obviously, the evaluation of properties implicating mechanisms peculiar to language must stay within the domain of the language faculty. Non-language particular properties are perhaps best assimilated to general systems of reasoning and cognition.

Ontological relationships play a large role in lexical semantics and, more generally, semantic inference. Any account of language phenomenon involving the interaction of lexical entries with inheritance and (semantic) class-based behavior falls into this category. Computation involving defeasible reasoning and knowledge about the physical properties of objects in the real world should therefore fall outside the scope of the lexicon.

Furthermore, as (11), from (Lascarides and Copestake, 1995), illustrates, telic roles are easily overridden through discourse priming:

(11) a. He really enjoyed your book (*reading*)
b. My goat eats anything. He really enjoyed your book (*eating*)

Even in cases where arguably no felicitous telic role exists to be overridden, as in (12a), discourse may play a part in supplying the missing event, as in (12b):

(12) a. !He enjoyed your shoe[4]
b. My dog eats everything. He really enjoyed your shoe (*eating*)

3. The WORDNET Framework

3.1. The Hypernym Hierarchy

In WORDNET, nouns are grouped into synonym sets, known as "synsets", representing single concepts. For example, *cigarette*, *coffin nail*, *butt* and *fag* are generally substitutable, and thus belong to the same synset. Concepts are related through (possibly iterated) application of the hypernymy ("$\rightarrow$") or *isa*-relation, illustrated in (13).[5] Inheritance is strictly unidirectional in this model. For example, *tobacco* may be termed a *street drug*, but the reverse need not be true. Furthermore, multiple inheritance may obtain for some concepts. For example, *tobacco* is a *plant product* as well as a *street drug*.

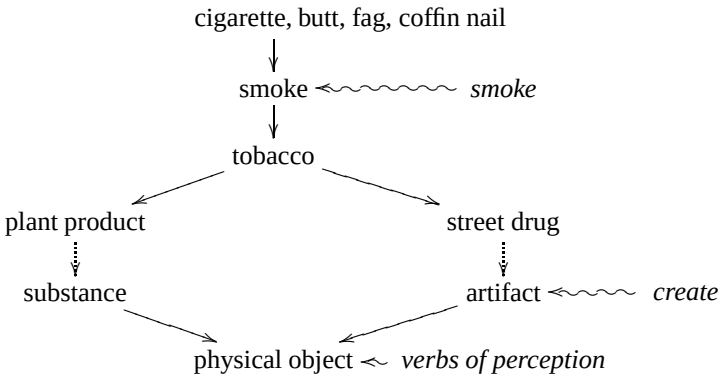
[2]The idea of using WORDNET on object NPs to pick out contexts in which those NPs represent events on a class-based model is not new. (Siegal, 1998) performed a (medical) corpus study in conjunction with WORDNET to distinguish eventive and stative *have*, e.g. *the patient had a fever* (stative)/*blood loss* (eventive).

[3]As (Gangemi et al., 2001), have noted, WORDNET's hypernymy relation is a heterogeneous one, merging functional and non-functional *isa*-relations alike, e.g. isa(tobacco,plant_product) and isa(tobacco,street_drug).

[4]In the framework described in this paper, *shoe* is a "foot covering". The telic role is cover(NP,FOOT), which is incompatible with prototype V(PRO,NP) defined in section 3.. The next higher concept is "footwear" with telic role *wear*, perhaps accessed in contexts like *He enjoyed the comfortable shoes you lent him*.

[5]For brevity, a dotted arrow ("$\dashrightarrow$") will sometimes be used to represent a hypernym sequence.

(13)

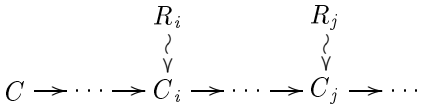


In this paper, we will assume annotation of concepts with characteristic verbs where relevant (to be indicated by "$\lesssim$"). For example, in (13) *artifact*, defined in the gloss as a "man-made object", is associated with the verb *create*. Similarly, the noun *smoke* is associated with the related verb *to smoke*.[6] Finally, the concept *physical object*, defined as "a tangible and visible entity", is characterized by verbs of perception such as *see*/*look at* and *touch*.

3.2. Contextual Function Search Rules

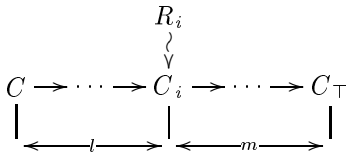
In this paper, we employ two simple principles of contextual function search over the hierarchy outlined above. In the following section, concepts will be denoted by (subscripted) C. R_i will denote a characteristic verb for a concept C_i. Given a noun $N \in C$, we have the rule of preference (14):

(14) **Principle of Specificity**: Prefer R_i to R_j in the sequence



In other words, prefer a closer role R_i over a more general one R_j in the concept chain. The (one-way) hypernymy relation relates a specific concept to a more general concept, so the closer a matching concept is in terms of the number of links, the more specific it will be. Next, given a noun $N \in C$ and $C_\top$ representing the top or most general concept relative to N, we have the rule of evaluation of the "goodness" of a characteristic verb R_i (15):[7]

(15) **Principle of Locality**: Plausibility of R_i scales with m and inversely with l in



Scalars l and m represent the length of sequences $< C, \ldots, C_i >$ and $< C_i, \ldots, C_\top >$, respectively. The closer C_i is to C (l small), the more plausible R_i will be. On the other hand, if C_i is close to $C_\top$, m will be small, encoding the intution that R_i (then) is a general characteristic that is not strongly associated with specific concept C. Rules (14) and (15) operate in tandem. Although the closest concept is always preferred, *ceteris paribus*, it will be deemed implausible or requiring of strong contextual support if it is many links from C or close to $C_\top$.

3.3. Grammatical Constraints

In what follows, we will consider the problem of determining the value of the verb V in the configuration (16b) given (16a), a restricted version of the telic role determination problem.

(16) a. EXP enjoy NP
b. EXP_i enjoy [PRO_i [V(ing) NP]]

In (16), EXP is the experiencer subject of *enjoy*, NP the object, PRO the controlled subject of V, and V a transitive verb V(PRO,NP). The twin requirements that the NP as must be the embedded object and that the subject be controlled limits the possibilities for telic roles to appear as V, as will be seen in the next section.

4. Worked Examples

Cigarette: Consider (17).

(17) Mary enjoyed the cigarette (*smoking*)

Given the hypernym hierarchy in (13), *smoke*(PRO,*cigarette*) is the strongly preferred interpretation since the concept *smoke* is highly specific (l small) and distant from general concepts *artifact* and *physical object* (m large).

Sonata: Consider the possibilities in (18).

(18) a. Mary enjoyed the sonata (*listening to*/*playing*)

[6] Concepts in WORDNET have associated glosses. A gloss will typically contain a brief definition and examples of use. In some cases, the characteristic verbs can be inferred from the gloss or from members of the synset. Further exploration of this idea is beyond the scope of this paper.

[7] In WORDNET's hypernym hierarchy there is no unique $C_\top$ concept. For example, *dirt* as *material* and as *gossip* have top concepts *entity* and *act*, respectively. See (34).

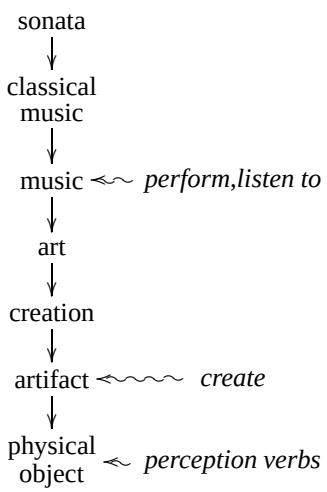
b. Mary began the sonata (*playing*/*composing*)

According to (Asher and Pustejovsky, forthcoming), the agentive and telic roles associated with *sonata* are *compose* and *play*, expressed in their type logic notation as (19).

(19) sonata: $(p \bullet i) \otimes_{A,T} (\text{compose}, \text{play})$

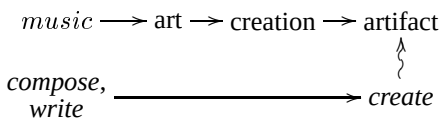
The hierarchy for *sonata* is given in (20).[8]

(20)



(20) predicts that *perform* and *listen to* are preferred in (18a). Verbs *begin* and *enjoy* differ in that *begin* allows an agentive role. This excludes subject-experiencer *listen to* but allows for *perform* and is also compatible with *create*. Note that *create* is associated with the general concept *artifact*. We can turn to WORDNET's verb hierarchy, shown superimposed in (21), to pick out the music-specific sense of *compose*.[9]

(21)

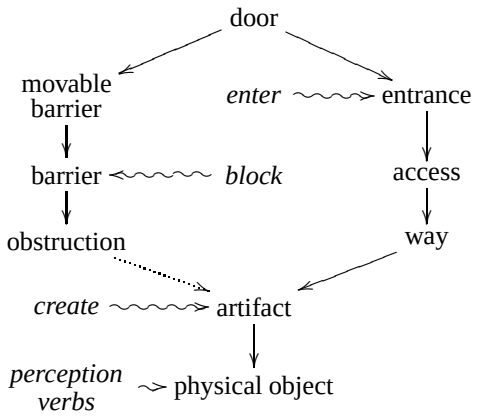


(18b) is explained since *compose* (or *write*) and *perform* are effectively equidistant from *sonata*.

Door: Consider (4b), repeated here as (22), with WORDNET hierarchy (23).

(22) !!John enjoyed the door

(23)

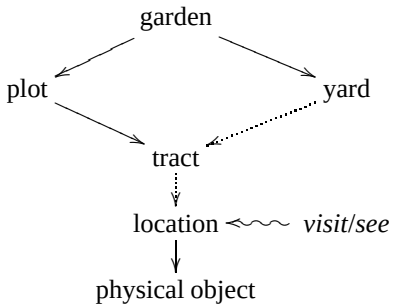


Specifically, a door can function both as an entrance (*enter*) and a barrier (*block*) to an enclosure. However, the telic verb *block* has form *block*(*door*,ENCLOSURE), which is incompatible with the prototype V(PRO,*door*), thus ruling out *block*. Similar reasoning applies to *enter*(PRO,ENCLOSURE). At the other end of the hierarchy, the canonical events associated with *physical object* are predicted to be implausible (l large, m small).

Garden: Consider (5), repeated here as (24), with WORDNET hierarchy (25).

(24) Mary enjoyed the garden (*seeing*/*visiting*)

(25)

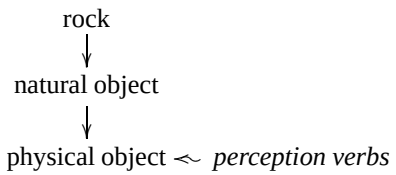


Assuming *visit* and visibility are characteristic of locations in general, (24) is accounted for. General mechanisms involved in reasoning about entailment may also play a large role in grounding *visit*. Note that the possibilities exemplified in (6) all entail *visit*.

Rock: Consider (4a), repeated here as (26).

(26) !John enjoyed the rock

(27)



Unlike *door* in (22), *rock* has no obvious function, as the simple hierarchy in (27) suggests. Hence, relatively speaking, we predict that (26), when picking out perceptual *looking at* or *touching*, is more acceptable than (22) (since l is smaller). However, the value of m is still small, indicating its acceptability can be improved significantly by contextual (discourse) support.

[8] Note, *physical object* $\rightarrow$ *entity* in WORDNET. $C_\top$ = *entity* has been omitted in (20) since *entity* has no possible characteristic functions.

[9] *Compose* and *write* belong to the same synset glossed as "*write music*". Thus the gloss locates this synset with the concept *music*.

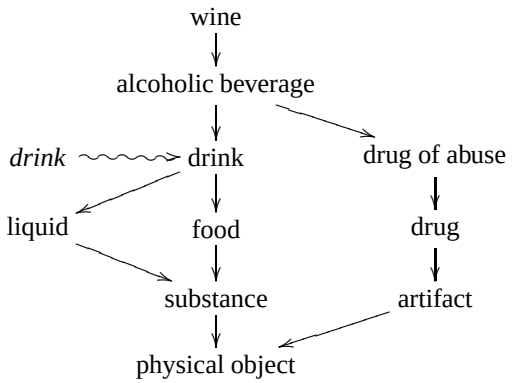
Note that WORDNET does not classify *rock* as a location, cf. *garden* in (5). Given the right context, the characteristic function *visit* may also be felicitous for *rock*, as in (28), where the rock in question is geographically significant.

(28) Mary enjoyed Ayer's Rock (*visiting*)

Wine: Consider (29) with hierarchy (30).

(29) Mary enjoyed the wine (*drinking*)

(30)

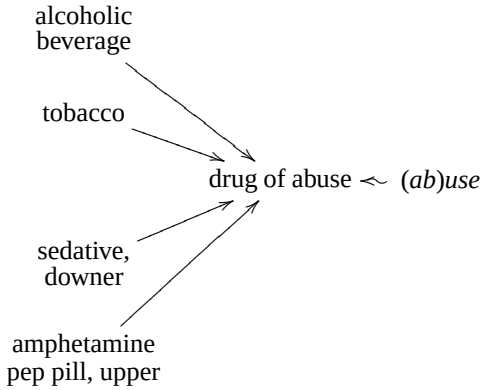


(30) strongly predicts (29) (l small, m large). However, this assumes the branch containing *drug of abuse* (with telic role (*ab*)*use*) is marginalized, i.e. *wine* as *drink* is preferred over *drug of abuse*. Contrast (29) with (31).

(31) Mary enjoyed the amphetamine/sedative (*using*)

(31) is also strongly predicted in our analysis as the elaborated WORDNET hierarchy fragment in (32) illustrates.

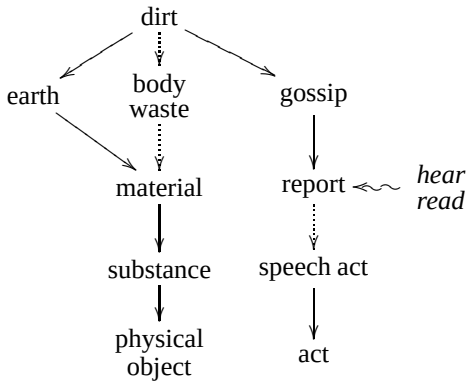
(32)



Dirt: Consider (33) with hierarchy (34).

(33) !John enjoyed the dirt

(34)



In (33), *dirt* as a natural substance has no plausible telic role. The corresponding WORDNET hierarchy is shown in (34). The relevant sense is given by the sequence <*dirt,earth,material,substance*>; the elements of which have no obvious purpose or function. Hence the status of (33).

According to WORDNET, *dirt* is also, perhaps little used, slang for fecal matter. Other (more common) words sharing the same synset are *crap*, *shit*, *poop* and *turd*. The telic role for *body waste*, perhaps *discharge*, is generally available for the synset, as can be seen by substitution of *dirt* in (33). So an appropriately annotated WORDNET makes essentially the right prediction for the synset as a whole. Finally, the right prediction is also made for *dirt* in the sense of malicious gossip, as in (35).

(35) John enjoyed the dirt on OJ Simpson (*hearing about*/*reading about*)

5. Conclusions

In this paper, we have argued for an ontological approach to the problem of logical metonymy using WORDNET's hypernymy relation for non-eventive nominals. That is, we interpret logical metonymy to be a phenomenon belonging to systems of semantic interpretation and general reasoning, governed by simple rules of specificity and locality with respect to concept hierarchy. We have shown, through worked examples, how such a mechanism accounts for data of the sort commonly cited in the literature.

Interesting questions remain for future work. For example, not all concepts in the WORDNET hierarchy have simple lexical realization satisfying the grammatical constraints, the question of what happens with lexical gaps remains. Since languages vary with respect to concept lexicalization, the question of whether the results obtained here generalize to other languages exhibiting logical metonomy remains open.

6. References

N. Asher and J. Pustejovsky. (forthcoming). The metaphysics of words in context. *Journal of Logic, Language and Information*.

N.A. Chomsky. 1965. *Aspects of the Theory of Syntax*. MIT Press.

A. Gangemi, N. Guarino, and A. Oltramari. 2001. Conceptual analysis of lexical taxonomies: The case of wordnet top-level. In *Proceedings of FOIS 2001*.

A. Lascarides and A. Copestake. 1995. Pragmatics of word meaning. In *Semantics and Linguistic Theory (SALT5)*, Austin, Texas.

J. Pustejovsky. 1995. *The Generative Lexicon*. MIT Press.

E. V. Siegal. 1998. Disambiguating verbs with the wordnet category of the direct object. In *Workshop on Usage of WordNet in Natural Language Processing Systems*, University of Montreal, Montreal, Canada.

C. M. Verspoor. 1997. Conventionality-governed logical metonymy. In H. Bunt, L. Kievit, R. Muskens, and M. Verlinden, editors, *2nd International Workshop on Computational Semantics*, pages 302–312, Tilburg, Netherlands.

Differentiae Specificae in EuroWordNet and SIM*u*LLDA

Maarten Janssen

UiL-OTS, Utrecht University
Trans 10, 3512 ED Utrecht, The Netherlands
m.janssen@let.uu.nl

Abstract

(Euro)WordNet, like all other semantic network based formalisms, does not contain differentiae specificae. In this article, I will argue that this lack of differentiae specificae leads to a number of unsurmountable problems, not only from a monolingual point of view, but also in a multilingual setting. As an alternative, I will present the framework proposed in my thesis: SIM*u*LLDA. The SIM*u*LLDA set-up not just contains differentiae specificae (called definitional attributes), but differentiae specificae form the building blocks of the system: the relations between meanings are derived from the application of Formal Concept Analysis to the set of definitional attributes.

1. Introduction

Given the many shortcomings of systems based on semantic primitives, WordNet, like many other lexical databases and knowledge bases, is based on semantic networks (see for instance Miller (**?**)). In semantic networks, there is no need for anything like semantic markers or, as you would call them from a lexicographers point of view, differentiae specificae, since all information is formulated in terms of relations between (in the case of WordNet) synsets. In this article, I will argue that this lack of differentiae specificae leads to a number of insurmountable problems, not only from a monolingual point of view, but also in a multilingual setting.

As an alternative, I will present the framework proposed in my thesis (**?**): SIM*u*LLDA, a Structured Interlingua MultiLingual Lexical Database Application. The SIM*u*LLDA set-up not just contains differentiae specificae (which are called definitional attributes in the system), but differentiae specificae form the building blocks of the system: the relations between meanings are derived from the application of a logical formalism called Formal Concept Analysis (FCA) to the set of definitional attributes.

After the presentation of the framework, I will indicate why definitional attributes do not give these traditional problems by showing that the resulting framework should not be viewed as an ontological hierarchy, nor as a knowledge base, but as a modest lexical database.

In this article, the following notational conventions will be used: meaning-units, in the case of WordNet the synsets, will be typeset in SMALL-CAPS, word-forms are set in sans serif, differentiae specificae, as well as the relations in WordNet, in **bold-face**.

2. The Need for Differentiae Specificae

One of the main aspects of the WordNet system is its ontological hierarchy, provided by the **is_a** links. Although not de facto a separate system (the **is_a** link is just a link as any other), the hierarchy is often presented that way, and many applications of the WordNet database only make use of this ontology. So for the moment I will consider the (ontological) hierarchy of WordNet as a system on its own.

The **is_a** relation links a synset to its *genus proximum* (to use the lexicographer's term), hence strongly characterising the meaning of the synset by indicating what kind of meaning it is. But on its own, the **is_a** link does not fully characterise the meaning of the synset: it fails to distinguish the various hyponyms of the same synset. From the point of view of the hierarchy we also need *differentiae specificae* to keep the meanings/synsets within the same genus apart.

In the WordNet approach, this differentiation is done by means of the other links. As an example, one could define the synset ACTRESS by means of an **is_a** relation to ACTOR, and a **female** relation the other way around (or alternatively a **is** relation to FEMALE). But although the other links in WordNet do provide additional information about the synset, they are not designed to provide differentiae specificae. This shows in two ways: firstly, the other links give information independent of the **is_a** link, which means that they are independent of the information already provided by the **is_a** link. So they cannot structurally supplement the information lacking from the **is_a** link.

Secondly, not all differentiating information can be modelled by means of these other links. Consider for instance the word *millpond*, which **is_a** AREA OF WATER. But a millpond is not just any area of water, it is specifically one *used for driving the wheel of a watermill* (according to LDOCE). And there are no WordNet links for this type of differentiating information.

So differentiae specificae as such do not exist in WordNet, even though in some (or many) cases the differentiating information will be present or can be provided somehow. This absence of a structural modelling of differentiae specificae leads to serious problems. Let me illustrate this using three examples.

The first example is that, according to Vossen & Copestake (**?**), (Euro)WordNet has problems dealing with verb nominalisations: SMOKER is a hyponym of PERSON, but so are RUNNER, SLEEPER, JOGGER, etc. The point here is not so much that distinguishing these nominalisations is impossible in WordNet: in principle, these can be distinguished by means of the **involved_agent** relation. So we can express that the involved agent for SMOKE is SMOKER, and hence by means of backward search say that a smoker is a person *who smokes*. The point is that for synsets with large numbers of hyponyms, there is no structural way of telling them apart: WordNet in many cases depends on the ontological hierarchy, so the less layered it is, the less informative it is.

The second example makes a similar point: because of

the high dependence on hierarchy, WordNet is forced to accept as layered a structure as possible: to indicate the relation between ENEMY and MURDERER, WordNet has to introduce a synset for BAD PERSON, even though there are no words related to that synset. This introduction of 'empty synsets' is not really incorrect, but at least conceptually unattractive.

The lack of differentiae specificae is most disturbing when considered in a multilingual setting. As a third example, consider the Spanish word DEDO. It is a (translational) hyperonym of both the English FINGER, and the English TOE, since a finger is a *dedo del mano*, and a toe is a *dedo del pie*. The way this is modelled in EuroWordNet is as follows: the Spanish DEDO has an **eq_synonym** relation to an InterLingual Item (ILI) DEDO, and both the English FINGER and TOE are related to this same ILI with a relation **eq_has_hyperonym**[1]. In this way, the words finger and toe are correctly modelled as translational hyponyms of dedo.

But in this cross-linguistic linking, there is nothing keeping the two translational hyponyms *finger* and *toe* apart. That is to say, language internally, FINGER will have a **part_of** relation to HAND, and TOE to FOOT, but this information is not (directly) related to the cross-linguistic link to DEDO. Furthermore, if we would use these **part_of** relations to tell the translational hyponyms apart, they would be used as differentiae specificae. And there are other examples in which such differentiae specificae are not available. For instance, the French BIEF will be linked as a translational hyponym of CANAL, but the reason why bief is more specific (namely that it is a canal *bringing water from a stream to a hydraulic installation*) would not be modelled, because WordNet has no links to provide for it.

Such examples show that in a lexical database, there is a definite need for a structural modelling of differentiae specificae, especially in a multilingual setting. Although in this section, the criticism is specifically aimed at (Euro)WordNet, any hierarchy based system without a structural modelling of differentiae specificae will encounter the same problem, though they might show up in a different guise. Let me now turn to the system proposed in my thesis which does use differentiae specificae.

3. SIM*u*LLDA

In my thesis, I describe a multilingual lexical database set-up called SIM*u*LLDA, in which differentiae specificae play a crucial role. The differentiae specificae are modelled within the system by means of entities called *definitional attributes*. The SIM*u*LLDA system is designed to be a multilingual lexical database system from which bilingual definitions between arbitrary pairs of languages in the system can be derived.

The SIM*u*LLDA set-up consists of a number of steps: the data from monolingual dictionaries are reduced to sets of definitional attributes. These sets of definitional attributes are turned into a lattice structure by means of a logical formalism called Formal Concept Analysis (FCA). The result is a lattice structure, which can serve as a structured interlingua, connecting words from different languages. Let me show how this works using a simple example: the words for horses in English. This explanation is very brief; for a more complete explanation I refer to my thesis (**?**).

3.1. Creating Sets of Definitional Attributes

The hierarchical set-up of the SIM*u*LLDA system is best shown using a small and simple lexical field, such as the words for male, female, young, and adult horses in English. The SIM*u*LLDA system aims at modelling lexicographic data, so takes the definitions of these words as found in a monolingual dictionary as a starting point. The relevant definitions are given in table 1 (these are cleaned-up version of the definitions in the Longman Dictionary of Contemporary English, henceforth LDOCE).

colt	a young male horse
fil·ly	a young female horse
foal[1]	a young horse
mare	a fully-grown female horse
stal·lion	a fully-grown male horse

Table 1: Definitions of Words for Horses

The definitions in table 1 are analysed in the SIM*u*LLDA set-up as relating English words to defining aspects of the meanings expressed by these words. These defining attributes are called *definitional attributes*. As an example, the first definition in table 1 relates the word colt to the definitional attributes **male** and **young**. On top of these definitional attributes, colt is related to a sense of horse. But this meaning of horse is itself also related in the dictionary to definitional attributes and a further meaning of animal, etc. This will go on until the genus term is what you might call an *empty genus term*. The claim is that *thing* in a definition reading *a thing which ...* is just there because a lexical definition without a genus term is hard to formulate (in some cases). In this way, all lexical definition can be 'unravelled' into sets of definitional attributes. For simplicity, I will here ignore the relation of the words in table 1 to the word horse, and treat **horse** as if it were a definitional attribute. This leads to a situation in which the definitions in table 1 are analysed as in table 2.

	horse	**male**	**female**	**adult**	**young**
HORSE	×				
STALLION	×	×		×	
MARE	×		×	×	
FOAL	×				×
FILLY	×		×		×
COLT	×	×			×

Table 2: Definitional Attributes for Horses

So in the SIM*u*LLDA set-up, every word expresses a number of meanings, and these meanings are analysed in terms of sets of definitional attributes. And these defini-

[1]This situation is symmetrical in EuroWordNet: DEDO and FINGER are also related via the ILI FINGER. But that has no impact on the example.

tional attributes are nothing more than the accumulated differentiae specificae from their lexical definitions in monolingual dictionaries.

3.2. Formal Concept Analysis

The data in table 2 are organised within the SIM*u*LLDA set-up by means of a logical framework called Formal Concept Analysis (henceforth FCA). FCA was developed by Ganter and Wille in Darmstadt (**?**). It is an attempt to give a formal definition of the notion of a 'concept', within the boundaries of a model-theoretic framework. The idea behind FCA is the following: in a model, those objects that share a common set of attributes belong together; they form the extension of a concept, the intention of which is the set of attributes that they share.

The formal representation of FCA is follows. Take a set of objects *G*, a set of attributes *M*, and a relation *I* relating the objects to the attributes. We define the set of formal concepts $\mathfrak{B}$ over a context (G, M, I) in the following way:

$$B^{\downarrow} = \{g \in G \mid \forall b \in B \,.\, (g, b) \in I\} \quad (1)$$

$$A^{\uparrow} = \{m \in M \mid \forall a \in A \,.\, (a, m) \in I\} \quad (2)$$

$$\mathfrak{B}(G, M, I) = \{\langle A, B\rangle \mid A = B^{\downarrow} \wedge B = A^{\uparrow}\} \quad (3)$$

The way FCA is applied in SIM*u*LLDA is as follows: the meanings in table 2 are taken as formal objects (the elements of *G*), and the definitional attributes relation to them are taken as formal attributes (the elements of *M*). This lead to a set $\mathfrak{B}$ of formal concepts consisting of pairs of sets of meanings and sets of definitional attributes. There are ten such formal concepts in total, which are listed in table 3.

$\langle\{$HORSE, COLT, STALLION, MARE, FOAL, FILLY$\}, \{$**horse**$\}\rangle$
$\langle\{$MARE, FILLY$\}, \{$**horse, female**$\}\rangle$
$\langle\{$MARE$\}, \{$**horse, female, adult**$\}\rangle$
$\langle\{$STALLION, COLT$\}, \{$**horse, male**$\}\rangle$
$\langle\{$STALLION, MARE$\}, \{$**horse, adult**$\}\rangle$
$\langle\{$STALLION$\}, \{$**horse, male, adult**$\}\rangle$
$\langle\{$FOAL, COLT, FILLY$\}, \{$**horse, young**$\}\rangle$
$\langle\{$COLT$\}, \{$**horse, male, young**$\}\rangle$
$\langle\{$FILLY$\}, \{$**horse, female, young**$\}\rangle$
$\langle\emptyset, \{$**horse, female, young, male, adult**$\}\rangle$

Table 3: Formal Concepts for Horses

The formal concepts in $\mathfrak{B}$ have a natural order: formal concepts with more defining attributes are more specific those with less defining attributes. And also, all those objects that belong to a subconcept also belong to its superconcept. So we define an order relation $\leq$ over $\mathfrak{B}$ as follows:

$$\langle A_1, B_1\rangle \leq \langle A_2, B_2\rangle \Leftrightarrow A_1 \subseteq A_2 \Leftrightarrow B_2 \subseteq B_1 \quad (4)$$

The relation $\leq$ orders the formal concepts in table 3 into a lattice structure, which can be displayed in a Hasse-diagram as in figure 1. The nodes in this lattice represent the formal concepts, where the related sets of meanings and attributes can be found as follows: all formal concept below the node above which the definitional attribute **young** is placed have **young** in their set of definitional attributes, and conversely, all nodes above COLT have COLT in their set of meanings (i.e. a definitional attributes **a** is put above $\langle \mathbf{a}^{\downarrow}, \mathbf{a}^{\downarrow\uparrow}\rangle$, and a meaning A is depicted under $\langle \mathrm{A}^{\uparrow\downarrow}, \mathrm{A}^{\uparrow}\rangle$).

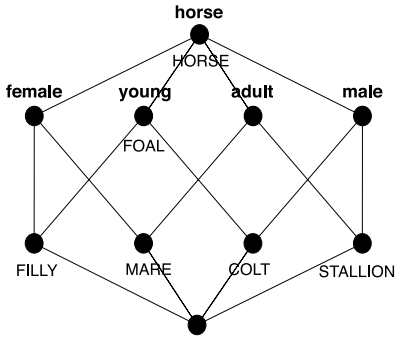


Figure 1: Concept Lattice for Horses

The construction of a concept lattice from a tabular representation of a context can be done automatically on-line by means of Java Applet written as part of my thesis. The Java-Applet is called JaLaBA (a Java Lattice Building Application). JaLaBA gives ask for a set of formal objects and a set of definitional attributes, and a relation between them, gives the related set of formal concepts, and then displays a 3D rotatable model of the corresponding Hasse diagram. JaLaBA can be found on the web-site of my thesis: `http://maarten.janssenweb.net/simullda`.

3.3. Interlingual Concept Lattice

The meanings in SIM*u*LLDA are abstracted from monolingual dictionaries. So the meanings STALLION in table 2 is derived from LDOCE. But the meaning STALLION as such is not an English meaning: the same meaning can be expressed by the French word étalon. Therefore the formal objects in SIM*u*LLDA are not taken to be language dependent meanings, but rather *interlingual meanings*, which can be expressed by words in various languages. It is clear that the definitional attributes defining these interlingual meanings cannot be language specific themselves. So also definitional attributes in SIM*u*LLDA are interlingual entities: **female** is a language independent definitional attribute, that can be lexicalised in English by the expression *female*, but also in French by the expression *femelle*, or in Dutch by the expression *mannelijk*.

Since the lattice in figure 1 thus contains only language independent entities, it can be taken as an interlingual structure, to which words of various languages can be related. This gives the situation as depicted in figure 2. Some notational conventions related to this figure: every interlingual meaning *y* has a (possibly empty) set of words lexicalising it in every language *X*, denoted by $\mathrm{wrd}_X(y)$, and every word *x* of every language has a set of interlingual meanings *Y* it expresses, denoted by $\mathrm{mng}(x)$.

In the set-up depicted in figure 2, it is possible to find translational synonyms: *x* is a translational synonym of *y*, iff $\mathrm{wrd}_Y(\mathrm{mng}(x)) \supseteq y$. To give an example:

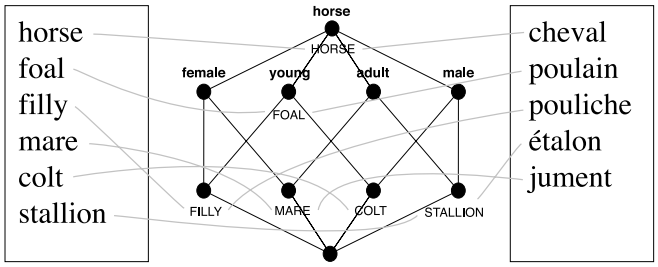


Figure 2: Concept Lattice with Words

mng(stallion) $\supseteq$ STALLION, and $\text{wrd}_{French}(\text{STALLION}) \supseteq$ étalon, so étalon is a translational synonym of stallion. In other words, just following the lines gives you translational synonyms.

More interesting is the situation when there is a lexical gap. In the SIM*u*LLDA set-up, there is a lexical gap iff $\text{wrd}(\text{mng}(x)) = \emptyset$. An example of a lexical gap in figure 2 is that there is no French translational synonym for colt. There only is the more general translational hyperonym poulain.

To find a translational hyperonym for a word *x*, first take mng*x*), and look up the lattice to find the first superconcept which has an interlingual meaning depicted under it for which there is a lexicalisation in the target language. So for colt, this interlingual meaning would be FOAL, and the fact hat poulain is a translational hyperonym of colt is modelled by the fact that COLT $\subseteq$ mng(colt), the related formal concept $\langle \text{COLT}^{\uparrow\downarrow}, \text{COLT}^{\uparrow} \rangle$ (I will use **COLT** as a name for this formal concept) is a subconcept of **FOAL**, and $\text{wrd}_{French}(\text{FOAL}) \supseteq$ poulain.

As claimed in the previous section, the things keeping colt and poulain apart should be the differentiae specificae. And differentiae specificae are implicitly present in the SIM*u*LLDA set-up: if we consider the formal concepts **COLT** and **FOAL**, then by the simple fact that **COLT** $\leq$ **FOAL**, we know that **COLT** has more definitional attributes than **FOAL**. If we define a function *ext* to give the set of definitional attributes of a formal concept ($\text{ext}(\langle A, B \rangle) = B$), then this *definitional surplus* will be $\text{ext}(\textbf{COLT}) \backslash \text{ext}(\textbf{FOAL}) =$ **male**. So **male** is the differentiam specificam distinguishing COLT from other hyponyms of FOAL such as FILLY.

The differentiae specificae, as well as the genus proximum, are hence modelled at the interlingual level. Within the interlingua, you could say that 'COLT = FOAL + **male**'. The language specific differentiae specificae are obtained by taking the lexicalisation in the desired language of this definitional surplus. We get the translation of our lexical gap by lexicalising both parts of the right-hand side of this equation in the target language. Since the French lexicalisation of **male** is *mâle*, we can conclude that colt in French is *poulain mâle*. This process of generating a translation for a lexical gap is called *lexical gap filling*. Notice that the lexical gap filling procedure renders what Zgusta (**?**) calls an *explanatory equivalent*, and not a *translational equivalent*.

We could also have opted to lexicalise all elements of the above equation within the same language, hence in English relating the word colt to the description *male foal*. In this way, also lexical definitions can be retrieved from the system. Notice that this lexical definition *male foal* is not the same definition as the one that formed the starting point of the analysis (see table 1): LDOCE does in fact not give the genus proximum, but a more remote genus term. But firstly, the rendered definition is nevertheless correct, and secondly, the LDOCE definition can also be rendered in the same way: we also have that **COLT** $\leq$ **HORSE**, with a larger definitional surplus: {**young, male**}. This leads to the original definition of colt as *young male horse*. The claim is that the generation of lexical definitions, as well as the lexical gap filling procedure, does not give a unique result, but does give only correct results.

Let me conclude this section by observing that not all definitional attributes are as 'simple' as the ones in this example. For instance, the Petit Robert definition of bief is *canal qui conduit les eaux d'un cours d'eau vers une machine hydraulique*[2]. There is no translational synonym in English for bief, but given an analysis of the data in SIM*u*LLDA, we would have that 'BIEF = CANAL + **qcled-cvumh**', where the lexicalisation in English of CANAL would be canal, and the English lexicalisation of **qcled-cvumh** would be *bringing water from a stream to a hydraulic installation*. So any differentiam specificam can be captured by a definitional attribute.

4. Definitional Attributes

As I have tried to show in the previous two sections, there is a definite need for differentiae specificae in a lexical database, especially in a multilingual one. That it is possible to set up a system using such differentiae specificae such as in the SIM*u*LLDA set-up. And that such a set-up leads to a correct modelling of lexical relations even in such problematic cases as lexical gaps. But of course the differentiae specificae introduced in a system, such as the definitional attributes in the case of SIM*u*LLDA, are at least reminiscent of the very thing WordNet reacted against: Katz & Fodor style semantics primitives (**?**). So naturally, from the perspective of semantic network theories, there is a reluctance to introduce differentiae specificae.

In the theory of Katz & Fodor, semantic markers are supposed to provide the foundation of knowledge, by their being innate building blocks to which all concepts can be reduced. But the presence of semantic primitives does not necessarily entail such a strongly reductionistic theory of meaning; there are more modest versions of semantic primitives, such as for instance in the French tradition of *sémantique interpretative*, as advocated by Rastier (**?**), Pottier (**?**) and others. The semantic primitives in this theory are called *sèmes*, which constitute meaning units calles *sémèmes*. Rastier explicitly discusses that sèmes do not have any of the strong properties semantic markers are supposed to have: they are not innate, not universal, not (interestingly) indivisible, they are not (necessarily) small in number, and they are not qualities of a referent or part of

[2]It actually is *canal de dérivation qui* ..., but I want to avoid here the for this point irrelevant question whether *canal de dérivation* should be taken as a complex genus term, or whether *de dérivation* counts as a differentiam specificam.

a concept. Especially in its description by Messelaar (**?**), sèmes have a striking resemblance to definitional attributes.

I do not want to give here an elaborate description of sèmes, their relation to semantic markers or a comparison to the SIM*u*LLDA set-up: definitional attributes are not sèmes either. But it is important to observe that the introduction of definitional attributes does not entail a strong theory of meaning. Definitional attributes are meant to be little more than what they are: theoretical entities that help to distinguish hyponyms of the same genus, and that make it possible to generate bilingual lexical definitions even for non-corresponding meanings. In my thesis, I give a lengthy discussion of the nature of the basic element of the SIM*u*LLDA set-up: words, word-forms, languages, interlingual meanings, and definitional attributes. For the moment, I will merely mention three properties definitional attributes are explicitly *not* supposed to have.

Firstly, definitional attributes do not form a special closed set of indivisible, innate semantic primitives. This should be clear from the example in section 2: the differentiam specificam *used for driving the wheel of a watermill* will constitute a definitional attribute, even though it has a clear internal structure. As a definitional attribute, it will count as an atomic entity, disregarding its internal structure[3]. So it is not an interestingly indivisible definitional attributes. And it would clearly be absurd to suppose that such a definitional attribute is in any way innate. New concepts arise every day, and new concepts can entail new definitional attributes, so there is not even a closed set of definitional attributes: new definitional attributes are introduced when need arises.

Secondly, sets of definitional attributes do not constitute a complete description of the concept related to the word that expresses the interlingual meaning in question. That is to say, interlingual meanings in the SIM*u*LLDA set-up are in a way defined in terms of sets of definitional attributes. But that does not result in saying that all information related to the word expressing that interlingual meaning is captured by the definitional attributes. For instance, stylistic information and other language-internal characteristics of the word are not modelled by the interlingual meaning, but handled at the level of the individual languages. Also, prototypes play an important role in the information/concept related to a word. But prototypes cannot be interlingual since, as shown by for instance Putnam (**?**), prototypes do not translate[4]. So the SIM*u*LLDA set-up is not supposed to provide a knowledge base: it is a lexical database, containing some aspects of word-meaning. In particular those aspects necessary for producing the kind of bilingual definition found in bilingual dictionaries.

Thirdly, definitional attributes are not denotational in nature. Definitional attributes are aspects of word meanings, not of (the) objects denoted by those words. And the interlingual meaning and/or the related set of definitional attributes are not supposed to fix the denotation of the word. Denotational semantics is very problematic, and it is even very dubious if every word(meaning) can be said to have a fixed denotation at any given moment. Furthermore, denotational semantics can never give a complete picture of word meaning. For instance, words can be metaphorically attributed to objects, where the meaning of the word is applied without the claim that the object to which it is attributed falls under the denotation of the word. So the fact that within the SIM*u*LLDA set-up, **COLT** is a subconcept of **FOAL** is not intended to express the ontological inclusion of the class of colts in the class of foals[5]: SIM*u*LLDA provides a lexical hierarchy, which should not be taken as an ontological hierarchy.

This last point is independent of the presence of differentiae specificae: also hierarchical systems without differentiae specificae, such as WordNet, should be taken as providing a lexical hierarchy, and not an ontological hierarchy. It is even dubious whether there really is an ontological ordering on the world. This is not to say that SIM*u*LLDA is not an ontology in the sense often used in computer science. For instance, the set-up is in many ways comparable to the *ontology clustering* set-up proposed by Visser & Tamma (**?**), which has a shared ontology and attributes over the concepts in it. Also in their set-up, a translation for a lexical gap is created after "*the attributes of the concept in the source ontology are compared with the attributes of the hypernym [found in the shared ontology] to select the distinguishing features.*" The point is that SIM*u*LLDA does not provide an ontological hierarchy in the philosophical sense.

Given the modest nature of definitional attributes, it will be clear that there are no strong claims concerning the meanings in the SIM*u*LLDA set-up. This is not surprising if you consider that SIM*u*LLDA aims at modelling lexicographic definitions, and lexicographic definitions do not really 'give' a description of the meanings of a word; they rely on knowledge of related words to 'hint at' the meaning of the word. A nice example of this is given by Hanks (**?**), who shows that a lexicographic definition of a chinaman (say *a left-hander's googly*) is only useful if you know about googlies, leg breaks, off-breaks and related cricket terms. Given the elusive nature of words, any theory that makes strong(er) claims is likely to runs into grave problems.

5. Conclusion

In this article, I hope to have shown the need for a structural modelling of differentiae specificae in a (multilingual) lexical database, and the advantages of the SIM*u*LLDA set-up which has such differentiae specificae by means of its definitional attributes. As already said, the criticism in this article was mainly directed at the EuroWordNet set-up, but applies equally to other hierarchical systems without differentiae specificiae. For instance, as far as I can tell, the

[3] In my thesis, I discuss some cases in which adopting a certain internal structure for definitional attributes proves beneficial, and also discuss order sets of definitional attributes, but in general, definitional attributes are atomic.

[4] Putnam goes on to claim that *perceptual prototypes may be psychologically important, but they just aren't* meanings – *not even "narrow" ones (op.cit. p.46).*. Although I am not unsympathetic with this point, it is not this strong claim I am aiming at here.

[5] This independently of the questions whether all colts are in fact foals.

SIMPLE framework, which in a way is a succesor of EuroWordNet, does not add structure to overcome the problems described in section 2.

Of course, the question whether SIM*u*LLDA could really provide a better alternative for a system like EuroWordNet is an (at least partly) empirical question: lexical databases and knowledge bases are designed for practical applicability. The SIM*u*LLDA approach is, however, a theoretical feasibility study, performed as a PhD-project, and the SIM*u*LLDA system has not (yet) been implemented or tested at large scale.

This is not to say that there is no empirical evidence for the applicability of the system: in my thesis, there is an empirical test whether the around 50 words for bodies of water from 6 different languages (English, French, Dutch, German, Italian, and Russian) can be correctly handled within the SIM*u*LLDA set-up. Describing the results of this test here would be too lengthy, and the test did bring forward some problems (or weaknesses) of the set-up. But the claim is that all the problems that have a solutions could be solved to satisfaction within the system. Although this does not provide a large-scale test, it does show that within an actual domain of lexical definitions, the systems works properly. The lexical field was not arbitrarily chosen, but was taken because it is a lexical field that is often quoted as problematic, both in terms of definability, as in terms of cross-linguistic differences, such as the often cited case of *river* and *fleuve*. So it is intended to provide some empirical evidence for the practical applicability of the system. But the only way to really test it is of course to build an application and fill it with data.

6. References

Bernhard Ganter and Rudolf Wille. 1996. *Formale Begriffsanalyse: mathematische grundlagen*. Springer Verlag, Berlin.

Patrick Hanks. 2000. Contributions of lexicography and corpus linguistics to a theory of language performance. In *Proceedings of the Ninth Euralex International Congress*, Stuttgart.

Maarten Janssen. 2002. *SIMuLLDA: a Multilingual Lexical Database Application using a Structured Interlingua*. Ph.D. thesis, Universiteit Utrecht, Utrecht.

Jerrold J. Katz and Jerry A. Fodor. 1963. The structure of a semantic theory. *Language*, vol. 39:170 – 210.

P.A. Messelaar. 1990. *La Confection du Dictionnaire Générale Bilingue*. Peeters, Leuven.

George A. Miller. 1998. Foreword. In Christiane Fellbaum, editor, *Wordnet: an Electronic Lexical Database*. MIT Press, Cambridge.

Bernard Pottier. 1980. Sémantique et noémique. *Annuario de Estudios filológicos*, vol. 3:169 – 177.

Hillary Putnam. 1988. Fodor and block on "narrow content". In *Representation and Reality*. MIT Press, Cambridge.

François Rastier. 1987. *Sémantique Interprétative*. Presses Universitaires de France, Paris.

Pepijn R.S. Visser and Valentina A.M. Tamma. 1999. An experience with ontology-based agent clustering. In Benjamins, Chandrasekaran, Gomez-Perez, Guarino, and Uschold, editors, *Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-Solving Methods (KRR5)*, Stockholm.

Piek Vossen and Ann Copestake. 1993. Untangling definition structure into knowledge representation. In Ted Briscoe, Valeria de Paiva, and Ann Copestake, editors, *Inheritance, Defaults, and the Lexicon*. Cambridge University Press, Cambridge.

Ladislav Zgusta. 1971. *Manual of Lexicography*. Mouton, Den Haag.

Merging Global and Specialized Linguistic Ontologies

Bernardo Magnini and Manuela Speranza

ITC-irst, Istituto per la Ricerca Scientifica e Tecnologica
I-38050 Povo (Trento), Italy
{magnini, manspera}@itc.it

Abstract

There is an increasing interest in linguistic ontologies (e.g. WordNet) for a variety of content-based tasks, including conceptual indexing, word sense disambiguation and cross-language information retrieval. A relevant contribution in this direction is represented by linguistic ontologies with domain specific coverage, which are a crucial topic for the development of concrete application systems.
This paper tries to go a step further in the direction of the interoperability of specialized linguistic ontologies, by addressing the problem of their integration with global ontologies. This scenario poses some simplifications with respect to the general problem of merging ontologies, since it enables to define a strong precedence criterion so that terminological information overshadows generic information whenever conflicts arise. We assume the EuroWordNet model and propose a methodology to "plug" specialized linguistic ontologies into global ontologies. Experimental data related to an implemented algorithm, which has been tested on a global and a specialized linguistic ontology for the Italian language, are provided.

1. Introduction

Ontologies have become an important topic in research communities across several disciplines in relation to the key challenge of making the Internet and the Web a more friendly and productive place by filling more meaning to the vast and continuously growing amount of data on the net. The surging interest in the discovery and automatic or semi-automatic creation of complex, multi-relational knowledge structures, in fact, converges with recent proposals from various communities to build a Semantic Web relying on the use of ontologies as a means for the annotation of Web resources.

There is also an increasing interest in linguistic ontologies, such as WordNet, for a variety of content-based tasks, such as conceptual indexing and semantic query expansion to improve retrieval performance. More recently, the role of linguistic ontologies is also emerging in the context of distributed agents technologies, where the problem of meaning negotiation is crucial. A relevant perspective in this direction is represented by linguistic ontologies with domain specific coverage, whose role has been recognized as one of the major topics in many application areas.

This paper tries to go a step further in the direction of the interoperability of specialized linguistic ontologies, by addressing the problem of their integration with global linguistic ontologies. The possibility of merging information at different levels of specificity seems to be a crucial requirement at least in the case of large domains where terminologies include both very specific terms and a significant amount of common terms that may be shared with global ontologies.

The global-specialized scenario poses some simplifications with respect to the general problem of merging ontologies at the same degree of specificity (Hovy, 1998); in particular, in the case of conflicting information, it is possible to define a strong precedence criterion according to which terminological information overshadows generic information. We assume the EuroWordNet model and propose a methodology to "plug" specialized linguistic ontologies into global ontologies. The formal apparatus to realize this is based on plug relations that connect *basic concepts* of the specialized ontology to corresponding concepts in the generic ontology. We provide experimental data to support our approach, which has been tested on a global and a specialized linguistic ontology for the Italian language.

The paper is structured as follows. Section 2 presents the main features and uses of linguistic ontologies as opposed to formal ontologies. Section 3 introduces specialized linguistic ontologies, i.e. linguistic ontologies with domain specific coverage, as opposed to global linguistic ontologies containing generic knowledge. Section 4 deals with the problem of the interoperability of linguistic ontologies and describes the relations and the procedure enabling an integrated access of pairs of global and specialized linguistic ontologies.

2. Linguistic ontologies versus formal ontologies

In the recent years the increasing interest in ontologies for many natural language applications has led to the creation of ontologies for different purposes and with different features; therefore, it is worth pointing out the distinction between two main kinds of existing ontologies, i.e. formal and linguistic ontologies.

Linguistic ontologies are large scale lexical resources that cover most words of a language, while at the same time also providing an ontological structure where the main emphasis is on the relations between concepts; linguistic ontologies can therefore be seen both as a particular kind of lexical database and as particular kind of ontology.

Linguistic ontologies mainly differ from formal ontologies as far as their degree of formalization is concerned. Linguistic ontologies, in fact, do not reflect all the inherent aspects of formal ontologies. As Guarino et al. (1999) point out, for instance, WordNet's upper level structure shows no distinction between types and roles, whereas most of the original Pangloss (Knight and Luk, 1994) nodes in the Sensus ontology are actually types; to give a further example, WordNet's hierarchical structure lacks information about mutual disjointness between concepts.

Moreover, what distinguishes linguistic ontologies from formal ontologies, is their size: linguistic ontologies are very large (WordNet, for instance, has several dozen thousand synsets), while formal ontologies are generally much smaller.

The duality characterizing linguistic ontologies is reflected in their most prominent features. If we consider the linguistic level, they are strongly language-dependent, like electronic dictionaries, glossaries and all other linguistic resources, which focus on the words used in one specific language (in the case of monolingual resources) or in two or more specific language (in the case of bilingual or multilingual resources). On the other hand, if we consider the semantic level, we can observe that concepts denotated by different words in different languages can be shared, as it happens with the concepts in formal ontologies. In fact it is possible, at least for the core Indo-European languages, to identify a common ontological backbone behind the lexical surface of different languages (Guarino et al., 1999).

WordNet (Fellbaum, 1998), the best-known linguistic ontology, is an electronic lexical database where each sense of a lemma belongs to a different synset, i.e. a set of synonyms. Synsets are organized hierarchically by means of hypernymy and hyponymy relations. In WordNet other kinds of semantic relations among synsets are defined (e.g. role relation, part-of relation and cause relation), so as to build a more rich and complex semantic net. WordNet thus offers two distinct services: a lexicon, which describes the various word senses, and an ontology, which describes the semantic relationships among concepts.

As a linguistic ontology, WordNet is strongly language-dependent, but as an ontology it could also be adapted to a cross-language environment using the EuroWordNet multilingual database (Vossen, 1998) and mapping synsets into the EuroWordNet InterLingual Index, i.e. the index that links monolingual wordnets for all the languages covered by EuroWordNet. There are several examples of monolingual wordnets for many other languages, such as Dutch, Spanish, Italian, German and Basque.

A formal ontology based on linguistic motivation is the Generalized Upper Model (GUM) knowledge base (Bateman et al., 1995), an ontology primarily developed for Natural Language Processing applications. An upper model is an abstract linguistically motivated ontology meeting two requirements at the same time: i) a sufficient level of abstraction in the semantic types employed, as to escape the idiosyncrasies of surface realization and ease interfacing with domain knowledge, and ii) a sufficiently close relationship to surface regularities as to permit interfacing with natural language surface components.

2.1. Uses of formal ontologies

Recently ontologies have been used in the context of the Semantic Web. Ontologies can be employed to associate meaning with data and documents found on the Internet thus boosting diverse applications of information-retrieval systems. For the retrieval of information from the Web, Luke et al. (1996) propose a set of simple HTML Ontology Extensions to manually annotate Web pages with ontology-based knowledge, which performs high precision but is very expensive in terms of time.

OntoSeek (Guarino et al., 1999) is also based on content, but uses ontologies to find user's data in a large classical database of Web pages. Erdmann and Studer (1999) use an ontology to access sets of distributed XML documents on a conceptual level. Their approach defines the relationship between a given ontology and a document type definition (DTD) for classes of XML document. Thus, they are able to supplement syntactical access to XML documents by conceptual access.
However, as pointed out by Guarino et al. (1999), the practical adoption of ontologies in information-retrieval systems is limited by their insufficiently broad coverage and their need to be constantly updated; linguistic ontologies encompass both ontological and lexical information thus offering a way to partly overcome these limitations.

2.2. Uses of linguistic ontologies

Linguistic ontologies, and WordNet in particular, are proposed for content-based indexing, where semantic information is added to the classic word-based indexing. As an example, *Conceptual Indexing* (Woods, 1997) automatically organizes words and phrases of a body of material into a conceptual taxonomy that explicitly links each concept to its most specific generalizations. This taxonomic structure is used to organize links between semantically related concepts, and to make connections between terms of a request and related concepts in the index.

Mihalcea and Moldovan (2000) designed an IR system which performs a combined word-based and sense-based indexing exploiting WordNet. The inputs to IR systems consist of a question/query and a set of documents from which the information has to be retrieved. They add lexical and semantic information to both the query and the documents, during a preprocessing phase in which the input question and the texts are disambiguated. The disambiguation process relies on contextual information, and identifies the meaning of the words using WordNet.

The proble of sense disambiguation in the context of an IR task has been addressed, among the others, also by Gonzalo et al. (1998). In a preliminary experiment where disambiguation had been done manually, the vector space model for text retrieval gives better results if WordNet synsets are chosen as the indexing space, instead of word forms.

Desmontils and Jacquin (2001) present an approach where linguistic ontologies are used for information retrieval on the Internet. The indexing process is divided into four steps: i) for each page a flat index of terms is built; ii) WordNet is used to generate all candidate concepts which can be labeled with a term of the previous index; iii) each candidate concept of a page is studied to determine its representativeness of this page content; iv) all candidate concepts are filtered via an ontology, selecting the more representative for the content of the page.

More recently, the role of linguistic ontologies is also emerging in the context of distributed agents technologies, where the problem of meaning negotiation is crucial (Bouquet and Serafini, 2001).

3. Specialized linguistic ontologies

A particular kind of linguistic ontologies is represented by specialized linguistic ontologies, i.e. linguistic ontologies with domain specific coverage, as opposed to global linguistic ontologies, which contain generic knowledge. Focusing on one single domain, specialized linguistic ontologies often provide many sub-hierarchies of highly specialized concepts, whose lexicalizations tend to assume the shape of complex terms (i.e. multi-words); high level knowledge, on the other hand, tends to be simplified and domain oriented.

Many specialized linguistic ontologies have been developed, especially for practical applications, in domains such as art (see the Art and Architecture Getty Thesaurus), geography (see the Getty Thesaurus of Geographical Names), medicine (Gangemi et al., 1999), etc. and the importance of specialized linguistic ontologies is widely recognized in a number of works. The role of terminological resources for Natural Language Processing is addressed, for instance, by Maynard and Ananiadou (2000), who point out that high quality specialized resources such as dictionaries and ontologies are necessary for the development of hybrid approaches to automatic term recognition combining linguistic and contextual information with statistical information.

Buitelaar and Sacaleanu (2002) address the problem of tuning a general linguistic ontology such as WordNet or GermaNet to a specific domain (the medical domain, in the specific case). This involves both selecting the senses that are most appropriate for the domain and adding novel specific terms. Similarly, Turcato et al. (2000), describe a method for adapting a general purpose synonym database, like WordNet, to a specific domain (in this case, the aviation domain), adopting an eliminative approach based on the incremental pruning of the original database.

The use of domain terminologies also arises the problem of the (automatic) acquisition of thematic lexica and their mapping to a generic resource (Buitelaar and Sacaleanu, 2001; Vossen, 2001; Lavelli et al., 2002). As far as automatic term extraction is concerned, Basili et al. (2001) investigate whether syntactic context (i.e. structural information on local term context) can be used for determining "termhood" of given term candidates, with the aim of defining a weakly supervised "termhood" model suitably combining endogenous and exogenous syntactic information.

4. Merging global and specialized linguistic resources: the plug-in approach

One of the basic problems in the development of techniques for the Semantic Web is the integration of ontologies. Indeed the Web consists of a variety of information sources, and in order to extract information from such sources, their semantic integration is required.

Merging linguistic ontologies introduces issues concerning the amount of data to be managed (in the case of WordNet we have several dozen thousand synsets), which are typically neglected when upper levels are to be merged (Simov et al., 2001).

This paper tries to go a step further in the direction of the interoperability of linguistic ontologies, by addressing the problem of the integration of global and specialized linguistic ontologies. The possibility of merging information at different levels of specificity seems to be a crucial requirement at least in the case of domains, such as Economics or Law, that includes both very specific terms and a significant amount of common terms that may be shared by the two ontologies. We assume the EuroWordNet model and propose a methodology to "plug" specialized ontologies into global ontologies, i.e. to access them in conjunction through the construction of an integrated ontology.

4.1. Correspondences between global and specialized linguistic ontologies

A global linguistic ontology and a specialized one complement each other. The one contains generic knowledge without domain specific coverage, the other focuses on a specific domain, providing sub-hierarchies of highly specialized concepts. This scenario allows some significant simplifications when compared to the general problem of merging two ontologies. On the one hand, we have a specialized ontology, whose content is supposed to be more accurate and precise as far as specialized information is concerned; on the other hand, we can assume that the global ontology guarantees a more uniform coverage as far as high level concepts are concerned. These two assumptions provide us with a powerful precedence criterion for managing both information overlapping and inheritance in the integration procedure.

In spite of the differences existing between the two ontologies, in fact, it is often possible to find a certain degree of correspondence between them. In particular, we have information *overlapping* when the same concept belongs to the global and to the specialized ontology, and *over-differentiation* when a terminological concept has two or more corresponding concepts in the global ontology or the other way round. Finally, some specific concepts referring to technical notions may have no corresponding concept in the global ontology, which means there is a *conceptual gap*; in such cases a correspondence to the global ontology can be found through a more generic concept.

The sections highlighted in the global and the specialized ontology represented in Figure 1 reflect the correspondences we typically find between the two kinds on ontologies.

As for the global ontology (the bigger triangle), area *B1* is highlighted since it corresponds to the sub-hierarchies containing the concepts belonging to the same specific domain of the specialized ontology (the smaller triangle). The middle part of the specialized ontology, which we call *B* area, is also highlighted and it corresponds to concepts which are representative of the specific domain but are also present in the global ontology.

When the two ontologies undergo the integration procedure, an integrated ontology is constructed (Figure 2). Intuitively, we can think of it as if the specialized ontology somehow shifts over the global. In the integrated ontology, the information of the generic is maintained, with the exclusion of the sub-hierarchies containing the concepts belonging to the domain of the specialized ontology, which are covered by the corresponding area of the specialized. The

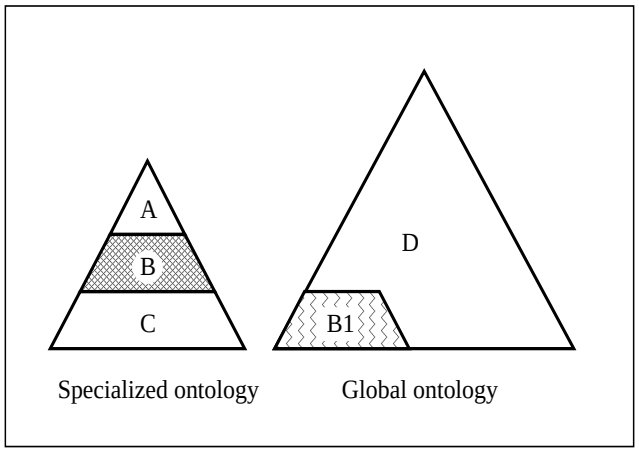


Figure 1: Separate specialized and global ontologies. Overlapping is represented in colored areas

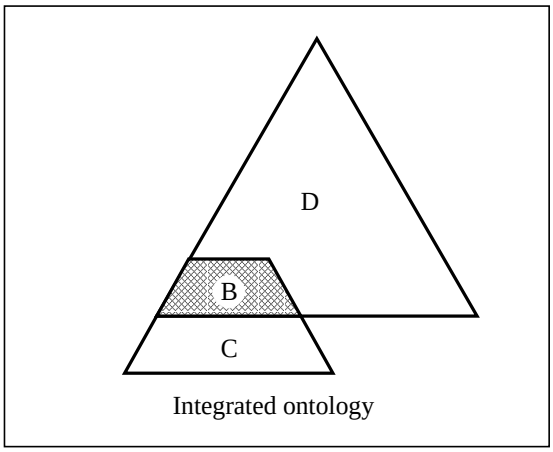


Figure 2: Integrated ontology. As to overlapping, precedence is given to the specialized ontology

integrated ontology also contains the most specific concepts of the specialized ontology (*C* area), which are not provided in the generic. What is excluded from the integrated ontology is the highest part of the hierarchy of the specialized ontology; it is represented by area *A* and contains generic concepts not belonging to a specialized domain, which are expected to be treated more precisely in the generic ontology.

4.2. Plug relations

The formal apparatus to realize an integrated ontology is based on the use of three different kinds of relations (plug-synonymy, plug-near-synonymy and plug-hyponymy) that connect basic concepts of the specialized ontology to the corresponding concepts in the global ontology, and on the use of eclipsing procedures that shadow certain concepts, either to avoid inconsistencies, or as a secondary effect of a plug relation.

A plug relation directly connects pairs of corresponding concepts, one belonging to the global ontology and the other to the specialized ontology. The main effect of a plug relation is the creation of one or more "plug concepts", which substitute the connected concepts, i.e. those directly involved in the relation. To describe the relations inherited by a plug concept, the following classification, adapted from Hirst and St-Onge (1998) is used: *up-links* of a concept are those whose target concept is more general (i.e. hypernymy and instance-of relations), *down-links* are those whose target is more specific (i.e. hyponymy and has-instance relations) and *horizontal-links* include all other relations (i.e. part-of relations, cause relations, derivation, etc.).

Plug-synonymy is used when overlapping concepts are found in the global ontology (hereafter *GO*) and in the specialized ontology (hereafter *SO*). The main effect of establishing a relation of plug-synonymy between concept *C* belonging to the global ontology (indicated as C^{GO}) and $C1^{SO}$ (i.e. concept *C1* belonging to the specialized ontology) is the creation of a plug concept $C1^{PLUG}$. The plug concept gets its linguistic forms (i.e. synonyms) from *SO*, up-links from GO, down-links from *SO* and horizontal-links from *SO* (see Table 1). As a secondary effect, the up relations of $C1^{SO}$ and the down relations of C^{GO} are eclipsed.

	$C1^{PLUG}$
Up links	*GO*
Down links	*SO*
Horizontal links	*GO* + *SO*

Table 1: Merging rules for plug-synonymy and plug-near-synonymy.

Plug-near-synonymy is used in two cases: (i) over-differentiation of the *GO*, i.e. when a concept in the *SO* has two or more corresponding concepts in the *GO*; this happens, for instance, when regular polysemy is represented in the GO but not in the SO; (ii) over-differentiation of the *SO*, i.e. when a concept in the *GO* corresponds to two or more concepts in the *SO*; this situation may happen as a consequence of subtle conceptual distinctions made by domain experts, which are not reported in the global ontology. Establishing a plug-near-synonymy relation has the same effect of creating a plug-synonymy (see Table 1).

Plug-hyponymy is used to connect concepts of the specialized ontology to more generic concepts in the case of conceptual gaps. The main effect of establishing a plug-hyponymy relation between C^{GO} (i.e. concept *C* of the global ontology) and $C1^{SO}$ (i.e. concept *C* of the specialized ontology) is the creation of the two plug concepts C^{PLUG} and $C1^{PLUG}$ (see Table 2). C^{PLUG} gets its linguistic forms from the *GO*, up-links from the *GO*, down-links are the hyponyms of C^{GO} plus the link to $C1^{PLUG}$ and horizontal-links from the *GO*. The other plug node, $C1^{PLUG}$, gets its linguistic form from the *SO*, C^{PLUG} as hypernym, down links from the *SO* and horizontal links from the *SO*. As a secondary effect, the hypernym of $C1^{SO}$ is eclipsed.

Eclipsing is a secondary effect of establishing a plug re-

	C^{PLUG}	$C1^{PLUG}$
Up links	*GO*	C^{PLUG}
Down links	*GO* + $C1^{PLUG}$	*SO*
Horizontal links	*GO*	*SO*

Table 2: Merging rules for plug-hyponymy

lation and is also an independent procedure used to avoid the case that pairs of overlapping concepts placed inconsistently in the taxonomies are included in the merged ontology; this could happen, for instance, when "whale" is placed under a "fish" sub-hierarchy in a common sense ontology, while also appearing in the mammal taxonomy of a scientific ontology.

4.3. Integration procedure

The plug-in approach described in the previous subsection has been realized by means of a semi-automatic procedure with the following four main steps.

(1) Basic concepts identification. The domain expert identifies a preliminary set of "basic concepts" in the specialized ontology. These concepts are highly representative of the domain and are also typically present in the global ontology. In addition, it is required that basic concepts are disjoint among each other and that they assure a complete coverage of the specialized ontology, i.e. it is required that all terminal nodes have at least one basic concept in their ancestor list.

(2) Alignment. This step consists in aligning each basic concept with the more similar concept of the global ontology, on the basis of the linguistic form of the concepts. Then, for each pair a plug-in configuration is selected among those described in Section 4.2.

(3) Merging. For each plug-in configuration an integration algorithm reconstructs the corresponding portion of the integrated ontology. If the integration algorithm detects no inconsistencies, the next plug-in configuration is considered, otherwise step 4 is called.

(4) Resolution of inconsistencies. An inconsistency occurs when the implementation of a plug-in configuration is in contrast with an already realized plug-in. In this case the domain expert has to decide which configuration has the priority and consequently modify the other configuration, which will be passed again to step 2 of the procedure.

5. Experiments

The integration procedure described in Section 4.3 has been tested within the SI-TAL project [1] to connect a global wordnet and a specialized wordnet that have been created independently. ItalWordNet (IWN) (Roventini et al., 2000), which was created as part of the EuroWordNet project (Vossen, 1998) and further developed through the introduction of adjectives and adverbs, is the lexical database involved in the plug-in as a generic resource and consists of about 45,000 lemmas. Economic-WordNet (ECOWN) is a specialized wordnet for the economic domain and consists of about 5,000 lemmas distributed in about 4,700 synsets. Table 3 summarizes the quantitative data of the two resources considered.

	Specialized	Generic
Synsets	4,687	49,108
Senses	5,313	64,251
Lemmas	5,130	45,006
Internal Relations	9,372	126,326
Variants/synsets	1.13	1.30
Senses/lemmas	1.03	1.42

Table 3: IWN and ECOWN quantitative data

As a first step, about 250 basic synsets (5.3% of the resource) of the specialized wordnet were manually identified by a domain expert, including, for instance "azione" ("share"), and excluding less informative synsets, such as "azione" ("action"). Alignment with respect to the generic wordnet (step 2 of the procedure) is carried out with an algorithm that considers the match of the variants. Candidates are then checked by the domain expert, who also chooses the proper plug relation. In the case of gaps, a synset with a more generic meaning was selected and a plug-hyponymy relation was chosen.

At this point the merging algorithm takes each plug relation and reconstructs a portion of the integrated wordnet. In total, 4,662 ECOWN synsets were connected to IWN: 577 synsets (corresponding to area *B* in Figure 2) substitute the synsets provided in the global ontology to represent the corresponding concepts (*B1* area in Figure 1); 4085 synsets, corresponding to the most specific concepts of the domain (*C* area in Figure 2) are properly added to the database. 25 high level ECOWN synsets (*A* area in Figure 1) were eclipsed as the effect of plug relations. The number of plug relations established is 269 (92 plug-synonymy, 36 plug-near-synonymy and 141 plug-hyponymy relations), while 449 IWN synsets with an economic meaning were eclipsed, either as a consequence of plug relations (when the two taxonomic structures are consistent) or through the independent procedure of eclipsing (when the taxonomies are inconsistent). Each relation connects on average 17,3 synsets.

6. Conclusions

After discussing the main features and uses of linguistic ontologies as opposed to formal ontologies, we have addressed the problem of the interoperability between linguistic ontologies. We have presented a methodology for the integration of a global and a specialized linguistic ontology. The global-specialized situation allows to define a strong precedence criterion to solve cases of conflicting information. The advantage of the approach is that a limited number of plug relations allows to connect a large amount of concepts (i.e. synsets) in the two ontologies.

[1] Si-TAL (Integrated System for the Automatic Treatment of Language) is a National Project devoted to the creation of large linguistic resources and software for Italian written and spoken language processing.

7. References

R. Basili, M.T. Pazienza, and F.M. Zanzotto. 2001. Modelling syntactic context in automatic term extraction. In *Proc. of Recent Advances in Natural Language Processing (RANLP '01)*, Tzigov Chark, Bulgaria, September.

J.A. Bateman, B. Magnini, and G. Fabris. 1995. The generalized upper model knowledge base: Organization and use. In *Proc. of International Conference on Bulding and Sharing of Very Large-Scale Knowledge Bases*, Twente, The Netherlands, April.

P. Bouquet and L. Serafini. 2001. Two formalizations of a context: a comparison. In *Proc. of Third International Conference on Modeling and Using Context*, Dundee, Scotland, July.

P. Buitelaar and B. Sacaleanu. 2001. Ranking and selecting synsets by domain relevance. In *Proc. of NAACL Workshop WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, Pittsburgh, June. held in conjunction with NAACL2001.

P. Buitelaar and B. Sacaleanu. 2002. Extending synsets with medical terms. In *Proc. of the First Global WordNet Conference*, Mysore, India, January.

E. Desmontils and C. Jacquin. 2001. Indexing a web site with a terminology oriented ontology. In *Proc. of SWWS International Semantic Web Working Symposium*, Stanford University, USA, July, August.

M. Erdmann and R. Studer. 1999. Ontologies as conceptual models for XML documents. In *Proc. of the Twelfth Workshop on Knowledge Acquisition, Modeling and Management (KAW '99)*, Voyager Inn, Banff, Alberta, Canada, October.

C. Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, US.

A. Gangemi, D.M. Pisanelli, and G. Steve. 1999. Overview of the ONIONS project: Applying ontologies to the integration of medical terminologies. *Data and Knowledge Engineering*, 31.

J. Gonzalo, F. Verdejio, Chugur, and J. Cigarran. 1998. Indexing with WordNet synsets can improve text retrieval. In S. Harabagiu, editor, *Proceeding of the Workshop "Usage of WordNet in Natural Language Processing Systems"*, Montreal, Quebec, Canada, August.

N. Guarino, C. Masolo, and G. Vetere. 1999. OntoSeek: Contet-based access to the web. *IEEE Intelligent Systems and Their Application*, 14(3):70–80.

G. Hirst and D. St-Onge. 1998. Lexical chains representations of context for the detection and correction of malapropisms. In C. Fellbaum, editor, *WordNet. An Electronic Lexical Database*. The MIT Press.

E. Hovy. 1998. Combining and standardizing large-scale, pratical ontologies for machine translation and other uses. In *Proc. of the First International Conference on Language Resources and Evaluation*, Granada, Spain, August.

K. Knight and S. Luk. 1994. Building a large knowledge base for machine translation. In *Proceedings of the American Association of Artificial Intelligence Conference AAAI-94*, Seattle, WA.

A. Lavelli, B. Magnini, and F. Sebastiani. 2002. Building thematic lexical resources by bootstrapping and machine learning. In *Proc. of the Workshop "Linguistic Knowledge Acquisition and Representation: Bootstrapping Annotated Language Data", Workshop at LREC-2002*. to appear.

S. Luke, L. Spector, and D. Rager. 1996. Ontology-based knowledge discovery on the world-wide-web. In *Proc. of the AAAI1996 Workshop on Internet-based Information Systems*, Portland, Oregon, August.

D. Maynard and S. Ananiadou. 2000. Creating and using domain-specific ontologies for terminological applications. In *Proc. of Second International Conference on Language Resources and Evaluation (LREC-2000)*, Athens, Greece, May, June.

R. Mihalcea and D. Moldovan. 2000. Semantic indexing using WordNet senses. In *Proc. of the ACL workshop on Recent Advances in Natural Language Processing and Information Retrieval*, Hong Kong, October.

A. Roventini, A. Alonge, F. Bertagna, B. Magnini, and N. Calzolari. 2000. ItalWordNet: a large semantic database for Italian. In *Proc. of the Second International Conference on Language Resources and Evaluation (LREC-2000)*, Athens, Greece, May, June.

K. I. Simov, K. Kiryakov, and M. Dimitrov. 2001. OntoMap - the guide to the upper-level. In *Proc. of SWWS International Semantic Web Working Symposium*, Stanford University, USA, July, August.

D. Turcato, F. Popowich, J. Toole, D. Fass, D. Nicholson, and G. Tisher. 2000. Adapting a synonym database to specific domains. In *Proc. of Workshop on Information Retrieval and Natural Language Processing*, Hong-Kong, October. held in conjunction with ACL2000.

P. Vossen, editor. 1998. *EuroWordNet: a Multilingual Database with Lexical Semantic Networks*. Kluwer Academic Publishers.

P. Vossen. 2001. Extending, trimming and fusing WordNet for technical documents. In *Proc. of NAACL Workshop WordNet and Other Lexical Resources: Applications, Extensions and Customizations*, Pittsburgh, June. held in conjunction with NAACL2001.

W.A. Woods. 1997. Conceptual indexing: A better way to organize knowledge. Technical report, SUN Technical Report TR-97-61.

Automatic Adaptation of WordNet to Domains

Roberto Navigli, Paola Velardi

Università di Roma "La Sapienza", Dipartimento di Scienze dell'Informazione, Via Salaria 113
00198 Roma, Italy, e-mail: velardi@dsi.uniroma1.it

Abstract

The objective of this paper is to present a method to automatically enrich WordNet with sub-trees of concepts in a given language domain. WordNet is then trimmed to reduce unnecessary ambiguity and singleton nodes. The process is based on the use of statistical method and linguistic processing to extract candidate domain *terms*. Multiword terms are semantically disambiguated and interpreted using ontological and contextual knowledge stored in WordNet on singleton words.

1. Introduction

As already pointed out by many researchers, WordNet is a very useful tool, but has some important drawbacks, namely, over-ambiguity and lack of domain terminology. Several published studies attempted to solve this problem in some automatic way, for example, (Vossen, 2001) (Harabagiu et al., 1999) (Milhalcea et al., 2001) and (Agirre et al. 1999). Other studies related to the work presented in this paper deal with the more general issue of automatic ontology construction. These contributions are collected in the web proceedings of two workshops dedicated to Ontology learning, (ECAI-OL, 2000) and (IJCAI-OL, 2001).

In many described approaches for ontology learning, domain terms are firstly extracted using a variety of statistical methods; then, taxonomic relations and other types of relations between terms are detected. In the literature, the notion of domain *term* and domain *concept* are used interchangeably, though no semantic interpretation of terms takes place. For example, in (Vossen, 2001) the "concept" *digital printing technology* is considered as a kind-of *printing technology* by virtue of simple string inclusion. However, *printing* has four senses in WordNet, and *technology* has two senses. There are hence 8 possible concept combinations for *printing technology*!

In this paper we propose a method for semantic interpretation of terms, using the information available in WordNet for the individual words that appear in a terminological string. Semantic interpretation allows us to detect non-trivial taxonomic relations between *concepts*, and other types of semantic relations.

The method described in this paper is implemented in a system called OntoLearn. OntoLearn is part of an Ontology Engineering architecture, described in (Missikoff et al., 2002), developed in the context of two European projects[1], aimed at improving interoperability in the Tourism sector.

Taxonomic information is extracted from the documents available in the considered domain in 5 steps: domain terminology is identified (section 2) and structured in syntactic trees (section 3), terms are mapped to concepts (section 4), that are arranged in a domain concept forest (section 5), and then used to create a domain-specific view of WordNet (section 6).

[1] ITS – 13015 (FETISH) and ITS- 29329 (HARMONISE).

2. Identification of Relevant Domain Terminology

The objective of this phase is to extract from the available documents a domain terminology. First, we use a linguistic processor, ARIOSTO[2], to extract from a corpus of documents a list of syntactically plausible terminological patterns, e.g. compounds (*credit card*), prepositional phrases (*board of directors*), adjective-noun relations (*manorial house*).

Then, two information theory based measures are used to filter out non-terminological (e.g. *last week*) and non-domain specific terms (e.g. *world wide web* in a Tourism domain). The first measure, called *Domain Relevance*, computes the probability of occurrence of a candidate term in the application domain (e.g. Tourism), as compared with other corpora that we use for a contrastive analysis (e.g. Medicine, Economy, Novels, etc.). The second measure, called *Domain Consensus*, computes the entropy of the probability of seeing a candidate term across the documents of the application domain. The underlying idea is that only terms that are *frequently* and *consistently* referred in the available domain documents reflect some *consensus* on the use of that term. These two measures have been formally defined and extensively evaluated in (Velardi et al, 2001).

3. Generation of Syntactic Trees

From the list of filtered terminology we generate *lexicalized trees*, on the basis of a simple inclusion relation. For example, given two strings x and wx (e.g. *telephone service* and *service*), we generate $wx \rightarrow^{@} x$, where '$\rightarrow^{@}$' stands for the hyperonymy relation. Figure 1 provides an example of a generated lexicalized tree $\mathfrak{I}$. It is clear that many taxonomic relations are not captured by this simple inclusion mechanism, like *bus service* $\rightarrow^{@}$ *public transport service*.

4. Semantic Disambiguation of Terms

The process of *semantic interpretation* is one that associates to each multiword term $t = w_n \cdot \ldots \cdot w_2 \cdot w_1$ (where w_i is an atomic word) the appropriate *concept name*.

[2] ARIOSTO is a joint effort of the Universities of Roma "La Sapienza" and "Tor Vergata".

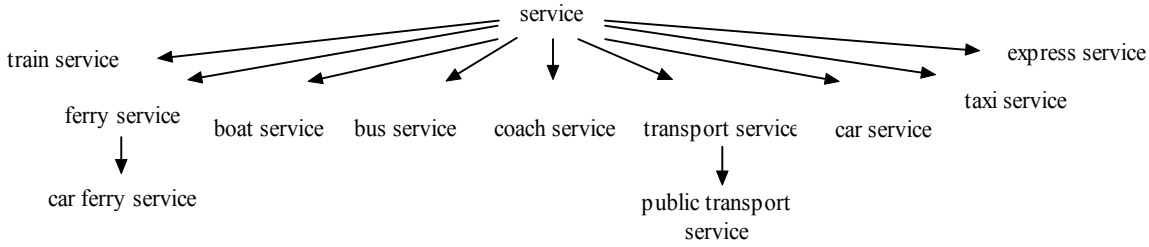


Figure 1. Example of a lexicalized tree.

Though complex terms are usually absent in WordNet, singleton words and occasionally word pairs included in a terminological string are mostly present. For example, *printing technology* as a unique term is not included, but *printing* and *technology* have an associated WordNet entry.

We derive the meaning of a complex terminological string *compositionally*, as explained hereafter.

Formally, a *semantic interpretation* is defined as follows: let $t = w_n \cdot \ldots \cdot w_2 \cdot w_1$ be a valid term belonging to a lexicalized tree $\mathfrak{I}$. The process of *semantic interpretation* is one that associates to each word w_k in t the appropriate WordNet synset S_i^k, the i-th synset ($i \in \{1,...,m\}$) associated to w_k in WordNet. The *sense* of t is hence defined as:

$$S(t) = \bigcup_k S^k,\ S^k \in Synsets(w_k) \text{ and } w_k \in t.$$

where $Synsets(w_k)$ is the set of synsets each representing a sense of the word w_k.

For instance: S("transport company") = { { transportation#4, shipping#1, transport#3 }, { company#1 } } corresponding to sense #1 of *company* ("*an institution created to conduct business*") and sense #3 of *transport* ("*the commercial enterprise of transporting goods and material*").

In order to disambiguate the words in a term $t = w_n \cdot \ldots \cdot w_2 \cdot w_1$ we proceed as follows:

a) If t is the first analyzed element of $\mathfrak{I}$, manually disambiguate the root node (w_1 if t is a compound) of $\mathfrak{I}$.

b) For any $w_k \in t$ and any synset S_i^k of w_k, create a *semantic net SN*. Semantic nets are automatically created using the following semantic relations: hyperonymy ($\rightarrow^{@}$), hyponymy ($\rightarrow^{\sim}$), meronymy ($\rightarrow^{\#}$), holonymy ($\rightarrow^{\%}$), pertonymy ($\rightarrow^{\backslash}$), attribute ($\rightarrow^{=}$), similarity ($\rightarrow^{\&}$), gloss ($\rightarrow^{gloss}$) and topic ($\rightarrow^{topic}$). The *gloss* and the *topic* relation are obtained parsing with ARIOSTO the WordNet concept definitions (*glosses*) and SemCor sentences (*topic*) including that sense. Every other relation is directly extracted from WordNet. To reduce the dimension of a SN, concepts at a distance of more than 3 relations from the SN centre, S_i^k, are removed. Figure 2a is an example of SN generated for sense #1 of *room*.

Let then $SN(S_i^k)$ be the semantic network for sense i of word w_k.

c) Starting from the "head" w_1 of t, and for any pair of words w_{k+1} and w_k (k=1,…,n-1) belonging to t, intersect alternative pairs of SNs. Let $I = SN(S_i^{k+1}) \cap SN(S_j^k)$ be one of such intersections for sense i of word w_{k+1} and sense j of word w_k. Note that, in each step k, the word w_k is already disambiguated, either manually (for k=1) or as a result of step k-1.

To identify common semantic patterns several heuristic rules are used, e.g.:

$$\exists G, M \in Synset_{wn} : S_1 \xrightarrow{gloss} G \xrightarrow{@^{\leq 3}} M \xleftarrow{@^{\leq 3}} S_2$$

The heuristic (named "gloss+parallelism") reads: "*given two central concepts S_1 and S_2, there exist two concepts G and M such that G appears in the gloss of S_1 and both G and S_2 reach the concept M in $SN(S_1) \cap SN(S_2)$ through a hyperonimy path.*

An example is the bold pattern in Figure 2b:

transport#3 $\xrightarrow{gloss}$ *enterprise*#2 $\xrightarrow{@^1}$ *organization*#1 $\xleftarrow{@^2}$ *company*#1.

5. Creating a Domain Concept Forest

Initially, all the terms in a tree $\mathfrak{I}$ are independently disambiguated. Subsequently, taxonomic information in WordNet is used to detect *is-a* relations between *concepts*, e.g. *ferry service* $\rightarrow^{@}$ *boat service*. In this phase, since all the elements in $\mathfrak{I}$ are jointly considered, some interpretation errors produced in the previous disambiguation step are corrected. In addition, certain concepts are *fused* in a unique concept name on the basis of pertonimy, similarity and synonymy relations (e.g. respectively: *manor house* and *manorial house*, *expert guide* and *skilled guide*, *bus service* and *coach service*).

Notice again that we detect semantic relations between *concepts*, not words. For example, *bus#1* and *coach#5* are synonyms, but this relation does not hold for other senses of these two words. Each lexicalized tree $\mathfrak{I}$ is finally transformed in a *domain concept* tree Υ.

Figure 3 shows the concept tree obtained from the lexicalized tree of Figure 1.

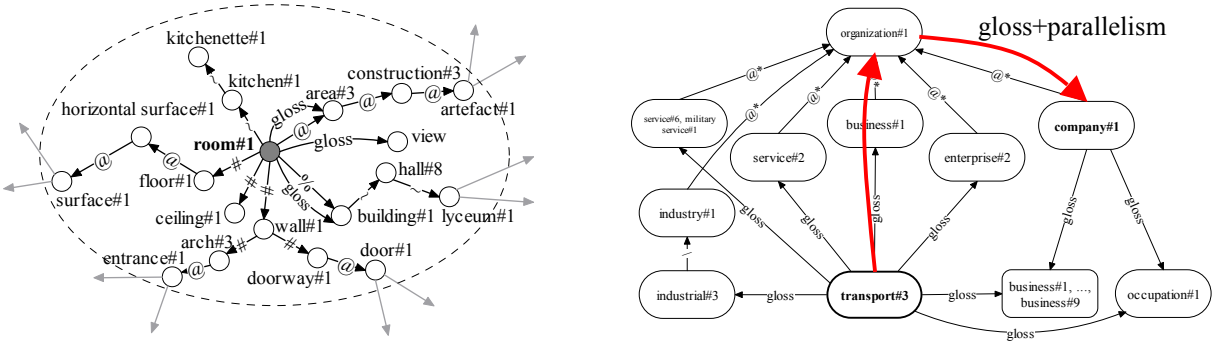


Figure 2. a) example of semantic net for *room#1*; b) example of intersecting semantic patterns for transport#3 and *company#1*.

For clarity, in Figure 3 concepts are labeled with the associated terms (rather than with synsets), and numbers are shown only when more than one semantic interpretation holds for a term, as for *coach service* and *bus service* (e.g. sense #3 of "bus" refers to "old cars").

6. Pruning and Trimming WordNet

The final phase consists in creating a domain-specialization of WordNet. In short, WordNet pruning and trimming is accomplished as follows:

1. The Domain Concept trees are attached under the appropriate nodes in WordNet.
2. An intermediate node in WordNet is pruned whenever the following conditions hold together

   i. it has no "brother" nodes;
   ii. it has only one direct hyponym;
   iii. it is not the root of a Domain Concept tree;
   iv. it is not at a distance ≤ 2 from a WordNet *unique beginner* (this is to preserve a "minimal" top ontology).

Figure 4 shows an example of pruning the nodes located over the Domain Concept tree with root *wine#1*. Appendix A shows an example of domain-adapted branch of WordNet in the tourism domain.

7. Evaluation

OntoLearn is a knowledge extraction system aimed at improving human productivity in the time-consuming task of building a domain ontology. Our experience in building a tourism ontology for the European project Harmonise reveals that, after one year of ontology engineering activities, the tourism experts were able to release the most general layer of the tourism ontology, comprising about 300 concepts.

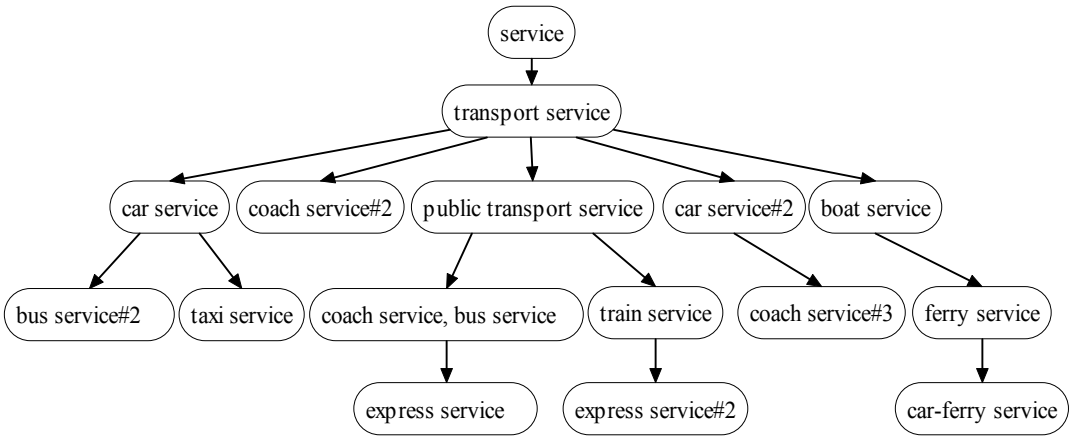


Figure 3. A Domain Concept Tree.

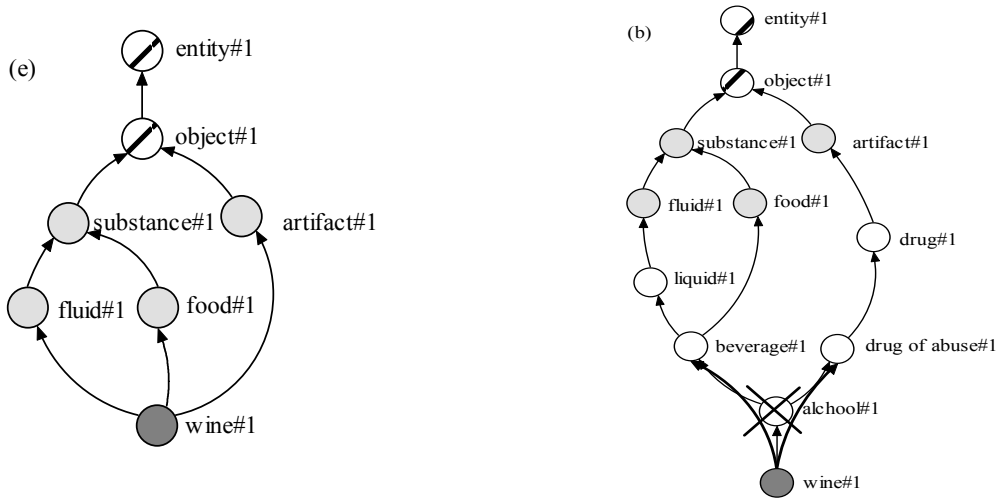


Figure 4. An intermediate step and the final pruning step over the Domain Concept Tree for "wine#1".

Then, we decided to speed up the process developing the *OntoLearn* system, aimed at supporting the ontology engineering tasks. This produced a significant acceleration in ontology building, since in the next 6 months[3] the tourism ontology reached about 3,000 concepts.

The OntoLearn system has been also evaluated independently from the ontology engineering process. We extracted from a 1 million-word corpus of travel descriptions (downloaded from Tourism web sites) a terminology of 3840 terms, manually evaluated[4] by domain experts participating in the Harmonise project. We obtained a precision ranging from 72.9% to about 80% and a recall of 52.74%. The precision shift is motivated by the well-known fact that the intuition of experts may significantly differ.

After this expert evaluation, we added few *ad hoc* heuristics that brought the precision to 97%. However, the use of heuristics limits the generality of the method.

The recall has been estimated by submitting a list of 6000 syntactic candidates to the experts, requiring them to mark truly terminological entries, and then comparing this list with that obtained by our statistical filtering method described in section 2.

We personally evaluated the semantic disambiguation algorithm using a test bed of about 650 extracted terms, which have been manually assigned to the appropriate WordNet concepts. These terms contributed to the creation of 90 syntactic trees. The entire process of semantic disambiguation and creation of domain trees has been evaluated, leading to an overall 84.5% precision. The precision grows to about 89% for highly structured sub-trees, as those in Figure 3. In fact, the phase described in section 5 significantly contributes at eliminating disambiguation errors (in the average, 5% improvement). We also analyzed the individual contribution of each of the heuristics mentioned in section 4 to the performance of the method, but a detailed performance report is omitted here for sake of space. The results of this performance analysis led to a refinement of the algorithm and the elimination of one heuristic.

[3] The time span includes also the effort needed to test and tune OntoLearn. Manual verification of automatically acquired domain concepts actually required few days.

[4] Here manual evaluation is simply deciding whether an extracted term is relevant, or not, for the tourism domain.

8. References

Agirre E., Ansa O., Hovy E. and Martinez D. *Enriching very large ontologies using the WWW*, in (ECAI-OL 2000).

Harabagiu S., Moldovan D. *Enriching the WordNet Taxonomy with Contextual Knowledge Acquired from Text*. AAAI/MIT Press, 1999.

Milhalcea R., Moldovan D. I. *eXtended WordNet: progress report.* NAACL 2001 Workshop, Pittsburg, June 2001.

Missikoff M., Velardi P. and Fabriani P. *Using Text Processing Techniques to Automatically enrich a Domain Ontology*. Proc. of ACM Conf. On Formal Ontologies and Information Systems, ACM_FOIS, Ogunquit, Maine, October 2002.

Velardi P., Missikoff M. and Basili R. *Identification of relevant terms to support the construction of Domain Ontologies*. ACL-EACL Workshop on Human Language Technologies, Toulouse, France, July 2001.

Vossen P. *Extending, Trimming and Fusing WordNet for technical Documents*, NAACL 2001 workshop on WordNet and Other Lexical Resources, Pittsbourgh, July 2001.

ECAI 2000, workshop on Ontology Learning http://ol2000.aifb.uni-karlsruhe.de/

IJCAI 2001, workshop on Ontology Learning http://ol2001.aifb.uni-karlsruhe.de/

Appendix A: A fragment of trimmed WordNet for the Tourism domain

{ activity%1 }
{ work%1 }
{ project:00508925%n }
{ tourism_project:00193473%n }
{ ambitious_project:00711113%a }
{ service:00379388%n }
{ travel_service:00191846%n }
{ air_service#2:00202658%n }
{ air_service#4:00194802%n }
{ transport_service:00716041%n }
{ ferry_service#2:00717167%n }
{ express_service#3:00716943%n }
{ exchange_service:02413424%n }
{ guide_service:04840928%n }
{ restaurant_service:03233732%n }
{ rail_service:03207559%n }
{ maid_service:07387889%n }
{ laundry_service:02911395%n }
{ customer_service:07197309%n }
{ guest_service:07304921%n }
{ regular_service#2:07525988%n }
{ outstanding_customer_service:02232741%a }
{ tourism_service:00193473%n }
{ waiter_service:07671545%n }
{ regular_service:02255650%a,scheduled_service:02255439%a }
{ personalized_service:01703424%a,personal_service:01702632%a }
{ secretarial_service:02601509%a }
{ religious_service:02721678%a }
{ church_service:00666912%n }
{ various_service:00462055%a }
{ helpful_service:02376874%a }
{ quality_service:03714294%n }
{ air_service#3:03716758%n }
{ room_service:03250788%n }
{ car_service#3:02384960%n }
{ car_service#4:02385109%n }
{ car_service#5:02364995%n }
{ hour_room_service:10938063%n }
{ transport_service#2:02495376%n }
{ car_service:02383458%n }
{ bus_service#2:02356871%n }
{ taxi_service:02361877%n }
{ coach_service#2:02459686%n }
{ public_transport_service:03184373%n }
{ bus_service:02356526%n,coach_service:02356526%n }
{ express_service#2:02653414%n }
{ local_bus_service:01056664%a }
{ train_service:03528724%n }
{ express_service:02653278%n }
{ car_service#2:02384604%n }
{ coach_service#3:03092927%n }
{ boat_service:02304226%n }
{ ferry_service:02671945%n }
{ car-ferry_service:02388365%n }
{ air_service:05270417%n }
{ support_service:05272723%n }

Extraction of Implicit Knowledge from WordNet

Wim Peters

NLP Group
Department of Computer Science
University of Sheffield
Regent Court
211 Portobello Street
Sheffield S1 4DP
U.K.
wim@dcs.shef.ac.uk

Abstract

Lexical knowledge databases such as WordNet contain much semantic information that is left implicit. In order to make maximal use of these resources it is important to make this implicit semantic information explicit. Metonymy and regular polysemy constitute a type of implicit ontological knowledge. This paper describes the semi-automatic extraction of systematically related word senses from WordNet by exploiting its hierarchical structure, and the identification of relations that link these on the basis of the glosses.

1. Introduction

WordNet (Fellbaum 1998) contains far more semantic information than its ontological organization shows. Word senses are related to senses of other words by means of a small number of basic semantic relations such as synonymy and hypernymy. Other types of encyclopaedic knowledge and semantic relations are implicitly present in the structure of WordNet in the form of taxonomic correspondences and glosses. This non-formalized semantic information in WordNet can be processed in order to distil more implicit knowledge (see e.g. Harabagiu 2000).

2. Relations between senses

Systematic relatedness between senses is one type of knowledge that is mostly left implicit in resources. This phenomenon is called metonymy, or, ore specifically, regular polysemy (Apresjan 1973).

Viewed traditionally, metonymy is a non-literal figure of speech in which the name of one thing is substituted for that of another related to it. It has been described as a cognitive process in which one conceptual entity, the vehicle, provides mental access to another conceptual entity (Radden 1999). In its basic form, it establishes a semantic relation between two concepts that are associated with word forms. The semantic shift expressed by the relation may or may not be accompanied by a shift in form. The semantic relation that is captured by metonymy is one of semantic contiguity, in the sense that in many cases there are systematic relations between metonymically related concepts that can be regarded as slots in conceptual frames (cf. Fillmore 1977).

Regular polysemy is a more specific instantiation of metonymy that covers the systematicity of the semantic relations involved. It can be defined as a subset of metonymically related senses of the same word displaying a conventional as opposed to novel type of semantic contiguity relation. Any systematic semantic relations between concepts are lexicalized, i.e. they are explicitly listed in dictionaries and independent of a pragmatic situation. For example, *The White House* is on the one hand an institution and on the other a building. The semantic relation between the two senses is 'is housed in'. It is a conventional pattern, not a nonce formation (a pragmatically defined novel metonymy), because it holds for related senses of two or more words (Apresjan, 1973) in the lexicon. It is this subtype of metonymy that we concentrate on in this paper.

3 Extraction from WordNet

A technique was developed (Peters 2000) for identifying sense combinations in WordNet where the senses involved potentially display a regular polysemic relation, i.e. where the senses involved are candidates for systematic relatedness.

In order to obtain these candidate patterns WordNet (WN) has been automatically analysed by exploiting its hierarchical structure for nouns. Wherever there are two or more nouns with senses in one part of the hierarchy, which also have senses in another part of the hierarchy, then we have a candidate pattern of regular polysemy. The patterns are candidates because there seems to be an observed regularity for two or more words. An example can be found in Figure 1 below.

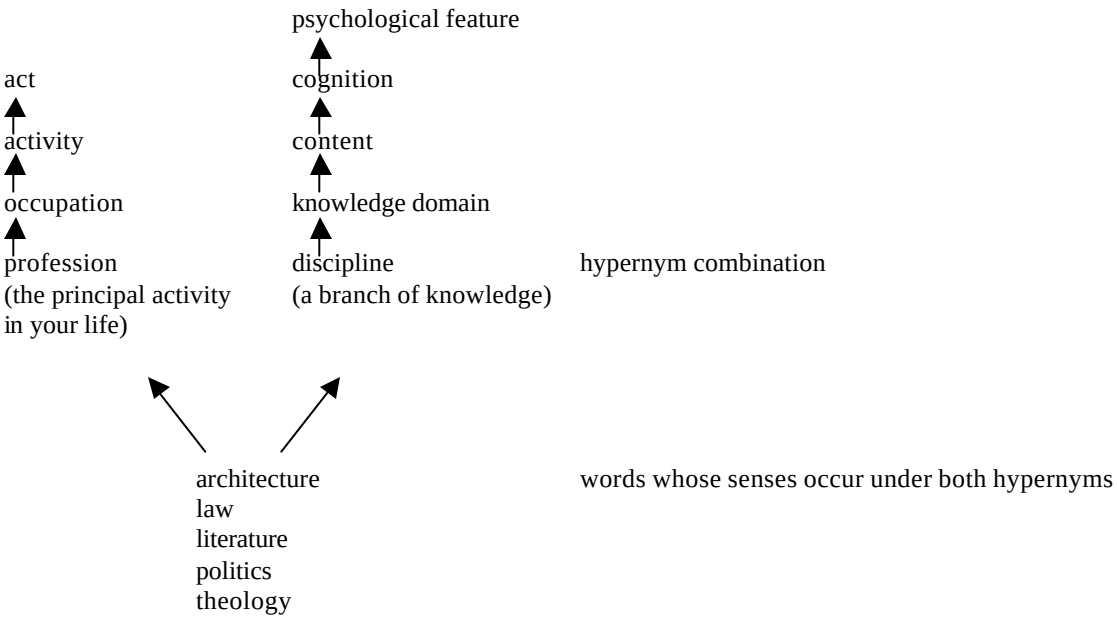


Figure 1: words in WordNet covered by the pattern profession/discipline

4 Relations

The results obtained from the manual analysis of reduced data sets according to (Peters 2001) and (Peters 2002) yields a set Regular Polysemic patterns. These patterns consist of combinations of the hypernymic nodes that subsume the words involved in the pattern. These combinations do not give any explicit information about the nature of the systematic relations that exist between them. This relationship can be determined by means of manual evaluation. The examination of the pair and the participating word senses will provide a human assessor with enough information to intuitively postulate a relationship. However, this is a costly and time consuming activity.

We have, up to a certain extent, automated this process of extracting explicit relations between the word senses involved in the regular polysemic pattern.

Our extraction process concentrates on the linguistic information available in the glosses associated with the word senses subsumed by the hypernymic pairs. The relations we have extracted take the form of verbs that link pairs of concepts. In each of these pairs one member is subsumed by member one of the hypernym pair and the other by number two. The glosses were first preprocessed. Part of speech tags were added and nominal and verbal content words were lemmatized.

For all nouns participating in the regular polysemic patterns listed above two bags of WordNet words were created, each associated with a sense captured by the regular polysemic pattern. The bag consisted of the noun under consideration, its synonyms and all the members of the hypernymic synsets. Then the words in the bag of the first word sense were matched against the processed gloss associated with the synset to which this sense belongs (henceforth synset 1).

If there was a match, the words from the bag of the second word sense (henceforth synset 2) were matched against the gloss.

If there was a match and the word from the synset 1 bag (word 1) preceded the word from the synset 2 bag (word 2) within the gloss, the text between the matches was extracted. If this span of text contained a verb, it was extracted, together with any associated prepositions. A distance of three positions between the matched nouns and the verb was applied in order to reduce spurious matches. Any extracted verb is considered to represent an instantiation of the relation(s) holding for the regular polysemic pattern.

The same matching process was repeated for the glosses associated with all hypernyms of synset 1. Then the whole process was repeated, looking for matches in the synset 2 gloss and all its hypernyms. Figure 2 below gives a graphical representation of the process.

The requirement that word 1 precedes word 2 is geared towards the extraction of transitive and prepositional verbs, both used in active form. The order constraint also determines the directionality of the relation, i.e. which hypernymic pair member is the subject and which is the object of an extracted verb.

We will illustrate this by means of an example.

The regular polysemic pattern **animal - food** is applicable to 172 words in WordNet. One of these words is '*herring*':

Sense 1: commercially important food fish of northern waters of both Atlantic and Pacific.

Sense 2: valuable flesh of fatty fish from shallow waters of northern Atlantic or Pacific; usually salted or pickled.

The bag of words associated with synset 1 contains 330 words (e.g. *fish, entity, life form, vertebrate, craniate*).

The synset 2 bag holds 518 words (*seafood, food, substance, food product, nutrient, object*). Only a subset of these words is related to *herring*, the rest are associated with the other words that are subsumed by the hypernymic pattern.
The concept 'food fish' is the hypernym of sense 1: "any fish used for food by human beings". Of the words in this gloss 'fish' is found in the synset1 bag and 'food' in the synset 2 bag. The intermediate text span is 'used for' which consists of a past participle and a preposition. The outcome is the relationship 'animal used for food'. This relation is found 37 times. The relation 'used for' is found 23 times.

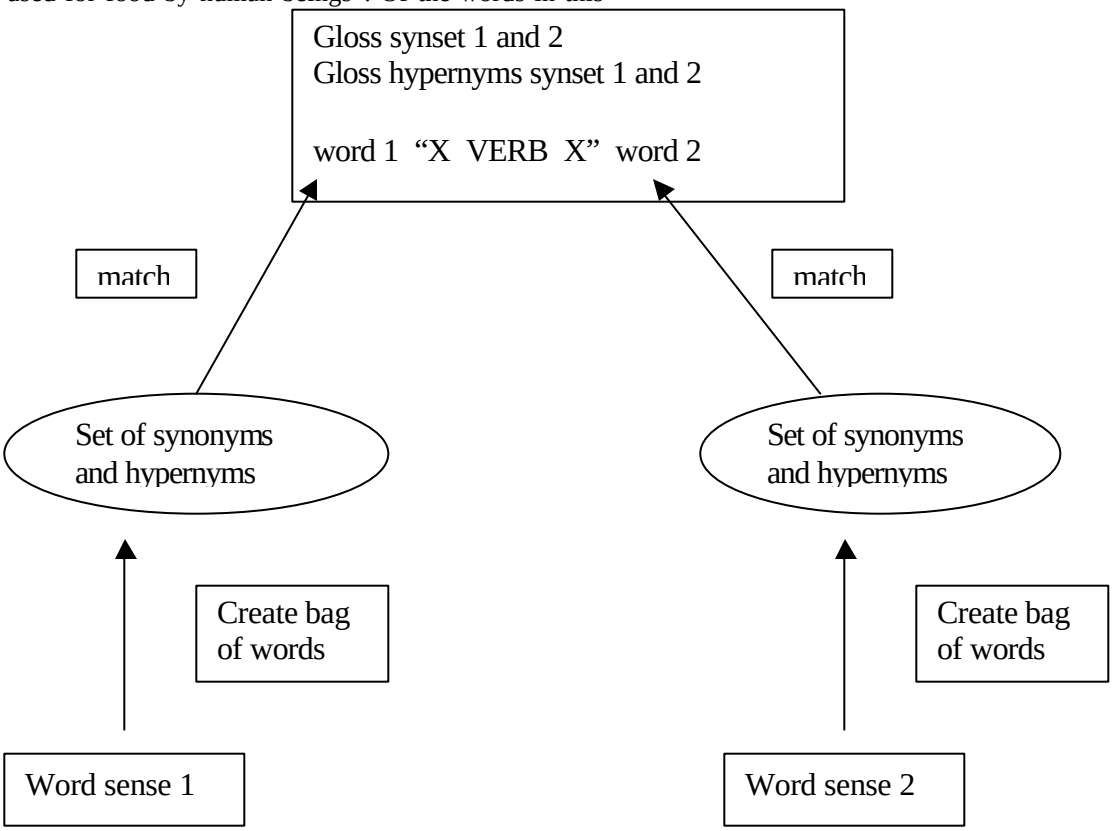


Figure 2: Mapping synset members onto glosses

The pattern **profession** and **discipline** (see figure 1) subsumes five words: *architecture, literature, politics, law* and *theology*.
Sense 6 of 'law' has the gloss 'the learned profession that is mastered by graduate study in a law school and that is responsible for the judicial system; "he studied law at Yale"'
Bag synset 1 contains 'profession', bag synset2 'study'. In between is the verb 'is mastered by' which yields the relation 'profession is mastered by discipline' for this regular polysemic pattern. This relation is found 2 times. One other relation was found: 'concerned with', which occurs only once.

Other relations are:

writing (reading matter; anything expressed in letters of the alphabet (especially when considered from the point of view of style and effect); "the writing in her novels is excellent")
message (what a communication that is about something is about)

This pattern covers 36 words. Examples are *account, conclusion, declaration, epitaph*. The relation 'express' occurs once, 'state' occurs 24 times.

fabric (something made by weaving or felting or knitting or crocheting natural or synthetic fibers)
covering (a natural object that covers or envelops)
This hypernymic combination subsumes 5 words: *fleece, hair, tapa, tappa, wool*.
'made from' occurs once.

person (a human being; "there was too much for one person to do")
language (a systematic means of communicating by the use of sounds or conventional symbols; "he taught foreign languages"; "the language introduced is standard throughout the text")
This pattern subsumes 257 words such as *Tatar, Assyrian, Hopi, Punjabi*.
The relation 'speak' occurs 132 times.

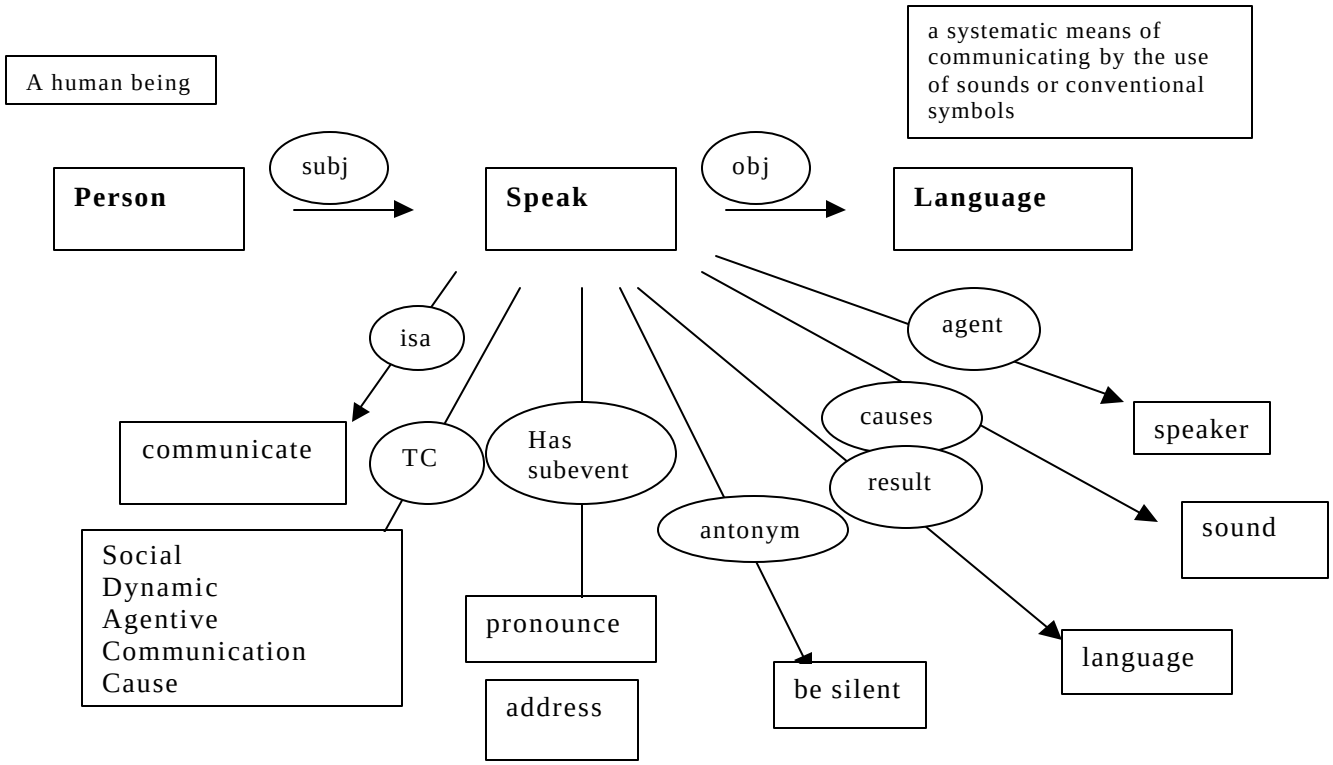


Figure 3: Expanded Ontological Fragment for the pattern person - speak - language

5 Expansion through EuroWordNet

Now we have obtained a number of patterns with specific relations it is possible to extend each ontological fragment consisting of concept triples (N-V-N) with explicit relations from EuroWordNet (Vossen, 1998). We have chosen this database over Wordnet because it contains more kinds of semantic relations than WordNet, such as thematic relations and links that hold between concepts lexicalized by different parts of speech.

First, the applicable verb senses was chosen manually. After that, relational chains in the database were extracted. Figure 3 and 4 exemplify this process for the verbs 'speak' associated with the pattern person - language and 'master' linking profession and discipline. The 'TC' relation indicates the EuroWordNet top concepts that are described in great detail in (Rodriguez et al., 1998).
The relations can all be considered additional slots in the partial knowledge frame that started as a regular polysemic pattern.
For instance, the additional knowledge fragments provided by EuroWordNet connect 'master' to 'knowledge', 'practice', 'learning' and 'teaching'. These can be used for inferencing purposes or knowledge extraction from texts.

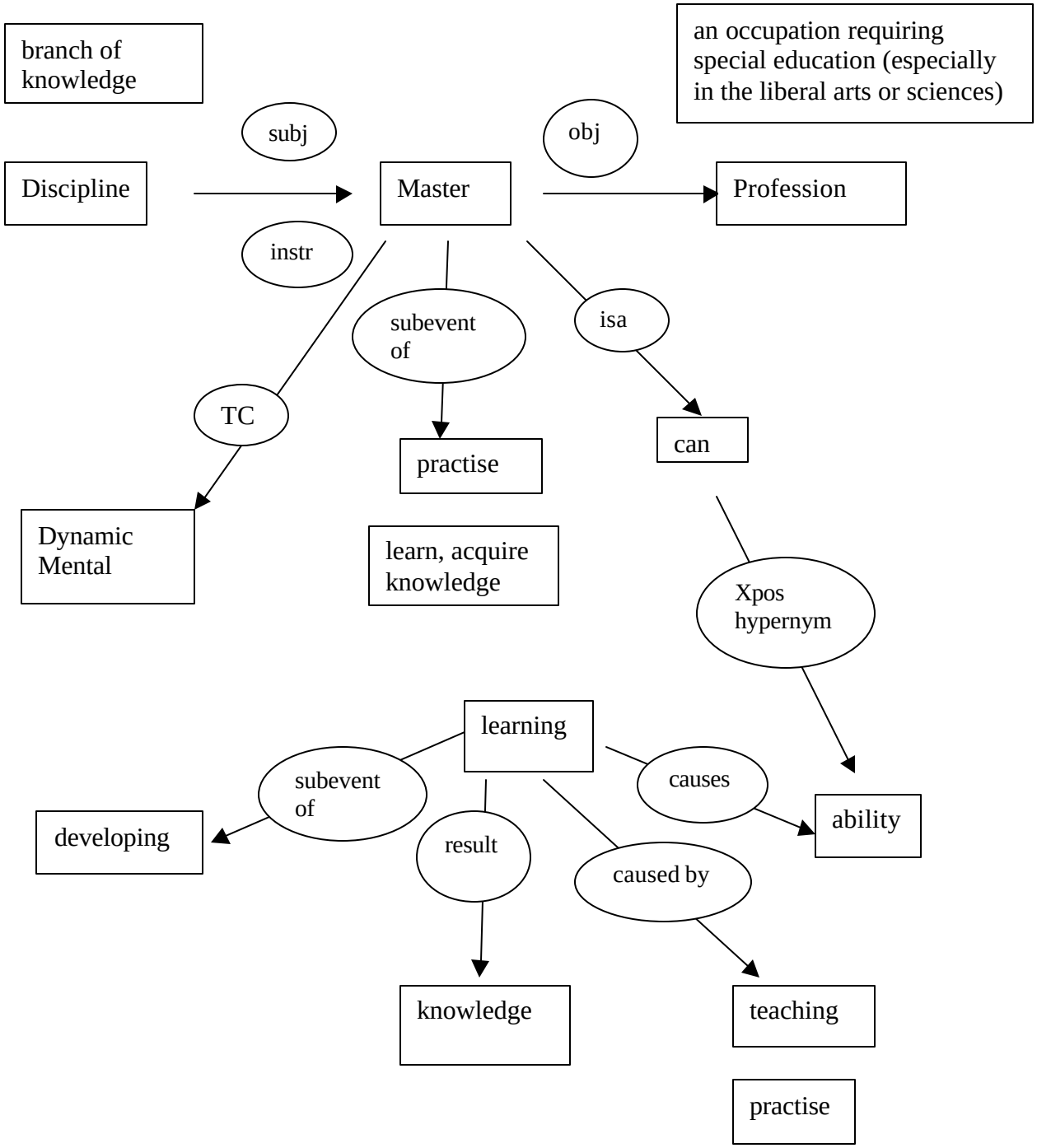


Figure 4: Expanded Ontological Fragment for the pattern **discipline - master - profession**

6 Discussion and conclusion

We have shown that the semi-automatic technique described above for extracting semantic relations between systematically related senses from WordNet glosses is feasible. There are cases, however, where no relations can be extracted, and where the extracted relations are wrong. Further experimentation with the syntactic properties of the glosses might improve results.

References

Apresjan, J. (1973), *Regular Polysemy*
In: Linguistics 142: 5-32

Fellbaum, Christiane (ed.) (1998)
WordNet: An Electronic Lexical Database.
Cambridge, Mass.: MIT Press.

Fillmore, C (1977)
Scenes and frames semantics.
In: Zampolli, A (ed.) Linguistic structures processing.
Amsterdam: Benjamins, 55-81.

Harabagiu, S. and Maiorano, S. (2000), *Acquisition of Linguistic Patterns for Knowledge-based Information Extraction*
LREC 2000, Athens

Peters, W. and Peters, I. (2000), *Lexicalised Systematic Polysemy in WordNet*
In *Proc. Secondt Intnl Conf on Language Resources and Evaluation*
Athens, Greece

Peters, W. and Wilks, Y. (2001), *Distribution-oriented Extension of WordNet's Ontological Framework,*
Proceedings RANLP2001, Tzigov Chark, Bulgaria

Peters, W., Guthrie, L. and Wilks, Y. (2002) , *Cross-linguistic Discovery of Semantic Regularity,*
Proceedings Global WordNet Association, Mysore, India

Radden, G. and Kövecses (1999), *Towards a Theory of metonymy*
In: Panther, K.U. and Radden, G. (eds.) Metonymy in language and Thought.
John Benjamins, Amsterdam

Rodriquez, H., Climent, S., Vossen, P. Bloksma, L., Roventini, A.,
Bertagna, F., Alonge, A., Peters, W. (1998), *The Top-Down Strategy for Building EuroWordNet: Vocabulary Coverage, Base Concepts and Top Ontology.*
In: Nancy Ide, Daniel Greenstein, Piek Vossen (eds), Special Issue on
EuroWordNet. Computers and the Humanities, Volume 32, Nos.
2-3 1998. 117-152.

Vossen, P. (1998), *Introduction to EuroWordNet.*
In: Nancy Ide, N., Greenstein, D. and Vossen, P. (eds), Special Issue on EuroWordNet. Computers and the Humanities, Volume 32, Nos. 2-3 1998. 73-89.

Rerendering Semantic Ontologies: Automatic Extensions to UMLS through Corpus Analytics

James Pustejovsky, Anna Rumshisky, José Castaño

Department of Computer Science, Brandeis University
Waltham, MA 02454
{jamesp, arum, jcastano}@cs.brandeis.edu

Abstract

In this paper, we discuss the utility and deficiencies of existing ontology resources for a number of language processing applications. We describe a technique for increasing the semantic type coverage of a specific ontology, the National Library of Medicine's UMLS, with the use of robust finite state methods used in conjunction with large-scale corpus analytics of the domain corpus. We call this technique "semantic rerendering" of the ontology. This research has been done in the context of Medstract, a joint Brandeis-Tufts effort aimed at developing tools for analyzing biomedical language (i.e., Medline), as well as creating targeted databases of bio-entities, biological relations, and pathway data for biological researchers (Pustejovsky et al., 2002). Motivating the current research is the need to have robust and reliable semantic typing of syntactic elements in the Medline corpus, in order to improve the overall performance of the information extraction applications mentioned above.

1. Introduction

Data mining and information extraction rely on a number of natural language tasks that require semantic typing; that is, the ability of an application to accurately determine the conceptual categories of syntactic constituents. Accurate semantic typing serves tasks such as relation extraction by improving anaphora resolution and entity identification. Domain-specific semantic typing also benefits statistical categorization and disambiguation techniques that require generalizations across semantic classes to make up for the sparsity of data. This applies, for example, to the problem of prepositional attachment, as well as identification of semantic relations between constituents within nominal compounds (see, for example, related discussion in Rosario & Hearst (2002)). Semantic typing has other direct applications, such as query reformulation, the filtering of results according to semantic type restrictions, and so on.

The set of categories used in semantic typing must be adequate enough to serve such tasks. In the biomedical domain, there are a number of efforts to develop specialized taxonomies and knowledge bases (UMLS, Gene Ontology, SWISS-PROT, OMIM, DIP). In this paper, we describe a method for adapting existing ontology resources for the natural language processing tasks and illustrate this technique on the National Library of Medicine's UMLS.

The UMLS, like many industry-standard taxonomies, contains a large number of word-concept pairings (over 1.5M typed terms), making it potentially attractive as a resource for semantic tagging information. However, these types are inadequate for NL tasks for two major reasons. First, the overall type structure is very shallow. For example, for the semantic tag "Amino Acid, Peptide, or Protein" (henceforth AAPP), there are 180,998 entries, for which there are dozens of functional subtypes that are routinely distinguished by biologists, but not in the UMLS.

One specific example of the type system deficiencies illustrates this point very clearly: the extraction of relations and their arguments from text is greatly improved with entity and anaphora resolution capabilities. However, entity and event anaphora resolution rely on (among other things) the semantic typing of the anaphor and its potential antecedents, particularly with sortal and event anaphora, as shown in (1) below.

(1) a. "For separation of nonpolar compounds, the prerun can be performed with *hexane*$_i$; ... The selection of *this solvent*$_i$ might be considered .."

b. [*p21*$_i$ inhibits the regulation of ...] ... [*This inhibitor*$_i$ binds to ...]

c. [A *phosphorylates*$_i$ B.] ... [*The phosphorylation*$_i$ of B ...]

Strict UMLS typing presents a problem for our anaphora resolution algorithm (Castaño et al., 2002). For example, for the case of anaphora in (1a), the UMLS Metathesaurus types *hexane* as either 'Organic Chemical' or 'Hazardous or Poisonous Substance'. However, *solvent* is typed as 'Indicator, Reagent, or Diagnostic Aid'. In the UMLS Semantic Network, these semantic types are not related. Therefore the resolution of the sortal anaphora would fail, due to the type mismatch. The fact is that hexane is a solvent, and this is simply not reflected in UMLS.

Functional subtyping is also missing, as (1b) illustrates. This example shows a known protein (p21) being subsequently referred to as an 'inhibitor' (a functional class of proteins). This type does not exist in UMLS and the noun 'inhibitor' is merely typed as 'Chemical Viewed Functionally', while p21 itself is typed as 'Gene or Genome', AAPP, or 'Biologically Active Substance'. It is therefore difficult to discriminate p21 from other proteins (as potential antecedents) for the sortal anaphor "this inhibitor".

A related difficulty is encountered with event anaphora cases such as (1c), where an event nominal anaphor binds to a tensed event as its antecedent, both of which are of different types in the UMLS. Hence, the existing UMLS system does not allow for recognition of type-subtype relations of the kinds that are needed in order to identify anaphoric bindings in Medline texts.

Given these motivations, we have developed a set of techniques for "rerendering" an existing semantic ontology to satisfy the requirements of specific features of a (set of) application(s). For the present case (i.e., the UMLS and bio-entity and relation extraction), we identify candidate subtypes for inclusion in the type system by two means: (a) corpus analysis on compound nominal phrases that express unique functional behavior of the compound head; (b) identification of functionally defined subtypes derived from bio-relation parsing and extraction from the corpus. The results of rerendering are evaluated for correctness against the original type system, and against additional taxonomies, should they exist, such as the GO ontology. In our preliminary experiments, we had domain experts partially verify it aganst the Gene Ontology. Full automatic verification will be done in the future.

2. Semantic Rerendering

Many NLP tasks in the service of information extraction can benefit from more accurate semantic typing of the syntactic constituents in the text. As mentioned above, the semantic taxonomy available from UMLS is lacking in several respects. With specific applications such as content summarization, anaphora resolution, and accurate relation identification in mind, we describe how an existing type system can be systematically adapted to better serve these needs, using a technique we call *semantic rerendering*. Semantic rerendering is a process that takes as input an existing type system (such as UMLS) and a text corpus, and proposes refinements to the taxonomy on the basis of two strategies:

- *Linguistic Rerendering*: Syntactic and semantic analysis of NP structures in the text;
- *Database Rerendering*: Analysis of "ad hoc abstractions" from a database of relations automatically derived from the corpus.

In the first strategy, we use the syntax of noun groups to identify candidate subtypes to an existing UMLS type. For example, categories that are of interest to biologists but which are not explicitly represented in the type system are functional categories such as *phosphorylators*, *receptors*, and *inhibitors*. These are each significant categories in their own right but also have a rich number of subtypes as well, as illustrated in (2) below.

(2)
```
CB(2) receptor
cannabinoid receptor
cell receptor
D1 dopamine receptor
epidermal growth factor receptor
functional GABAB receptor
gastrin receptororphan receptor
orphan nuclear receptor
major fibronectin receptor
mammalian skeletal muscle acetylcholine receptor
normal receptor
PTHrP receptor
protein-coupled receptor
ryanodine receptor
```

If individual proteins can be identified (i.e., semantically tagged) as belonging to a functionally defined class, such as *receptor*, then richer information extraction and textual binding is enabled.

There has been some recent research on extracting hyponym and other relations from corpora (Hearst, 1992; Pustejovsky et al., 1997; Campbell & Johnson, 1999; Mani, 2002). Our work extends the techniques described in (Pustejovsky et al., 1997) using more extensive corpus analytic techniques as developed in Pustejovsky & Hanks (2001).

2.1. Linguistic Rerendering

We first describe the linguistic rerendering procedure for inducing subtypes from corpus data, given an existing taxonomy such as the UMLS. We being by taking the strings classified as <*supertype*> in the current type system. On the basis of their behavior in the corpus, we identify candidate subtypes, derived from an analysis of the structure of nominal compounds and clusters. We use the RHHR (*righthand head rule*, cf. Pustejovsky et al. (1997)) for compound nominals (CN) and create subtype <*head-of-CN*> from the type of the head of CN. We then create a node for type N' and insert it into the existing UMLS hierarchy.

More explicitly, the procedure for identifying candidate subtypes from the structure of nominal compounds is given below.

(1) Acquire corpus C.

(2) Apply existing type system *UMLS*$_1$ over C:
TS-UMLS$_1(C) = C_{S-Tag}$.

(3) Select from the resulting semantically tagged corpus C_{S-Tag} all *NPs* with semantic tag A with $\theta > \delta$, where θ is a measure of how interesting semantic type is for rerendering:

(4) For a given noun *N* that is the headword of a phrase with semantic tag A, propose *N* as *name* of a subtype of *S-Tag A*, $N' \sqsubseteq A$, if:

- N appears as head in a certain number of NPs of length $l \geq 2$;
- N falls under the threshold set for the headwords above, but is an LCS (longest common subsequence) of a number of syntactic heads that achieve it when combined [1];
- there is sufficient variation in words comprising the remainder of phrase (so as to exclude using collocations as subtypes).

(We will refer to the nodes inserted into the ontology at this stage *first-level extension*)

(5) Nouns in the residue of *NP* with *N* as head α as modifier can be proposed as subtypes of $\alpha N' \sqsubseteq N'$ (*second-level extension*).

[1] E.g. For AAPP, *oxidase* might not achieve the threshold by itself. However, it does when all headwords containing it as a subsequence are combined (i.e.*myeloperoxidase, peroxidase, deepoxidase,* etc.)

Further subcategorization of induced types, based on the analysis of modifiers within the nominal phrases, uses a combination of template filtering of noun phrases and the LCS (longest common subsequence) algorithm (Cormen et al., 1990). Notice that one must use different thresholds for headwords and modifiers (in step (4) or step (5) of the algorithm). However, at step (4), a candidate subtype may replicate exactly the parent node ($receptor \sqsubseteq Receptor$). In that case, first-level extension types must derived from subphrase analysis, but using the threshold established for step (4).

Once the candidate subcategories are selected, the next step is to obtain the instances for the induced subtypes. These instances and their type bindings can be identified from the corpus using a number of standard methodologies developed in the field for the expansion of ontology coverage (Hearst, 1992; Campbell & Johnson, 1999; Mani, 2002). For the moment, in the experiments we conducted, we used syntactic pattern templates to identify the strings that map to the proposed extensions of UMLS types (see examples in Table 1 below).

This procedure will result in differential depth of UMLS extension for functionally defined vs. naming categories. For example no strings should map to $\{head, neck, arm, leg\} \sqsubseteq$ *<Body Location or Region>*, while string mappings are easily obtained for relational nouns such as $\{solvent, antibody, conjugate\} \sqsubseteq$ *<Indicator, Reagent, or Diagnostic Aid>*.

2.2. Database Rerendering

The second strategy uses a database of biological relations constructed through the application of robust natural language techniques as outlined in Pustejovsky et al. (2002) and Castaño et al. (2002). Over this database, "ad hoc" categories are created by projecting statistically thresholded arguments. More formally, for a particular relation, a typed projection is obtained:

$\pi X = \{X : T_1 | R(X, Y) \wedge T_1 \in UMLS_1\}$

R	X	Y
phosphorylate	"TNIK"	"Gelsolin"
phosphorylate	"GSK-3"	"NF-ATc4"
phosphorylate	"IKK-beta"	"IkappaB"
...	...	...
inhibit	"PD-ECGF"	"DNA synthesis"
inhibit	"BMP-7"	"terminal chondrocyte differentiation"
block	"DFMO"	"ODC activity"
abrogate	"Interleukin-4"	"hydrocortisone-induced apoptosis"
...	...	...

Table 2: A sample segment of relations database

The noun forms for such ad hoc categories are determined by checking each relation against the first-level extension subtypes derived through NP structure analysis as outlined above. Thus,

- For relation R and each subtype $N' \sqsubseteq T_1$, associate N' with πX if $Sim(N, \pi R) > \epsilon$.

e.g. $Sim($ "kinase", "phosphorylate" $)$,
$Sim($ "inhibitor", "inhibit" $)$, etc.

Note that the ad hoc category created through projection of the relation's argument can be matched with the types obtained at the second-level of NP-based ontology extension.

The similarity measure is constructed as a weighted combination of string similarity (e.g. *LCS*-based score), and an integrated composite measure derived from the training corpus and the outside knowledge sources. The latter might use standard IR similarity measures on contexts of occurrence of R and N in Medline abstracts, in definitions of R and N in domain-specific MRDs (such as the On-line Medical Dictionary), etc. Thus, we have:

$$Sim(N, \pi R) =$$
$$= z_0 * \text{ LCS-score } (N, \pi R) + \Sigma_{i=1}^{k} z_i * Sim_i(N, \pi R)$$

where $Sim_i(N, \pi R)$ is the similarity score derived from the source i, and z_i is the weight assigned to the source i.

3. Methodology

3.1. Seed Ontology

The Unified Medical Language System (UMLS) which was used as the seed ontology has three components: the UMLS Metathesaurus, the UMLS Semantic Network, and the SPECIALIST Lexicon (UMLS Knowledge Sources, 2001). The UMLS Metathesaurus maps single lexical items and complex nominal phrases into unique concept IDs (CUIs) which are then mapped to the semantic types from the UMLS Semantic Network. The latter type taxonomy is what was used in the experimental applications of rerendering procedure. It consists of 134 semantic types hierarchically arranged via the 'isa' relation and interlinked by a set of secondary non-hierarchical relations. UMLS Metathesaurus in the UMLS 2001 distribution contains over 1.5 million string mappings.

In the Metathesaurus, multiple semantic type bindings are specified for many of the concepts. Due to this ambiguity of UMLS concepts and to a lesser extent, the ambiguity of the strings themselves, the mappings obtained from the Metathesaurus give a number of semantic types for each lexical item or phrase. We intentionally avoid superimposing any disambiguation mechanism on this typing information while applying it in corpus analysis. Since corpus-based derivation of subtypes uses a frequency cutoff, this ambiguity essentially resolves itself. For example, if a given lexical item is typed as both T_1 and T_2 in the seed ontology, and occurs as a headword in $> 1\%$ of nominal phrases typed as T_1, but in $< 1\%$ of nominal phrases typed as T_2, it will only be proposed as a candidate subtype of T_1. Thus, under the 1% cutoff, *isozyme*, which the seed UMLS types as either $Enzyme$ or $AAPP$, will only be identified as a good candidate subtype for *Enzyme*.

3.2. Corpus preprocessing with UMLS types

The experimental application of the rerendering procedure was conducted on a relatively small corpus of Med-

Pattern Type	TEMPLATE	
apposition	"X, a Y inhibitor"	"X, the solvent
	"X, an inhibitor of Y"	"the solvent, X"
	"X, an inhibitor of Y"	"X, a common solvent for Y"
nominal compounds	"Y inhibitor"	"the solvent X"
definitional constructions	"X is an inhibitor of Y"	
aliasing constructions	"X (inhibitor of Y)"	"X (the solvent)"
	"an inhibitor of Y (X)"	"the solvent (X)"
enumeration	"Y inhibitors such as X, ..."	"solvents (e.g. X)"
		"solvents, e.g. X"
		"the following solvents: X, .."
relative clauses	"X which is an inhibitor of Y"	"the solvent used was X"
		"X proved to be a suitable solvent"
adjuncts		"in X and Y as solvents"
		"X as solvent"

Table 1: Sample syntactic patterns for string-to-semantic type mappings

line abstracts (around 40,000). Medline abstracts were tokenized, stemmed, and tagged. They were then shallow-parsed, with noun phrase coordination and limited prepositional attachment (only *of*-attachment) using finite-state techniques. The shallow parse was obtained using five separate automata each recognizing a distinct family of grammatical constructions, very much in the spirit of Hindle (1983), McDonald (1992) and Pustejovsky et al. (1997). The machinery used in preprocessing is described in more detail in Pustejovsky et al. (2002).

Semantic type assignment of the resulting nominal chunks is determined through lookup as follows. Each noun phrase is put through a cascade of hierarchically arranged type-assignment heuristics. Higher priority heuristics take absolute precedence; that is, if a semantic typing is possible, it is assigned. In this implementation, we use the full UMLS semantic type hierarchy, including the mappings to both leaves and intermediate nodes.

During direct lookup, a string is assigned a given semantic type if the UMLS Metathesaurus contains a mapping of that string to a concept so typed. If a semantic type for the whole phrase is not found in UMLS, we attempt to identify its syntactic head using a modification of RHHR (*righthand head rule*), and determine the semantic type of the headword. For chunks with *OF*-attachment, i.e. phrases of the form, *<NP-1> of <NP-2>*, the lookup is first attempted on *NP-1* as a whole.

If the lookup on a particular prospective head fails, it is tested for a match with morphological heuristics recognizing semantically vacant categories, such as 'NUMERIC', 'ABBREVIATION', 'SINGLE CAPITAL LETTER', 'SINGLE LOWER-CASE LETTER', etc. These, and phrases headed by common words occurring in a non-specialized dictionary are filtered out. The last layer of heuristics applied to a prospective syntactic head successively attempts to strip a groups of suffixes and prefixes and perform lookup on the remaining stem.

3.3. Inducing candidate subtypes

In these initial series of tests, we experimented primarily with the first part of the rerendering procedure as it is outlined in Section 2.1. In the first stage of identifying the subtypes based on the syntactic analysis of noun phrase structure, a headword was considered a candidate subtype of type T if it occured in more that 1% of all nominal chunks tagged as T. Note that the same chunk is frequently tagged with several UMLS types.

The candidate subtypes for the second (NP modifier-based) level of UMLS extension were identified using a combination of template and frequency-based filtering of noun phrases and the LCS (longest common subsequence) algorithm. Thus, for a given headword proposed as subtype at first level of extension (e.g., *kinase*) the LCS algorithm was run on all phrases with that headword that matched a certain template (e.g. <Indefinite Article> <Modifier>* N). The substrings that occurred in the corpus in more than a certain percentage of phrases with that headword were identified as candidate subtypes for insertion into the ontology at the next level. The cut-off threshold had to be kept very low for this series of experiments, as it was conducted over a relatively small corpus. In working with a larger corpus the thresholds are set separately for each template, so e.g. it is much higher for the unfiltered set of nominal compounds than for those occurring with an indefinite article. Frequency-based filtering involves discarding as potential candidates noun phrases with modifiers that occur frequently in separate non-specialized corpus, which allows to automatically discard phrases such as 'multiple receptors', 'specific kinase', etc. [2]

Identification of sample instances for the induced types was performed over shallow-parsed text using syntactic pattern templates. The definitional construction patterns were extracted using relation extraction machinery (see Pustejovsky et al. (2002) for details). It was applied to our test corpus and another sample set of Medline abstracts (approx. 60,000).

4. Results

Semantic typing over our sample set of Medline data produced type bindings for over 1 million noun phrases.

[2] Similar filtering was also applied to the first-level extensions

4.1. NP analysis-based subtypes

The choice of particular UMLS categories as supertypes for extension of the seed UMLS semantic type taxonomy is dictated by the particular natural language application. Semantic types given below are derived from nominal phrase analysis for some of the supertypes that have been used in anaphora resolution tasks (cf Castaño et al. (2002)). Each UMLS type is shown with the number of noun phrases of that type which occurred in our test corpus, followed by a list of derived candidate subtypes with their respective frequencies. The subtypes shown below were derived as described above in step 4 of the rerendering procedure specification in Section 2.1.

```
Enzyme 4724
      dehydrogenase 140
      protease 160
      reductase 73
      metalloproteinase 48
      isozyme 54
      oxidase 79
      phosphatase 111
      enzyme 1142
      kinase 741

Amino Acid, Peptide, or Protein 20830
      receptor 2444
      protein 4521
      peptide 947
      kinase 741
      cytokine 287
      isoform 412

Cell 16348
      macrophage 251
      clone 350
      neuron 1094
      lymphocyte 412
      fibroblast 257
      cell 11586

Cell Component 2508
      cytosol 84
      nucleus 469
      liposome 43
      organelle 40
      vacuole 35
      ribosome 28
      cytoskeleton 55
      dendrite 53
      cytoplasm 195
      soma 26
      granule 80
      chromatin 36
      microtubule 45
      chromosome 319
      axon 99
      microsome 132
```

Notice that the categories derived in this manner would include functionally defined types (e.g. *isoform*).

4.2. NP modifier-based extension (second-level)

As mentioned above, some of the UMLS extension candidates that are derived according to the procedure are replicas of the supertype category, e.g. $enzyme \sqsubseteq Enzyme$, or $receptor \sqsubseteq Receptor$. For example, among the lexical items tagged as *Receptor* in UMLS Metathesaurus, NPs headed by the word "receptor" comprise 87% of all NPs tagged as *Receptor* in our test corpus:

```
Receptor 2820
      integrin 91
      receptor 2444
```

The appropriate extensions to the comparable level within the type taxonomy in this case are derived from subphrase analysis. Thus, for the case of $enzyme$, the candidate subtypes so derived would be:

```
cytosolic enzyme
heterologous enzyme
male enzyme
metalloenzyme
multifunctional enzyme
proof-reading enzyme
proteolytic enzyme
rate-limiting enzyme
recombinant enzyme
rotary enzyme
tetrameric enzyme
```

These are identified at step 5 of rerendering procedure through a combination of template filtering of noun phrases and longest common substring identification. They are then added to the same level of the type taxonomy as all $N' \sqsubseteq Enzyme$ (see Figure 1).

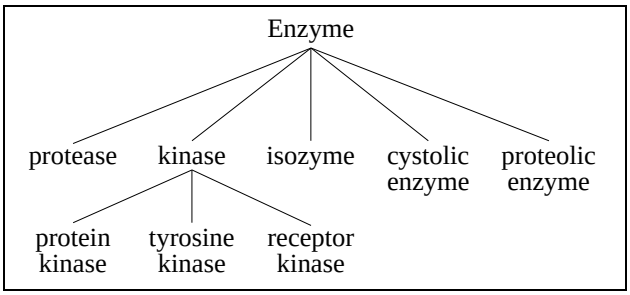


Figure 1: Extension subtree for $Enzyme$ (partial)

The results produced at this stage by the automated processing described above need further filtering before good subtype candidates can be identified. This can be achieved by fine-tuning the use of corpus frequencies, as well as type filtering of modifiers using the seed ontology type system. Table 3 below shows UMLS types for selected NP modifier-based subcategories of $receptor$.

4.3. Corpus-based identification of the instances of induced semantic categories

The rerendering procedure gives different results for different segments of the taxonomy, depending on the class of supertype category. Thus, for functionally defined semantic types, such as, "Chemical Viewed Functionally", or "Indicator, Reagent, or Diagnostic Aid", corpus-based derivation of instances for the induced subcategories is clearly much more feasible. Consider the first level extension types for the categories below:

```
Indicator, Reagent, or Diagnostic Aid 3424
      buffer 151
      conjugate 112
      stain 75
      agar 38
      antibody 1640
      indicator 373
      solvent 38
      tracer 53
      dye 95
      reagent 113
      nitroprusside 51
      hydrogen peroxide 58
```

Candidate Subtypes $\alpha N' \sqsubseteq N'$	**Seed UMLS Type for Modifier** α
cell surface receptor	'Cell Component'
membrane receptor	'Tissue'
adhesion receptor	'Acquired Abnormality', 'Disease or Syndrome'
	'Natural Phenomenon or Process'
activation receptor	no type binding
contraction receptor	'Functional Concept'
estrogen receptor	'Steroid', 'Pharmacologic Substance', 'Hormone'
dopamine receptor	'Organic Chemical', 'Pharmacologic Substance',
	'Neuroreactive Substance or Biogenic Amine'
adenosine receptors	'Nucleic Acid, Nucleoside, or Nucleotide',
	'Pharmacologic Substance', 'Biologically Active Substance'
insulin receptor	'Amino Acid, Peptide, or Protein',
	'Pharmacologic Substance', 'Hormone'
TSH receptor	'Amino Acid, Peptide, or Protein', 'Hormone'
	'Neuroreactive Substance or Biogenic Amine'
EGF receptor	'Amino Acid, Peptide, or Protein', 'Hormone',
	'Pharmacologic Substance',
transferrin receptor	'Amino Acid, Peptide, or Protein', 'Biologically Active Substance',
	'Indicator, Reagent, or Diagnostic Aid', 'Laboratory Procedure'
receptor	**'Amino Acid, Peptide, or Protein', 'Receptor'**

Table 3: UMLS Typing of modifiers α for some sample subtypes $\alpha N' \sqsubseteq N'$ for N' =*receptor*

```
Chemical Viewed Functionally 3494
      inhibitor 1668
      prodrug 62
      basis 1075
      vehicle 107
      radical 144
      base 265
      pigment 36
      surfactant 36

Pathologic Function 17752
      impairment 383
      stenosis 274
      other 450
      illness 209
      problem 1133
      dysfunction 493
      block 244
      carrier 219
      inflammation 243
      pathogenesis 497
      cavity 273
      hemorrhage 180
      occlusion 266
      lesion 1820
      infarction 449
      regression 237
      pathology 242
      infection 1782
      complication 1248
      separation 320
      degeneration 180
      stress 487
```

Table 4 shows the derivation of instances for the categories induced through noun phrase analysis (step 5), using the definitional construction template. The first column shows the actual strings that get the new type binding as $kinase$ (in blue) and their original UMLS types (in black). Notice that for many of the strings that can be so typed, the seed UMLS type is either generic $AAPP$ or the type binding is absent altogether.

If the candidate subtype is a valid semantic category, such corpus-based identification of instances should work equally well irrespective of the level at at which the induced type is inserted. For example, see Table 5 below for NP modifier extensions of *receptor*.

cell-surface receptors:
polycystin-1 is a *cell surface receptor*
Fas is a *cell surface* death $receptor$
CD40 is a *cell surface receptor*
CD44 is a *cell surface receptor*
The scavenger receptor BI is a *cell surface* lipoprotein $receptor$
membrane receptors:
Neuropilin-1 is a trans*membrane receptor*
APJ is a seven trans$membrane$ domain G-protein-coupled $receptor$
HER2 is a *membrane receptor*

Table 5: Sample semantic type instances derived with the definitional construction template for subtypes of *receptor*

5. Evaluation of Rerendering Procedure

The evaluation of the performance for rerendering essentially boils down to whatever improvement is produced in precision and recall for the client applications. However, in order to do an earnest evaluation of performance of the rerendering algorithm, we would need to run it on a much larger corpus. This would allow for better candidate choices for the portions of the procedure that have been plagued by sparsity (e.g., in NP modifier-based candidate subtype selection). But most importantly, it would increase the coverage in terms of instances for which the type bindings are produced in the new type system.

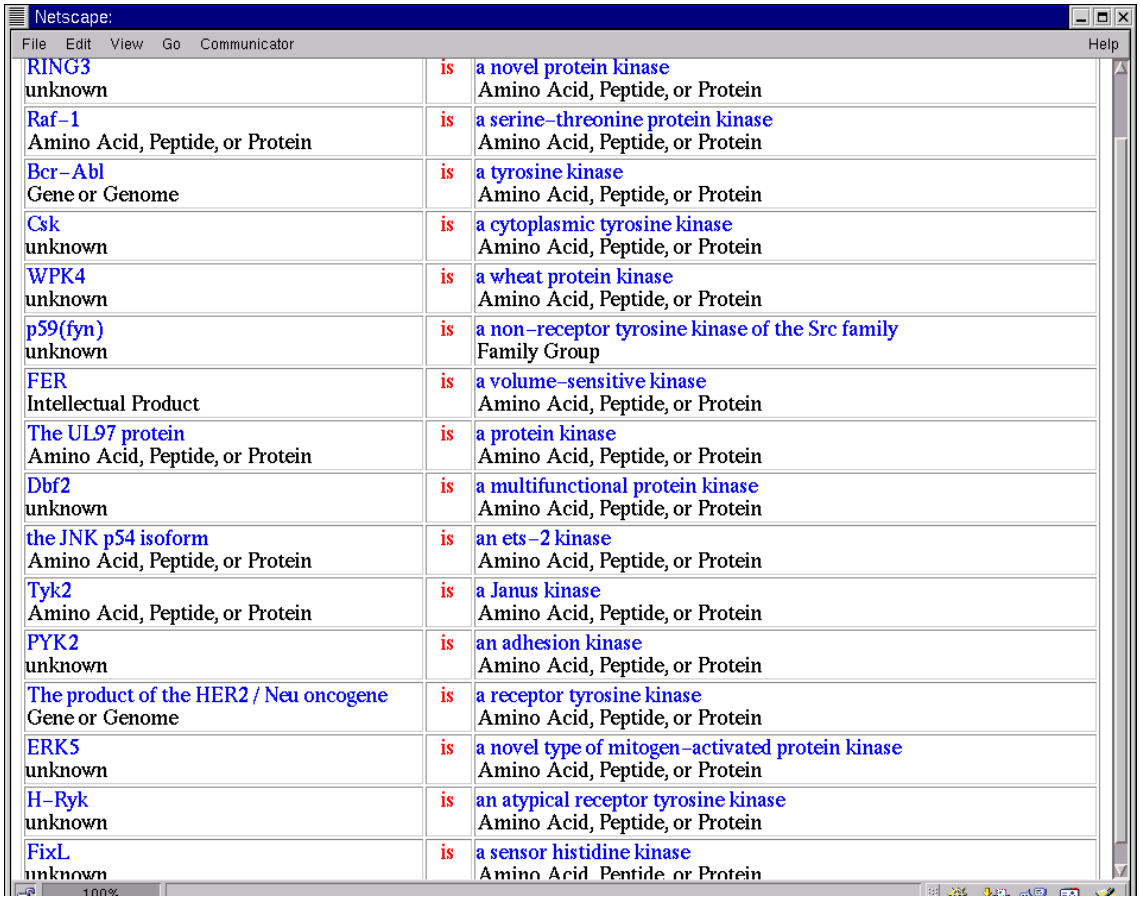


Table 4: Definitional construction template at work for the $N' = kinase$

5.1. Usability in natural language applications

One of the client applications for the experiments we report here is coreference resolution. The anaphora examples in (3) below illustrate the impact of using the derived types. Even the test corpus we used actually contained enough information to produce the type bindings for some of the strings used in (3).

(3) a. "Assays were conducted in *chloroform, toluene, amyl acetate, isopropyl ether, and butanol*. ... In *each solvent*,"

b. "The extracts were prepared separately in *methanol, ethanol, phosphate buffer saline (PBS), and distilled water* as part of our study to look at ... Our results have shown that *all four solvents* were ..."

c. "A 47-year-old man was found dead in a factory where *dichloromethane (DCM)* tanks were stocked. He was making an inventory of t he annual stock of DCM contained in several tanks (5- to 8000-L capacity) by transferring *the solvent* into an additional tank with the help of compressed air."

(emphasis added)

The seed ontology induces a type mismatch between the anaphor and the antecedent. For example, in (3c), the original type bindings are:

- *TS-UMLS*$_1$(solvent)= 'Indicator, Reagent, or Diagnostic Aid';
- *TS-UMLS*$_1$(dichloromethane)= {'Organic Chemical', 'Pharmacologic Substance', 'Injury or Poisoning'}

The rerendered ontology allows the induced semantic type *solvent* $\sqsubseteq$*<Indicator, Reagent, or Diagnostic Aid>* to be included in the type bindings for "dicloromethane".

5.2. Evaluation against existing ontologies

We performed some test evaluations of the second-level extension subtypes against the Gene Ontology. Despite the very modest side of our test corpus, we observed significant overlap in some categories. Thus, for example, the 388 second-level extension subtype candidates for *receptor*, 12% were identified as concept names in the Gene Ontology.

In general, the preliminary results of applying the first step of the rerendering procedure algorithm to the UMLS semantic type taxonomy appear quite encouraging. In the future, better automated methods for the evaluation of rerendering results against the existing ontologies must be developed. And most importantly, the utility and usefulness of the rerendering algorithm must be evaluated vis-a-vis achieving improvement in precision and recall for client NLP applications.

6. References

D. A. Campbell and S. B. Johnson. 1998. A Technique for Semantic Classification of Unknown Words Using UMLS Resources. In *Proceedings of AMIA Fall Symposium.*

S. A. Caraballo. 1999. Automatic construction of a hypernym-labeled noun hierarchy from text. In *Proceedings of 37th Annual Meeting of the Association for Computational Linguistics*.

J. Castaño, J. Zhang, and J. Pustejovsky. 2002. Anaphora Resolution in Biomedical Literature. *submitted to International Symposium on Reference Resolution 2002, Alicante Spain*.

T.H. Cormen, C.E. Leiserson, and R.L. Rivest. 1990. Introduction to Algorithms. *MIT press, Cambridge, MA, 1990*.

U. Hahn and S. Schulz 2002. Massive Bio-Ontology Engineering in NLP In *Proceedings of Human Language Technology Conference.*

M. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In *Proceedings of 14th International Conference on Computational Linguistics.*

D. Hindle. 1983. Deterministic Parsing of Syntactic nonfluencies. *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics.*

B. L. Humphreys, D. A. B. Lindberg, H. M. Schoolman, and G. O. Barnett. 1998. The Unified Medical Language System: An informatics research collaboration. *Journal of the American Medical Informatics Association*.

I. Mani. 2002. Automatically Inducing Ontologies from Corpora. *submitted to ?*.

A. McCray, A. Burgun, and O. Bodenreider. 2001. Aggregating UMLS Semantic Types for Reducing Conceptual Complexity. In *Proceedings of Medinfo, London*.

D. D. McDonald. 1992. Robust Partial Parsing through incremental multi-algorithm processing. In *Text-based Intelligent Systems*, P. Jacobs, ed. 1992.

On-line Medical Dictionary. 1998-2002. http://cancerweb.ncl.ac.uk/omd/ *Academic Medical Publishing & CancerWEB*

D. M. Pisanelli, A. Gangemi, and G. Steve. 1998. An Ontological Analysis of the UMLS Metathesaurus Journal of American Medical Informatics Association, vol. 5 S4, pp. 810-814.

J. Pustejovsky, B. Boguraev, M. Verhagen, P. Buitelaar, and M. Johnston. 1997. Semantic Indexing and Typed Hyperlinking. *AAAI Symposium on Language and the Web, Stanford, CA*

J. Pustejovsky and P. Hanks. 2001. Very Large Lexical Databases: A Tutorial Primer. Association for Computational Linguistics, Toulouse, July, 2001

J. Pustejovsky, J. Castaño, J. Zhang, B. Cochran, and M. Kotecki. 2002. Robust Relational Parsing over Biomedical Literature: Extracting Inhibit Relations. In *Pacific Symposium on Biocomputing*.

B. Rosario and M. Hearst. 2001. Classifying the semantic relations in noun compounds via a domain-specific lexical hierarchy. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing. ACL.*

UMLS Knowledge Sources. Documentation. 2001. 12th edition. U.S. National Library of Medicine.

Exploiting Sublanguage and Domain Characteristics in a Bootstrapping Approach to Lexicon and Ontology Creation

Dietmar Rösner, Manuela Kunze

Otto-von-Guericke-University Magdeburg,
Institute of Knowledge and Language Processing,
P.O.box 4120,
D-39016 Magdeburg, Germany
{roesner,makunze}@iws.cs.uni-magdeburg.de

Abstract

It is very costly to build up lexical resources and domain ontologies. Especially when confronted with a new application domain lexical gaps and a poor coverage of domain concepts are a problem for the successful exploitation of natural language document analysis systems that need and exploit such knowledge sources. In this paper we report about ongoing experiments with 'bootstrapping techniques' for lexicon and ontology creation.

1. Introduction

It is very costly to build up lexical resources and domain ontologies. Especially when confronted with a new application domain lexical gaps and a poor coverage of domain concepts are a problem for the successful exploitation of natural language document analysis systems that need and exploit such knowledge sources.

We are confronted with such a situation very often in our work with the XDOC document suite, a collection of tools created to support intelligent processing of corpora of interesting textual documents taken from domains like engineering and medicine. The XDOC document workbench is currently employed in a number of applications. These include:

- knowledge acquisition from technical documentation about casting technology,
- extraction of company profiles from WWW pages,
- analysis of autopsy protocols.

The latter application is part of a joint project with the institute for forensic medicine of our university. The paper is organised as follows: We start with background information about XDOC. Then we sketch characteristics of the sublanguage of autopsy protocols and describe the core idea of our experiments. This is followed by a description of syntactic structures that are currently processed. Then clustering of co-occurrence data and its exploitation is described. A discussion of results and problems and an outlook on future work completes the paper.

2. Background: the XDOC document suite

We have designed and implemented the XDOC document suite as a workbench for the flexible processing of electronically available documents in German. The tools in the XDOC document suite(Kunze and Rösner, 2001a), (Kunze and Rösner, 2001b) can be grouped according to their function:

- preprocessing
- structure detection
- POS tagging
- syntactic parsing
- semantic analysis
- tools for the specific application: e.g. information extraction

In the semantic analysis, similar to the POS tagging, the tokens are annotated with their meaning and a classification in semantic categories like e.g. concepts and relations. For the semantic tagging we apply a semantic lexicon. This lexicon contains the semantic interpretation of a word and its case frame combined with the syntactic valence requirements. When we are confronted with a new application domain, the lexical resources must be completed with the domain specific terms. Even semantic resources with broad coverage like the semantic lexicon GermaNet for German (GermaNet-Project-Site, 2002) and Wordnet(Wordnet-Project-Site, 2002) for English, can not cover all terms of all different domains.

2.1. Design principles

The work in the XDOC project is guided by the following design principles:

- The tools shall be usable for 'realistic' documents.
  One aspect of 'realistic' documents is that they typically contain domain-specific tokens that are not directly covered by classical lexical categories (like noun, verb, ...). Those tokens are nevertheless often essential for the user of the document (e.g. an enzyme descriptor like EC 4.1.1.17 for a biochemist).
- The tools shall be as robust as possible.
  In general it can not be expected that lexicon information is available for all tokens in a document. This is not only the case for most tokens from 'nonlexical' types – like telephone numbers, enzyme names, material codes, ... –, even for lexical types there will always

be 'lexical gaps'. This may either be caused by neologisms or simply by starting to process documents from a new application domain with a new sublanguage. In the latter case lexical items will typically be missing in the lexicon ('lexical gap') and phrasal structures may not or not adequately be covered by the grammar.

- The tools shall be usable independently but shall allow for flexible combination and interoperability.
- The tools shall not only be usable by developers but as well by domain experts without linguistic training.

2.2. XML as unifying framework

We have decided to exploit XML (Bray et al., 1998) and its accompanying formalisms (e.g. XSLT (Site, 2002)) and tools (e.g. xt (Clark, 2002)) as a unifying framework. All modules in the XDOC system expect XML documents as input and deliver their results in XML format.

This decision has positive consequences for many aspects in XDOC. Take e.g. the desideratum that the tools of XDOC shall not only be usable by developers but as well by domain experts without linguistic training. Here XML and XSLT play a major role: XSL stylesheets can be exploited to allow different presentations of internal data and results for different target groups; for end users the internals are in many cases not helpful, whereas developers will need them for debugging.

2.3. Bridging lexical gaps

We do not expect extensive lexicon coding at the beginning of an XDOC application. XDOC's POS tagger and syntactic parser have therefore been augmented with a number of techniques for dealing with such 'lexical gaps'.

For POS tagging we exploit the morphology component MORPHIX (Finkler and Neumann, 1988): If a token in a German text can be morphologically analysed with MORPHIX the resulting word class categorisation is used as POS information. Note that this classification need not be unique. Since the tokens are analysed in isolation multiple analyses are often the case. Some examples: the token 'der' may either be a determiner (with a number of different combinations for the features case, number and gender) or a relative pronoun, the token 'liebe' may be either a verb or an adjective (again with different feature combinations not relevant for POS tagging).

MORPHIX's coverage can be characterised as follows: most closed class lexical items of German as well as all irregular verbs are covered. The coverage of open class lexical items is dependent on the amount of coding. The paradigms for e.g. verb conjugation and noun declination are fully covered but to be able to analyze and generate word forms their roots need to be included in the MORPHIX lexicon.

Due to lexical gaps some tokens will not get a MORPHIX analysis, at least at the beginning of an XDOC application. We then employ two techniques: We first try to make use of heuristics that are based on aspects of the tokens that can easily be detected with simple string analysis (e.g. upper-/lowercase, endings, ...) and/or exploitation of the token position relative to sentence boundaries (detected in the structure detection module). If a heuristic yields a classification the resulting POS class is added together with the name of the employed heuristic (marked as feature SRC, cf. example 1). If no heuristics are applicable we classify the token as member of the class unknown (tagged with XXX).

To keep the POS tagger fast and simple the disambiguation between multiple POS classes for a token and the derivation of a possible POS class from context for an unknown token are postponed to syntactic processing (cf. below).

3. Bootstrapping in a new domain

XDOCs most recent application is part of a joint project with the institute for forensic medicine of our university. The medical doctors there are interested in tools that help them to exploit their huge collection of several thousand autopsy protocols for their research interests. The confrontation with this corpus from a new domain has stimulated experiments with 'bootstrapping techniques' for lexicon and ontology creation.

3.1. The core idea

The core idea is the following:

When you are confronted with a new corpus from a new domain, try to find linguistic structures in the text that are easy to detect automatically and that allow to classify unknown terms in a robust manner both syntactically as well as on the knowledge level. Take the results from a run of these simple but robust heuristics as an initial version of a domain dependent lexicon and ontology. Exploit these initial resources to extend the processing to more complicated linguistic structures in order to detect and classify more terms of interest automatically.

An example: In the sublanguage of autopsy protocols (in German) a very telegrammatic style is dominant. Condensed and compact structures like the following are very frequent:

- Harnblase leer. (Urinary bladder empty.)
- Harnleiter frei. (Ureter free.)
- Nierenoberflaeche glatt. (Surface of kidney smooth.)
- Vorsteherdruese altersentsprechend. (Prostate corresponding to age.)
- ...

These structures can be abstracted syntactically as <Noun><Adjective><Fullstop> and as semantically <Anatomic-entity><Attribute-value><Fullstop>. Furthermore they are easily detectable.

In our experiments we have exploited this characteristic of the corpus extensively to automatically deduce an initial lexicon (with nouns and adjectives) and an initial ontology (with concepts for anatomic regions or organs and their respective features and values).

3.2. A sublanguage analysis of autopsy protocols

The telegrammatic style of autopsy protocolls results in a preference for 'verbless' structures. It is e.g. much more likely that a finding like 'the mouth was open' is expressed as 'Mund geoeffnet.' (mouth open) although a more verbose paraphrase like 'Der Mund ist geoeffnet.' may occur sometimes.

Another consequence of the style is a preference for noun compounds in contrast to semantically equivalent noun phrases.

When referring to a concept like 'weight of the liver' the noun compound 'Lebergewicht' is more likely than the noun phrase 'Gewicht der Leber'. This generalizes for the weight of other organs: 'Organgewicht' is more likely than the noun phrase 'Gewicht des/der X'.

The need for contextual interpretation of terms may be seen as another consequence of the style. In local context with an organ as topic generic terms like 'Gewicht' (weight) or 'Durchmesser' (diameter) have to be interpreted as referring to the object in focus, i.e. the organ.

3.3. Refinements of the initial approach

In our corpus it is very likely that a syntactic structure of the type <Noun><Adjective><Fullstop> can semantically be interpreted as <Anatomic-entity><Attribute-value><Fullstop>, but there are exceptions. An example: 'Flachschnitt unauffaellig.' Here the noun does not denote an anatomic entity, but is referring to a diagnostic procedure in autopsy. On the other hand the adjective co-occurs with anatomic entities as well.

So the initial approach needs refinement: as long as the number of exceptions of a simple pattern (here: <Noun> <Adjective> <Fullstop>) in a heuristic remains small the exceptions (here: noun 'Flachschnitt') are simply checked first before the heuristic is applied for all cases in which the exceptions are not present.

3.4. Exploitation of syntactic constraints

Pattern based analysis is a first step only. For full syntactic parsing we apply a chart parser based on context free grammar rules augmented with feature structures. The output of a robust POS tagger is used as input to parsing. The POS tagger works on token in isolation. Its output may contain:

- multiple POS classes,
- unknown classes of open world tokens and
- tokens with POS class, but without or only partial feature information.

Example 1 *unknown token classified as noun with heuristics*

```
<NP TYPE="COMPLEX" RULE="NPC3" GEN="FEM"
        NUM="PL" CAS="_">
  <NP TYPE="FULL" RULE="NP1" CAS="_"
           NUM="PL" GEN="FEM">
       <N SRC="UNG">Blutanhaftungen</N>
  </NP>
  <PP CAS="DAT">
    <PRP CAS="DAT">an</PRP>
    <NP TYPE="FULL" RULE="NP2" CAS="DAT"
           NUM="SG" GEN="FEM">
      <DETD>der</DETD>
      <N SRC="UC1">Gekroesewurzel</N>
    </NP>
  </PP>
</NP>
```

The latter case results from some heuristics in POS tagging that allow to assume e.g. the class noun for a token but do not suffice to detect its full paradigm from the token (note that there are approximately two dozen different morphosyntactic paradigms for noun declination in German).

For a given input the parser attempts to find all complete analyses that cover the input. If no such complete analysis is achievable it is attempted to combine maximal partial results into structures covering the whole input (Rösner, 2000).

A successful analysis may be based on an assumption about the word class of an initially unclassified token (tagged XXX). This is indicated in the parsing result (feature AS) and can be exploited for learning such classifications from contextual constraints. In a similar way the successful combination from known feature values from closed class items (e.g. determiners, prepositions) with underspecified features in agreement constraints allows the determination of paradigm information from successfully processed occurrences. In example 2 features of the unknown word "Mundhoehle" (mouth) could be derived from the features of the determiner within the PP (e.g. gender feminine).

Example 2 *unknown token classified as adjective and features derived through contextual constraints*

```
<NP TYPE="COMPLEX" RULE="NPC3" GEN="MAS" NUM="SG"
    CAS="NOM">
  <NP TYPE="FULL" RULE="NP3" CAS="NOM" NUM="SG"
        GEN="MAS">
    <DETI>kein</DETI>
    <XXX AS="ADJ">ungehoeriger</XXX>
    <N>Inhalt</N>
  </NP>
  <PP CAS="DAT">
    <PRP CAS="DAT">in</PRP>
    <NP TYPE="FULL" RULE="NP2" CAS="DAT" NUM="SG"
        GEN="FEM">
      <DETD>der</DETD>
      <N SRC="UC1">Mundhoehle</N>
    </NP>
  </PP>
</NP>"
```

The grammar used in syntactic parsing is organised in a modular way that allows to add or remove groups of rules. This is exploited when the sublanguage of a domain contains linguistic structures that are unusual or even ungrammatical in standard German.

Example 3 *Excerpt from syntactic analysis*

```
<PP CAS="AKK">
  <PRP CAS="AKK">auf</PRP>
  <NP TYPE="COMPLEX" RULE="NPC3" GEN="MAS" NUM="SG"
   CAS="AKK">
    <NP TYPE="FULL" RULE="NP1" CAS="AKK" NUM="SG"
    GEN="MAS">
      <N>Flachschnitt</N>
    </NP>
    <PP CAS="AKK">
      <PRP CAS="AKK">in</PRP>
      <NP TYPE="FULL" RULE="NP2" CAS="AKK" NUM="SG"
      GEN="NTR">
        <DETD>das</DETD>
        <N>Gewebe</N>
      </NP>
    </PP>
  </NP>
</PP>
```

3.5. Beyond simple patterns

At the time we work with a 'light' grammar of 40 rules. This grammar contains basic rules (for the analysis of noun phrases and preposition phrases) and specific rules, based on the patterns of the sublanguage.

We have just started to extract binary relations from completely parsed sentences. Following patterns of the sublanguage are analysed in this manner: Simple structures like: <NP> <Adjective> <Fullstop> will be analysed as <Anatomic-entity> <Attribute-value> <Fullstop>.

Example 4 *For example: 'Gehirngaenge frei.'. The Analysis returns:*

```
<RATT-V>
   <ENTITY>Gehoergaenge</ENTITY>
   <VALUE CNT="1">frei</VALUE>
</RATT-V>
```

All results of this analysis are also marked as XML structure. The attribute 'CNT' contains the number of occurences of the attribute value in context with the anatomic entity. A similar pattern is the structure <NP> 'ist|sind'[1] <Adjective>|<Verb> <Fullstop>.

Example 5 *For example: 'Gangsysteme sind frei.' or 'Augen sind geschlossen'. The Analysis returns:*

```
<RATT-V>
   <ENTITY>Gangsysteme</ENTITY>
   <VALUE CNT="1">frei</VALUE>
</RATT-V>
<RATT-V>
   <ENTITY>Augen</ENTITY>
   <VALUE CNT="1">geschlossen</VALUE>
</RATT-V>
```

Further on we analyse structures which contain more than attribute and domain entity. We extended our analyses to structures, which e.g. contain a modifier like 'sehr' (very) or a negator like 'nicht' (not) and other adjectives.

Example 6 *Result of the example: 'Brustkorb nicht sehr breit.*'

```
<RATT-V>
   <ENTITY>Brustkorb</ENTITY>
   <VALUE CNT="1">nicht-sehr-breit</VALUE>
</RATT-V>
```

Here the attribute is compounded of a series of words from different wordclasses, because at the time we work with binary relations only. In ongoing work we will further detail this semantic interpretation. In addition we analyse complex structures like coordinated structures. There exist various pattern, e.g. <NP> <Adjective>|<Verb> 'und' <Adjective>|<Verb><Fullstop>. These structures are interpreted as <Anatomic-entity> <Attribute-value1> 'and' <Anatomic-entity> <Attribute-value2><Fullstop>.

Example 7 *For example: 'Beckengeruest festgefuegt und unversehrt.'. The result is:*

```
<RATT-V>
   <ENTITY>Beckengeruest</ENTITY>
   <VALUE CNT="1">festgefuegt</VALUE>
   <VALUE CNT="1">unversehrt</VALUE>
</RATT-V>
```

[1] is|are, | expresses alternatives in pattern

The inverse structure (the coordination at the beginning of the pattern) e.g. <Adjective> 'und' <Adjective> <NP> <Fullstop> can also be analysed.

Example 8 *For example: 'Akute und chronische Erweiterung des Herzens.*'

```
<RATT-V>
   <ENTITY>Erweiterung des Herzens</ENTITY>
   <VALUE CNT="1">akute</VALUE>
   <VALUE CNT="1">chronische</VALUE>
</RATT-V>
```

Another coordinated pattern is <NP> 'und' <NP> <Adjective>|<Verb> <Fullstop>. The semantic interpretation is similar to the analysis of the simple structures: <Anatomic-entity1> <Attribute-value> 'and' <Anatomic-entity2><Attribute-value><Fullstop>.

Example 9 *For example: 'Rippen und Wirbelsaeule intakt.' The result is:*

```
<RATT-V>
   <ENTITY>Rippen</ENTITY>
   <VALUE CNT="1">intakt</VALUE>
</RATT-V>
<RATT-V>
   <ENTITY>Wirbelsaeule</ENTITY>
   <VALUE CNT="1">intakt</VALUE>
</RATT-V>
```

The pattern, like the example 'Leber und Niere ohne Besonderheiten.'('Liver and kidney without findings.'), differs from the last described structures in the kind of the attribute. In this structure the attribute is described by a preposition phrase. The analysis returns

Example 10 *Result of 'Leber und Niere ohne Besonderheiten.':*

```
<RATT-V>
   <ENTITY>Leber</ENTITY>
   <VALUE CNT="1">ohne Besonderheiten</VALUE>
</RATT-V>
<RATT-V>
   <ENTITY>Niere</ENTITY>
   <VALUE CNT="1">ohne Besonderheiten</VALUE>
</RATT-V>
```

4. Ontology creation

4.1. Analysis of co-occurrence data

Co-occurrence data are used for clustering: We start e.g. with an adjective token that is related to a single noun type only in the analysed data.

If - again within the corpus given - this noun co-occurs only with this very adjective then the relation between the noun's concept and the property denoted by the adjective is very strong. It may even be the case that the adjective-noun-combination is a name like fixed phrase.

If the noun co-occurs with other adjectives as well it is interesting to uncover the relation between the adjectives and denoted properties respectively.

There are a number of possibilities:

- Two adjectives may be used as 'quasi-synonyms',
- Adjectives may be in an antinomy relation,
- Adjectives may refer to discrete values of a property that are linearly ordered on a scale,

- Adjectives refer to values of different properties.

We can proceed in a zig-zag-manner:

We have started with a single adjective and checked for its co-occurring noun. We then asked for other adjectives co-occurring with this noun. In the next step we extend the set of nouns with those nouns that co-occur with at least one of the adjectives in the adjective set.

Then we can extend the adjective set accordingly. The process will definitely stop if in a step the set to be expanded (either the noun or the adjective set) is no longer growing and has thus reached a fixed point.

As soon as the zig-zag-procedure adds an adjective to the adjective set thats co-occurs with many nouns of different type then in the next step, when the co-occurring adjectives of all these nouns are added, we may produce (nearly) a full covering of all adjectives and of all nouns respectively.

4.2. Exploiting co-occurrence information

4.2.1. Concept detection

A noun phrase of the type <Adj> <Noun> may be like the name of a concept but this does not always hold and depends on usage.

An example: 'fluessige Galle' as in 'In der Gallenblase fluessige Galle' is a property value, not a name. On the other hand 'harte Hirnhaut' is to be treated as nameing a concept. This can be inferred from the usage of the NP 'harte Hirnhaut' in structures of the type

<NP><Adj> like 'harte Hirnhaut perlmuttergrau'.

4.2.2. Concept classification

Currently linguistic structures are mapped into binary relations. An example:

Harte Hirnhaut grauweiss.

is an application of the grammar rule with

<NP> <Adjective> <Fullstop>

as right hand side. This establishes a <Property> <Concept> pair.

If we invert this relation (i.e. give a listing of all property values that co-occur – with number of occurrences above a threshold – with the concept) this yields:

Harte Hirnhaut: glaenzend, grauweiss, perlmuttergrau, weisslich-gelblich-verfaerbt, intakt, grauroetlich, blaeulich-durchscheinend

If we analyze these adjectives (and compounded adjective groups) we find the following:

- there is one very generic property 'intakt' (engl. 'intact') that is usable with almost any anatomic-entity
- the adjective 'glaenzend' is characterising the visual appearance of the brain skin as shiny
- all other adjectives denote a variety colors

Thus the brain skin can be classified as an anatomic-entity whose color values are relevant in autopsy reports.

4.2.3. Concept grouping

Clustering of co-occurrence data allows to detect candidates for semantic groups as well as synonyms and/or paraphrases.

- 'spiegelnd': 'Herzueberzug', 'Lungenueberzug'
- 'unversehrt': 'Haut des Rueckens', 'Stirnhaut'
- 'frei': 'Gehoergange', 'Ausfuehrungsgang', 'Kehlkopfeingang'

All concepts co-occurring with 'frei' are of the type tube.

4.3. Ontological relations

What ontological relations can be inferred?

- Is-a: Leber Is-a Organ
- Part-of: Schleimhaut Part-of Magen (generalized Schleimhaut Part-of Organ)
- other n-nary relations: e.g. 'nicht widernatuerlich beweglich'

Further on we can find a classification of relations resp. the domain range of an relation. For example the relation 'geoeffnet' (opened) can be changed by modifier in the attribute-value

- 'geoeffnet'
- 'spaltweit-geoeffnet'
- 'spaltfoermig-geoeffnet'
- 'geschlossen' (as opposite to 'geoeffnet')

5. Discussion

Our current work is of an investigative nature. The size of the corpus is still small. It is planned to apply the techniques developed with the initial corpus to the collection of several thousand protocols. The number of occurrences is still small and statistical methods are therefore not yet adequate. Even if quantitative measures are not applicable on the basis of this corpus occurrence data can be interpreted qualitatively.

Since we have just recently started with the domain of autopsy protocols there are e.g. still gaps in grammar coverage and in the tagging process (not every unknown word can be classified by heuristics). In the corpus currently approx 37 % of the sentences and telegrammatic structures can be fully processed (i.e. get at least one reading covering the structure as a whole; multiple readings are possible.) Experiments with the full corpus will allow to evaluate how reliable the results are.

The telegrammatic style results in shorter and – on the first sight – 'simpler' linguistic structures. As a trade-off these structures are less constrained and this e.g. complicates the derivation of morphosyntactic features from context or makes inferred results less reliable.

An example: If 'Nieren' is an unknown token the full sentence 'Die Nieren sind unversehrt' allows to infer that the token is a plural form, the same inference is not possible from the telegrammatic version 'Nieren unversehrt'.

5.1. XDOC as a workbench

We are aiming at a workbench with a rich functionality but we do not expect a fully automatic and autonomous solution. The user shall be supported as good as possible but s/he will still be involved in the process.

Our approach is interactive. The user has to confirm suggestions from the system. He is accepting or refusing, but can delegate searching, comparing, counting etc. to the system.

5.2. Acquisition of domain knowledge

Some findings in autopsy protocols are results of measurements: values of weights, sizes, diameters etc. are reported.

This allows to collect 'typical values' and to gain distributions for ranges of values.

For weights a typical pattern is:

<organ>gewicht <number> g.

'Lebergewicht ... g'

From the texts we derived the range of the weight-relation for example for the organ kidney as 135 g to 270 g (in a medical manual the weight of the kidney is defined in the range of 120 g to 300 g).

Sometimes contextual interpretation is necessary:

<organ><property-value>. Gewicht <number> g.

Here the generic term 'Gewicht' (weight) has to be interpreted as referring to the organ in focus.

Similar constructions are employed for other indicators like diameters.

5.3. Future work:

For the quality of inferences the detection of synonyms and paraphrases plays a major role, e.g. 'Blase' and 'Harnblase' do refer to the same organ, 'Stirnhaut' and 'Haut der Stirn' denote the same region: the skin of the forehead.

A general solution for coordinated structures will be necessary.

A subtype of coordinated structures includes truncation of compounds. An example: 'Wangen- und Kinnpartie unauffaellig.' The reconstruction of the untruncated term is not always as simple as in the example. For this task we need an approach similar to the one described in (Buitelaar and Scaleanu, 2002). It must not only be analysed which is the semantic meaning of the word, but rather which is the word, which was truncated. One criterion is, that the words must have the same semantic category.

A general component for the semantic treatment of noun compounds is needed. This will have to interact with contextual interpretation. In an example like

24. Hirngewicht 1490 g. Windungen abgeflacht, Furchen verstrichen. ...

it has to be detected that with the reference to the weight of the brain ('Hirngewicht') the brain is established as topic and that the terms 'Windungen' and 'Furchen' are referring to findings about the brain's visible appearance.

Autopsy protocols are written in a way such that the course of the autopsy is directly reflected in discourse structure. The autopsy on the other hand follows anatomic structures and their neighbourhood relations. In local contexts we both find part-of relations between anatomic structures as well as neighbourhood relations.

The analysis of noun phrases needs to be more fine grained. Structures like 'Haut des Rueckens' or 'Haut ueber der Nase' should e.g. be interpreted as localisation information that is specifying regions of the skin (here: 'skin of the back' and 'skin of the nose').

6. References

Tim Bray, Jean Paoli, and C.M. Sperberg-McQueen. 1998. Extensible Markup Language (XML) 1.0. *http://www.w3.org/TR/1998/REC-xml-19980210*.

P. Buitelaar and B. Scaleanu. 2002. Extending Synsets with Medical Terms.

J. Clark. 2002. http://www.jclark.com.

W. Finkler and G. Neumann. 1988. MORPHIX: a fast Realization of a classification-based Approach to Morphology. In H. Trost, editor, *Proc. der 4. Österreichischen Artificial-Intelligence Tagung, Wiener Workshop Wissensbasierte Sprachverarbeitung*, pages 11–19. Springer Verlag.

GermaNet-Project-Site. 2002. http://www.sfs.nphil.uni-tuebingen.de/lsd/.

M. Kunze and D. Rösner. 2001a. XDOC - Extraktion, Rep"asentation und Auswertung von Informationen. In *GLDV-Workshop: Werkzeuge zur automatischen Analyse und Verarbeitung*.

M. Kunze and D. Rösner. 2001b. An XML-based Approach for the Presentation and Exploitation of Extracted Information. In *International Workshop on Web Document Analysis*.

D. Rösner. 2000. Combining robust parsing and lexical acquisition in the XDOC system. In *KONVENS 2000 Sprachkommunikation*, ITG-Fachbericht 161, ISBN 3-8007-2564-9, pages 75–80. VDE Verlag, Berlin, Offenbach.

XSL Site. 2002. http://www.w3.org/style/xsl.

Wordnet-Project-Site. 2002. http://www.cogsci.princeton.edu/ wn/.

Supports

This workshop has been organised in part with support from the ISLE Project – International Standards for Language Engineering (IST-10647). The ISLE Project is funded by the European Union, the National Science Foundation of the USA, and the Swiss Government.

Workshop Programme

9:00 – 9:15	Introduction and welcome to the workshop – *Maghi King*
9:15 – 9:45	Introduction to the ISLE taxonomy for MT evaluation – *Maghi King* and *Andrei Popescu-Belis*
9:45 – 10:15	Overview of human-based metrics for MT evaluation – *Florence Reeder*
10:15– 10:45	The DARPA 2001 automated metric and its relation to IBM's BLEU – *George Doddington*
10:45 – 11:00	Summary of the proposed evaluation tasks – *Andrei Popescu-Belis*
11:00 – 11:40	**Coffee break** and extra evaluation time [general LREC break: 11:00-11:20]
11:40 – 12:00	Summary of the goals of our collective hands-on experiment – *Andrei Popescu-Belis*
12:00 – 13:00	Reports on individual evaluations (human vs. automatic) – *All workshop participants*
13:00 – 14:30	**Lunch break**
14:30 – 15:30	Reports on individual evaluations (human vs. automatic, *continued*) – *All workshop participants*
15:30 – 16:40	Synthesis of evaluation exercises: reliability and correlation of the metrics used in the hands-on exercises – *Workshop organisers*
16:40 – 17:00	**Coffee break** [synchronised with general LREC break]
17:00 – 18:30	Roundtable discussions of the observed results. Conclusions – *All workshop participants*

Workshop Organisers

Marianne Dabbadie	EVALING, Paris (France)
Anthony Hartley	Centre for Translation Studies, University of Leeds (UK)
Eduard Hovy	USC Information Sciences Institute, Marina del Rey (USA)
Margaret King	ISSCO/TIM/ETI, University of Geneva (Switzerland)
Bente Maegaard	Center for Sprogteknologi, Copenhagen (Denmark)
Sandra Manzi	ISSCO/TIM/ETI, University of Geneva (Switzerland)
Keith J. Miller	The MITRE Corporation (USA)
Widad Mustafa El Hadi	Université Lille III - Charles de Gaulle (France)
Andrei Popescu-Belis	ISSCO/TIM/ETI, University of Geneva (Switzerland)
Florence Reeder	The MITRE Corporation (USA)
Michelle Vanni	U.S. Department of Defense (USA)

Table of Contents

Author Index

An Introduction to MT Evaluation

Eduard Hovy[*], Maghi King[**], Andrei Popescu-Belis[**]

[*]USC Information Sciences Institute
4676 Admiralty Way
Marina del Rey, CA 90292-6695, USA
hovy@isi.edu

[**]ISSCO/TIM/ETI, University of Geneva,
École de Traduction et d'Interprétation
40 Bvd. du Pont d'Arve
CH-1211 Geneva 4 – Switzerland
margaret.king@issco.unige.ch
andrei.popescu-belis@issco.unige.ch

Abstract

This section of the workbook describes the principles and mechanism of an integrative effort in machine translation (MT) evaluation. Building upon previous standardization initiatives, above all ISO/IEC 9126, 14598 and EAGLES, we attempt to classify into a coherent taxonomy most of the characteristics, attributes and metrics that have been proposed for MT evaluation. The main articulation of this flexible framework is the link between a taxonomy that helps evaluators define a context of use for the evaluated software, and a taxonomy of the quality characteristics and associated metrics. The document overviews these elements and provides a perspective on ongoing work in MT evaluation.

1. Introduction

Evaluating machine translation is important for everyone involved: researchers need to know if their theories make a difference, commercial developers want to impress customers, and users have to decide which system to employ. Given the richness of the literature, and the complexity of the enterprise, there is a need for an overall perspective, something that helps the potential evaluator approach the problem in a more informed way, and that might help pave the way toward an eventual theory of MT evaluation.

Our main effort is to build a coherent overview of the various features and metrics used in the past, to offer a common descriptive framework and vocabulary, and to unify the process of evaluation design. Therefore, we present here a parameterizable taxonomy of the various attributes of an MT system that are relevant to its utility, as well as correspondences between the intended context of use and the desired system qualities, i.e., a quality model. Our initiative builds upon previous work in the standardization of evaluation, while applying to MT the ISO/IEC standards for software evaluation.

We first review (Section 2) the main evaluation efforts in MT and in software engineering (ISO/IEC standards). Then we describe the need for two taxonomies, one relating the context of use (analyzed in Section 3) to the quality characteristics, the other relating the quality characteristics to the metrics. In Section 4 we provide a brief overview of these taxonomies, together with a view on their dissemination and use. We finally outline (Section 5) our perspectives on current and future developments.

2. Formalizing Evaluation: from MT to Software Engineering

2.1. Previous Approaches to MT Evaluation

The path to a systematic picture of MT evaluation is long and hard. While it is impossible to write a comprehensive overview of the MT evaluation literature, certain tendencies and trends should be mentioned. First, throughout the history of evaluation, two aspects – often called quality and fidelity – stand out. Particularly MT researchers often feel that if a system produces syntactically and lexically well-formed sentences (i.e., high quality output), and does not distort the meaning (semantics) of the input (i.e., high fidelity), then the evaluation is sufficient. System developers and real-world users often add evaluation measures, notably system extensibility (how easy it is for a user to add new words, grammar, and transfer rules), coverage (specialization of the system to the domains of interest), and price. In fact, as discussed in (Church and Hovy, 1993), for some real-world applications quality may take a back seat to these factors.

Various ways of measuring quality have been proposed, some focusing on specific syntactic constructions (relative clauses, number agreement, etc.) (Flanagan, 1994), others simply asking judges to rate each sentence as a whole on an N-point scale (White et al., 1992 1994; Doyon et al., 1998), and others automatically measuring the perplexity of a target text against a bigram or trigram language model of ideal translations (Papineni et al., 2001). The amount of agreement among such measures has never been studied. Fidelity requires bilingual judges, and is usually measured on an N-point scale by having judges rate how well each portion of the system's output expresses the content of an equivalent portion of one or more ideal (human) translations (White et al., 1992 1994; Doyon et al., 1998). A proposal to measure fidelity automatically by projecting both system output and a number of ideal human translations into a vector space of words, and then measuring how far the system's translation deviates from the mean of the ideal ones, is an intriguing idea whose generality still needs to be proved (Thompson, 1992). In

similar vein, it may be possible to use the above mentioned perplexity measure also to evaluate fidelity (Papineni et al., 2001).

The Japanese JEIDA study of 1992 (Nomura, 1992; Nomura and Isahara, 1992), paralleling EAGLES, identified two sets of 14 parameters each: one that characterizes the desired context of use of an MT system, and the other that characterizes the MT system and its output. A mapping between these two sets of parameters allows one to determine the degree of match, and hence to predict which system would be appropriate for which user. In similar vein, various companies published large reports in which several commercial MT systems are compared thoroughly on a few dozen criteria (Mason and Rinsche, 1995; Infoshop, 1999). The OVUM report includes usability, customizability, application to total translation process, language coverage, terminology building, documentation, and others.

The variety of MT evaluations is enormous, from the influential ALPAC Report (Pierce et al., 1966) to the largest ever competitive MT evaluations, funded by the US Defense Advanced Research Projects Agency (DARPA) (White et al., 1992 1994) and beyond. Some influential contributions are (Kay, 1980; Nagao, 1989). Van Slype (1979) produced a thorough study reviewing MT evaluation at the end of the 1970s, and reviews for the 1980s can be found in (Lehrberger and Bourbeau, 1988; King and Falkedal, 1990). The pre-AMTA workshop on evaluation contains a useful set of papers (AMTA, 1992).

2.2. The EAGLES Guidelines for NLP Evaluation

The European EAGLES initiatives (1993-1996) came into being as an attempt to create standards for language engineering. It was accepted that no single evaluation scheme could be developed even for a specific application, simply because what counted as a "good" system would depend critically on the use of the system. However, it did seem possible to create a general framework for evaluation design, which could guide the creation of individual evaluations and make it easier to understand and compare the results. An important influence here was the 1993 report by Sparck-Jones and Galliers, later published in book form (1996), and the ISO/IEC 9126 (cf. next section).

These first attempts proposed the definition of a general quality model for NLP systems in terms of a hierarchically structured set of features and attributes, where the leaves of the structure were measurable attributes, with which specific metrics were associated. The specific needs of a particular user or class of users were catered for by extracting from the general model just those features relevant to that user, and by allowing the results of metrics to be combined in different ways in order to reflect differing needs. These attempts were validated by application to quite simple examples of language technology: spelling checkers, then grammar checkers (TEMAA, 1996) and translation memory systems (preliminary work), but the EAGLES methodology was also used outside the project for dialogue, speech recognition and dictation systems.

When the ISLE project (International Standards for Language Engineering) was proposed in 1999, the American partners had also been working along the lines of taxonomies of features (Hovy, 1999), focusing explicitly on MT and developing in the same formalism a taxonomization of user needs, along the lines suggested by the JEIDA study (Nomura, 1992). The evaluation working group of the ISLE project (one of the three ISLE working groups) therefore decided to concentrate on MT systems.

2.3. The ISO/IEC Standards for Software Evaluation

2.3.1. A Growing Set of Standards

The International Organization for Standardization (ISO) together with the International Electrotechnical Commission (IEC) have initiated in the past decade an important effort towards the standardization of software evaluation. In 1991 appeared the ISO/IEC 9126 standard (ISO/IEC-9126, 1991), a milestone that proposed a definition of the concept of quality, and decomposed software quality into six generic quality characteristics. Evaluation is the measure of the quality of a system in a given context, as stated by the definition of quality as "the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs" (ISO/IEC9126, 1991, p. 2).

Subsequent efforts led to a set of standards, some still in draft versions today. It appeared that a new series was necessary for the evaluation process, of which the first in the series (ISO/IEC-14598, 1998 2001, Part 1) provides an overview. The new version of the ISO/IEC 9126 standard will finally comprise four inter-related standards: standards for software quality models (ISO/IEC-9126-1, 2001), for external, internal and quality in use metrics (ISO/IEC 9126- 2 to 4, unpublished). Regarding the 14598 series (ISO/IEC14598, 1998 2001), now completely published, volumes subsequent to ISO/IEC 14598-1 focus on the planning and management (14598-2) and documentation (14598-6) of the evaluation process, and apply the generic organization framework to developers (14598-3), acquirers (14598-4) and evaluators (14598-5).

2.3.2. The Definition of a Quality Model

This subsection situates our proposal for MT evaluation within the ISO/IEC framework. According to ISO/IEC 14598-1 (1998 2001, Part 1, p. 12, fig. 4), the software life-cycle starts with an analysis of user needs that will be answered by the software, which determine in their turn a set of specifications. From the point of view of quality, these are the external quality requirements. Then, the software is built during the design and development phase, when quality becomes an internal matter related to the characteristics of the system itself. Once a product is obtained, it is possible to assess its internal quality, then the external quality, i.e., the extent to which it satisfies the specified requirements. Finally, turning back to the user needs that were at the origin of the software, quality in use is the extent to which the software really helps users fulfill their tasks (ISO/IEC-9126-1, 2001, p. 11).

Quality in use does not follow automatically from external quality since it is not possible to predict all the results of using the software before it is completely operational. In addition, for MT software, there seems to

be no straightforward link, in the conception phase, from the external quality requirements to the internal structure of a system. Therefore, the relation between external and internal qualities is quite loose.

Following mainly (ISO/IEC-9126-1, 2001), software quality results from six quality characteristics:

- functionality
- reliability
- usability
- efficiency
- maintainability
- portability

These characteristics have been refined into software sub-characteristics that are still domain-independent (ISO/IEC 9126-1). These form a loose hierarchy (some overlapping is possible), but the terminal entries are always measurable features of the software, that is, attributes. Following (ISO/IEC-14598, 1998-2001, Part 1), "a measurement is the use of a metric to assign a value (i.e., a measure, be it a number or a category) from a scale to an attribute of an entity".

The six top level quality characteristics are the same for external as well as for internal quality. The hierarchy of sub-characteristics may be different, whereas the attributes are certainly different, since external quality is measured through external attributes (related to the behavior of a system) while internal quality is measured through internal attributes (related to intrinsic features of the system).

Finally, quality in use results from four characteristics: effectiveness, productivity, safety, and satisfaction. These can only be measured in the operating environment of the software, thus seeming less prone to standardization (see however (Daly-Jones et al., 1999) and ISO/IEC 9126-4).

2.3.3. Stages in the Evaluation Process

The five consecutive phases of the evaluation process according to (ISO/IEC-9126, 1991, p. 6) and (ISO/IEC-14598, 1998 2001, Part 5, p. 7) are:

- establish the quality requirements (the list of required quality characteristics);
- specify the evaluation (specify measurements and map them to requirements);
- design the evaluation, producing the evaluation plan that documents the procedures used to perform measurements);
- execute the evaluation, producing a draft evaluation report;
- conclude the evaluation.

During specification of the measurements, each required quality characteristic must be decomposed into the relevant sub-characteristics, and metrics must be specified for each of the attributes arrived at in this process. More precisely, three elements must be distinguished in the specification and design processes; these correspond to the following stages in execution:

- application of a metric (*a*);
- rating of the measured value (*b*);
- integration (assessment) of the various ratings (*c*).

It must be noted that (a) and (b) may be merged in the concept of 'measure', as in ISO/IEC 14598-1, and that integration (c) is optional. Still, at the level of concrete evaluations of systems, the above distinction, advocated also by EAGLES (EAGLES-Evaluation-Workgroup, 1996), seems particularly useful: to evaluate a system, a metric is applied for each of the selected attributes, yielding as a score a raw or intrinsic score; these scores are then transformed into marks or rating levels on a given scale; finally, during assessment, rating levels are combined if a single result must be provided for a system.

A single final rating is often less informative, but more adapted to comparative evaluation. However, an expandable rating, in which a single value can be decomposed on demand into several components, is made possible when the relative strengths of the component metrics are understood. Conversely, the EAGLES methodology (EAGLES-Evaluation-Workgroup, 1996, p. 15) considers the set of ratings to be the final result of the evaluation.

3. Relation between the Context of Use, Quality Characteristics, and Metrics

Just as one cannot determine "what is the best house?", one cannot expect to determine the best MT system without further specifications. Just like a house, an MT system is intended for certain users, located in specific circumstances, and required for specific functions. Which parameters to pay attention to, and how much weight to assign each one, remains the prerogative of the user/evaluator. The importance of the context for effective system deployment and use has been long understood, and has been a focus of study for MT specifically in the JEIDA report (Nomura, 1992).

3.1. The Context of Use in the ISO/IEC Standards

While a good definition of the context of use is essential for accurate evaluation, in ISO/IEC the context of use plays a somewhat lesser role. The context of use is considered at the beginning of the software's life-cycle (ISO/IEC-14598, 1998 2001, Part 1), and appears in the definition of quality in use. No obvious connection between quality in use metrics and internal or external ones is provided. There is thus no overall indication how to take into account the context of use in evaluating a product.

There are however two interesting mentions of the context of use in ISO/IEC. First, the ISO/IEC standard for acquirers (ISO/IEC-14598, 1998 2001, Part 4, Annex B, pp. 21-22) exemplifies the link between the desired integrity of the evaluated software (integrity pertains to the risk of using the software) and the evaluation activities, in particular the choice of a quality model: for higher integrity, more evaluation procedures have to be fulfilled. The six ISO/IEC 9126 characteristics are also ordered differently according to the required integrity. Second, (ISO/IEC-14598, 1998 2001, Part 5, Annex B, pp. 22-25) gives another relation between "evaluation techniques" and the acceptable risk level. These proposals attempt thus to fill the gap between concrete contexts of use and generic quality models.

3.2. Relating the Context of Use to the Quality Model

When specifying an evaluation, the external evaluator – a person or a group in charge of estimating the quality of MT software – must mainly provide a quality model based on the expected context of use of the software. Guidelines for MT evaluation must therefore contain the following elements:

1. A classification of the main features defining a context of use: the user of the MT system, the task, and the nature of the input to the system.
2. A classification of the MT software quality characteristics, detailed into hierarchies of sub-characteristics and attributes, with internal and/or external attributes (i.e., metrics) at the bottom level. The upper levels coincide with the ISO/IEC 9126 characteristics.
3. A mapping from the first classification to the second, which defines (or at least suggests) the characteristics, sub-characteristics and attributes or metrics that are the most relevant for each context of use.

This broad view of evaluation is still, by comparison to ISO/IEC, focused on the technical aspect of evaluation. Despite the proximity between the taxonomy of contexts of use and quality in use, we do not extend our guidelines to quality in use, since this must be measured fully in context, using metrics that have less to do with MT evaluation than with ergonomics and productivity measures. Therefore, we have proposed elsewhere (Hovy, King and Popescu-Belis, 2002) a formal model of the mapping at point (3) above.

To summarize, building upon the definitions in Section 2.3.3., we consider the set of all possible attributes for MT software $\{A_1, A_2, \ldots, A_n\}$, and the process of evaluation is defined using three stages and the corresponding mappings: m_{Ai} (application of metrics), r_{Ai} (rating of measured value), and α (assessment of ratings).

From this point of view, the correspondence described at point (3) above holds between a context of use and the assessment or averaging function α. Point (3) is thus addressed by providing, for each context of use, the corresponding assessment function, i.e. the function that assigns a greater weight to the attributes relevant to that particular context. In the formal model, α is simplified by choosing a linear selection function.

4. The Contents of the Two Taxonomies

The schema below gives a general view of the contents of the two taxonomies. The first one enumerates non exclusive characteristics of the context of use grouped in three complementary parts (task, user, input). The second one develops the quality model, and its starting point is the six ISO/IEC quality characteristics. The reader will notice that our efforts towards a synthesis have not yet succeeded in unifying internal and external attributes under these six characteristics. As mentioned in Section 2.3.2., the link between internal features and external performance is not yet completely clear for MT systems. So, the internal attributes are structured here in a branch separate from the six ISO/IEC characteristics, which are measured by external metrics.

For lack of space, the hierarchies below represent a brief snapshot of the actual state of our proposal, which may be revised under feedback from the community. The full version available over the Internet (`http://www.issco.unige.ch/projects/isle/taxonomy2`) has about 30 pages, and expands each taxon with the corresponding metrics extracted from the literature. The website provides an interactive version and a printable version of the taxonomy.

- Specifying the context of use
  - Characteristics of the translation task
    - Assimilation
    - Dissemination
    - Communication
  - Characteristics of the user of the MT system
    - Linguistic education
    - Language proficiency in source language
    - Language proficiency in target language
    - Present translation needs
  - Input characteristics (author and text)
    - Document / text type
    - Author characteristics
    - Sources of error in the input
      - Intentional error sources
      - Medium-related error sources
      - Performance-related errors
- Quality characteristics, sub-characteristics and attributes
  - System internal characteristics
    - MT system-specific characteristics (translation process)
    - Model of translation process (rule-based / example-based / statistical / translation memory)
    - Linguistic resources and utilities
    - Characteristics related to the intended mode of use
      - Post-editing or post-translation capacities
      - Pre-editing or pre-translation capacities
      - Vocabulary search
      - User performed dictionary updating
      - Automatic dictionary updating
  - System external characteristics
    - Functionality
      - Suitability (coverage – readability – fluency / style – clarity – terminology)
      - Accuracy (text as a whole – individual sentence level – types of errors)
      - Interoperability
      - Compliance
      - Security
    - Reliability
    - Usability
    - Efficiency
      - Time behavior (production time / speed of translation – reading time – revision and post-editing / correction time)
      - Resource behavior
    - Maintainability
    - Portability
    - Cost

Practical work using the present taxonomy was the object of a series of workshops organized by the

Evaluation Work Group of the ISLE Project. There has been considerable continuity between workshops, with the result that the most recent in the series offered a number of interesting examples of using the taxonomy in practice. A very wide range of topics was covered, including the development of new metrics, investigations into possible correlation between metrics, ways to take into account different user needs, novel scenarios both for the evaluation and for the ultimate use of an MT system and ways to automate MT evaluation. The four workshops took place in October 2000 (at AMTA 2000), April 2001 (stand-alone hands-on workshop at ISSCO, Geneva), June 2001 (at NAACL 2001) and September 2001 (at MT Summit VIII).

Among the first conclusions drawn from the workshops is the fact that evaluators tend to favor some parts of the second taxonomy – especially attributes related to the quality of the output text – and to neglect some others – for instance the definition of a user profile. It appears that the sub-hierarchy related to the "hard problem", i.e. the quality of output text, should be better developed. Sub-characteristics such as the translation quality for noun phrases (which is further on split into several attributes) attracted steady interest.

The proposed taxonomies can be accessed and browsed through a computer interface. The mechanism that supports this function also ensures that the various nodes and leaves of the categories are stored in a common format (based on XML), and simplifies considerably the periodic update of the classifications (Popescu-Belis et al., 2001). A first version of our taxonomies is visible at `http://www.isi.edu/natural-language/mteval` and the second one at `http://www.issco.unige.ch/projects/isle/taxonomy2` – the two sites will soon mirror a third, updated version.

5. Towards the Refinement of the Taxonomies

The taxonomies form but the first step in a larger program – listing the essential parameters of importance to MT evaluation. But for a comprehensive and systematic understanding of the problem, one also has to analyze the nature and results of the actual evaluation measures used. In our current work, a primary focus is the analysis of the measures and metrics: their variation, correlation, expected deviation, reliability, cost to perform, etc. This section outlines first a theoretical framework featuring coherence criteria for the metrics, then lists the (unfortunately very few) examples from previous research.

5.1. Coherence Criteria for Evaluation Metrics

We have defined coherence criteria for NLP evaluation metrics in an EAGLES-based framework (Popescu-Belis, 1999). The following criteria, applied to a case where there is no golden standard to compare a system's response to, enable evaluators to choose the most suitable metric for a given attribute and help them interpret the measures.

A metric m_{Ai} for a given attribute A_i is a function from an abstract 'quality space' onto a numeric interval, say [0,1] or [0%, 100%]. With respect to definition (a) in Section 2.3.3., each system occupies a place in the quality space of A_i, quantified by that metric. Since the goal of evaluators is to quantify the quality level using a metric, they must poll the experts to get an idea of what the best and the worst quality levels are for A_i.

It is often easy to find the best quality of a response, but there are at least two kinds of very poor quality levels: (a) the worst imaginable ones (which a system may rarely actually descend to) and (b) the levels attained by simplistic or baseline systems. For instance, for the capacity to translate polysemous words, a system that always outputs the most frequent sense of source words does far better than the worst possible system (the one that always gets it wrong) or than a random system. Once these limits are identified, the following coherence criteria should be tested for:

- **UL – upper limit**: A metric for an attribute A_i must reach 1 for best quality of a system, and (reciprocally) only reach 1 when the quality is perfect;

- **LL – lower limit:** A metric for an attribute A_i must reach 0 for the worst possible quality of a system, and only reach 0 when the quality is extremely low. Since it is not easy to identify the set of lowest quality cases, one can alternatively check that:
  - receiving a 0 score corresponds to low quality;
  - all the worst quality responses receive a 0 score;
  - the lowest theoretical scores are close or equal to 0 (a necessary condition for the previous requirement).

- **M – monotonicity**: A metric must be monotonic, that is, if the quality of system A is higher than that of system B, then the score of A must be higher than the score of B.

One should note that it is difficult to prove that a metric does satisfy these coherence criteria, and much easier to use counter-examples to criticize a measure on the basis of these criteria. Finally, one can also compare two metrics, stating that m_1 is more severe than m_2 if it yields lower scores for each possible quality level.

5.2. Analyzing the Behavior of Measures

Since our taxonomy gathers numerous quality attributes and metrics, there are basic aspects of MT that may be rated through several attributes, and each attribute may be scored using several metrics. This uncomfortable state of affairs calls for investigation. If it should turn out, for a given characteristic, that one specific attribute correlates perfectly with human judgments, subsumes most or all of the other proposed measures, can be expressed easily through one or more metrics, and is cheap to apply, we should have no reason to look further: that aspect of the taxonomy would be settled.

The full list of desiderata for a measure is not immediately clear, but there are some obvious ones. The measure:

- must be easy to define, clear and intuitive;
- must correlate well with human judgments under all conditions, genres, domains, etc.;

- must be `tight', exhibiting as little variance as possible across evaluators, or for equivalent inputs;
- must be cheap to prepare (i.e., not require a great deal of human effort for training data or ideal examples);
- must be cheap to apply;
- should be automated if possible.

Unexpectedly, the literature contains rather few methodological studies of this kind. Few evaluators have bothered to try someone else's measures too, and correlate the results. However, there are some advances. In recent promising work using the DARPA 1994 evaluation results (White et al., 1992 1994), White and Forner have studied the correlation between intelligibility (syntactic fluency) and fidelity (White, 2001) and between fidelity and noun compound translation (Forner and White, 2001). As one would expect with measures focusing on aspects as different as syntax and semantics, some correlation was found, but not a clear one. Papineni et al. (2001) compared the scores given by BLEU, an algorithm mentioned above, with human judgments of the fluency and fidelity of translations. They found a very high level of agreement, with correlation coefficients of 0.99 (with monolingual judges) and 0.96 (bilingual ones).

Another important matter is inter-evaluator agreement, reported on by most careful evaluations. Although the way one formulates instructions has a major effect on subjects' behavior, we still lack guidelines for formulating the instructions for evaluators, and no idea how variations would affect systems' scores. Similarly, we do not know whether a 3-point scale is more effective than a 5- or 7-point. Experiments are needed to determine the optimal point between inter-evaluator consistency (higher on a shorter scale) and evaluation informativeness (higher on a longer scale). Still another important issue is the number of measure points required by each metric before the evaluation can be trusted, a figure that can be inferred from the confidence levels of past evaluation studies.

In the ISLE research we are now embarking on the design of a program that will help address these questions. Our very ambitious goal is to know, for each taxon in the taxonomy, which measure(s) are most appropriate, which metric(s) to use for them, how much work and cost is involved in applying each measure, and what final level of score should be considered acceptable (or not). Armed with this knowledge, a would-be evaluator would be able to make a much more informed selection of what to evaluate and how to go about it.

5.3. A View to the Future

It can be appreciated that building a taxonomy of features is an arduous task, made more difficult by the fact that few external criteria for correctness exist. It is easy to think of features and to create taxonomies; we therefore have several suggestions for taxonomy structure, and it is unfortunately very difficult to argue for the correctness of one against another. We therefore explicitly do not claim in this work that the present taxonomy is correct, complete, or not subject to change. We expect it to grow, to become more refined, and to be the subject of discussion and disagreement – that is the only way in which it will show its relevance. Nonetheless, while it is possible to continue refining the taxonomy, collecting additional references, and classifying additional measures, we feel that the most pressing work is only now being started. The taxonomy is but the first step toward a more comprehensive and systematic understanding of MT evaluation in all its complexity, including a dedicated program of systematic comparison between metrics.

The dream of a magic test that makes everything easy – preferably an automated process – always remains. A recent candidate, proposed by (Papineni et al., 2001), has these desirable characteristics. Should it be true that the method correlates very highly with human judgments, and that it really requires only a handful of expert translations, then we will be spared much work. But we will not be done. For although the existence of a quick and cheap evaluation measure is enough for many people, it still does not cover more than a small portion of the taxonomy; all the other aspects of MT that people have wished to measure in the past remain to be measured.

A general theme running throughout this document is that MT evaluation is simply a special, although rather complex, case of software evaluation in general. An obvious question then is whether the work described here can be extended to other fields. Some previous experience has shown that it applies relatively straightforwardly to some domains, for example, dialogue systems in a specific context of use. However, as the systems to be evaluated grow more complex, the contexts of use become potentially almost infinite. Trying to imagine them all and to draw up a descriptive scheme as we are doing for MT systems becomes a challenging problem, that must be addressed in the future. It is nevertheless our belief that the basic ISO notion of building a quality model and associating appropriate metrics to it should carry over to almost any application.

6. References

AMTA. 1992. MT evaluation: Basis for future directions (Proceedings of a workshop held in San Diego, CA). Technical report, Association for Machine Translation in the Americas (AMTA).

K. W. Church and E. H. Hovy. 1993. Good applications for crummy MT. *Machine Translation*, 8:239-258.

O. Daly-Jones, N. Bevan, and C. Thomas, editors. 1999. *Handbook of User-Centred Design*: INUSE 6.2. http://www.ejeisa.com/nectar/inuse.

J. Doyon, K. Taylor, and J.S. White. 1998. The DARPA MT evaluation methodology: Past and present. In *Proceedings of the AMTA Conference*, Philadelphia, PA.

EAGLES-Evaluation-Workgroup. 1996. EAGLES evaluation of natural language processing systems. Final report, Center for Sprogteknologi, Denmark, October 1996.

M. Flanagan. 1994. Error classification for MT evaluation. In *Proceedings of the AMTA Conference*, Columbia, Maryland.

M. Forner and J.S. White. 2001. Predicting MT fidelity from noun-compound handling. In *Workshop on MT Evaluation "Who did what to whom?" at Mt Summit VIII*, Santiago de Compostela, Spain.

E.H. Hovy. 1999. Toward finely differentiated evaluation metrics for MT. In *EAGLES Workshop on Standards and Evaluation*, Pisa, Italy.

Infoshop. 1999. Language translations: World market overview, current developments and competitive assessment. Technical report, Infoshop Japan, Global Information Inc., Kawasaki, Japan.

ISO/IEC-14598. 1998-2001. *ISO/IEC 14598 – Information technology – Software product evaluation – Part 1: General overview (1999), Part 2: Planning and management (2000), Part 3: Process for developers (2000), Part 4: Process for acquirers (1999), Part 5: Process for evaluators (1998), Part 6: Documentation of evaluation modules (2001)*. ISO/IEC, Geneva.

ISO/IEC-9126-1. 2001. *ISO/IEC 9126-1:2001 (E) – Software engineering – Product quality – Part 1: Quality model*. ISO/IEC, Geneva, June.

ISO/IEC-9126. 1991. *ISO/IEC 9126:1991 (E) – Information Technology – Software Product Evaluation – Quality Characteristics and Guidelines for Their Use*. ISO/IEC, Geneva.

M. Kay. 1980. The proper place of men and machines in language translation. Research Report CSL-80-11, XEROX PARC.

M. King and K. Falkedal. 1990. Using test suites in evaluation of MT systems. In *18th Coling Conference*, volume 2, Helsinki, Finland.

J. Lehrberger and L. Bourbeau. 1988. *Machine Translation: Linguistic Characteristics of MT Systems and General Methodology of Evaluation.* Lingvisticae Investigationes Suppl. 15. John Benjamins, Amsterdam.

J. Mason and A. Rinsche. 1995. Translation technology products. Report, OVUM Ltd.

M. Nagao. 1989. A Japanese view on MT in light of the considerations and recommendations reported by ALPAC, USA. Technical report, Japan Electronic Industry Development Association (JEIDA).

H. Nomura and J. Isahara. 1992. The JEIDA report on MT. In *Workshop on MT Evaluation: Basis for Future Directions*, San Diego, CA. Association for Machine Translation in the Americas (AMTA).

H. Nomura. 1992. JEIDA methodology and criteria on MTevaluation. Technical report, Japan Electronic Industry Development Association (JEIDA).

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. 2001. BLEU: a method for automatic evaluation of MT. Research Report, Computer Science RC22176 (W0109- 022), IBM Research Division, T.J.Watson Research Center, 17 September 2001.

J.R. Pierce, J.B. Carroll, E.P. Hamp, D.G. Hays, C.F. Hockett, A.G. Oettinger, and A. Perlis. 1966. Computers in translation and linguistics (ALPAC report). report 1416, National Academy of Sciences / National Research Council, 1966.

A. Popescu-Belis, S. Manzi, and M. King. 2001. Towards a two-stage taxonomy for MT evaluation. In *Workshop on MT Evaluation "Who did what to whom?" at MT Summit VIII*, pages 1-8, Santiago de Compostela, Spain.

A. Popescu-Belis. 1999. Evaluation of natural language processing systems: a model for coherence verification of quality measures. In Marc Blasband and Patrick Paroubek, editors, *A Blueprint for a General Infrastructure for Natural Language Processing Systems Evaluation Using Semi-Automatic Quantitative Black Box Approach in a Multilingual Environment*. ELSE Project LE4-8340 (Evaluation in Language and Speech Engineering).

K. Sparck-Jones and J.R. Galliers. 1996. *Evaluating Natural Language Processing Systems: An Analysis and Review*. Lecture Notes in Artificial Intelligence 1083. Springer-Verlag, Berlin / New York.

TEMAA. 1996. TEMAA final report. Technical Report LRE-62-070 (March 1996), Center fo Sprogteknologi, Copenhagen, Danemark, http://www.cst.ku.dk/ projects/temaa/D16/d16exp.html.

H. S. Thompson, editor. 1992. *The Strategic Role of Evaluation in Natural Language Processing and Speech Technology* (Record of a workshop sponsored by DANDI, ELSNET and HCRC). University of Edinburgh (Technical Report, May 1992).

G. Van Slype. 1979. Critical study of methods for evaluating the quality of MT. Technical Report BR 19142, European Commission / Directorate for General Scientific and Technical Information Management (DG XIII).

J.S. White et al. 1992-1994. ARPA workshops on MT (series of four workshops on comparative evaluation). Technical report, PRC Inc., McLean, Virginia.

J.S. White. 2001. Predicting intelligibility from fidelity in MT evaluation. In *Workshop on MT Evaluation "Who did what to whom?" at MT Summit VIII*, Santiago de Compostela, Spain.

A Hands-On Study of the Reliability and Coherence of Evaluation Metrics

Marianne Dabbadie[1], Anthony Hartley[2], Margaret King[3], Keith J. Miller[4], Widad Mustafa El Hadi[5], Andrei Popescu-Belis[3], Florence Reeder[4], Michelle Vanni[6]

[1] EVALING, Paris (France)
[2] Centre for Translation Studies, University of Leeds (UK)
[3] ISSCO/TIM/ETI, University of Geneva (Switzerland)
[4] The MITRE Corporation (USA)
[5] Université Lille III - Charles de Gaulle (France)
[6] U.S. Department of Defense (USA)

Abstract

This section of the workbook provides the description of the MT evaluation exercise that is proposed to the workshop participants, including the specification of the metrics for MT evaluation that the participants are suggested to use at the workshop.

1. A Collective Hands-on Exercise

1.1. Motivation

The motivations behind the LREC 2002 MT Evaluation workshop are grounded in previous work in the field, described at length in the previous section. The workshop is the sixth in a series of hands-on workshops on MT Evaluation, organized in the framework of the ISLE Project.

The goal of these hands-on evaluation workshops is to carry on a collective effort towards the standardization of MT evaluation. The ISLE taxonomy has been designed for standardization, but it would have not reached the present state without feedback from the participants at the workshops. Conversely, the participants have broadened their view of MT Evaluation, through the concrete use of the ISLE taxonomy for the design of toy evaluations, but also through extensive discussions with the organizers and other participants.

Some of the workshops have focused more on the setup of an evaluation depending on the desired context of use, others on metrics, others on reporting results obtained in this framework. As pointed out in the previous section, the need for a clear view of the performances of various metrics has prompted the organization of the present workshop, "Machine Translation Evaluation: Human Evaluators Meet Automated Metrics". Through hands-on application of selected metrics from the present workbook, the participants will be able to familiarize themselves with the current problems of MT Evaluation, to get a first-hand experience with recent metrics and to contribute to research in this field by their own observations of the metrics' behaviors.

1.2. Description of the exercise

The participants to the workshop are suggested to register with the organizers well before the day the workshop will take place (May 27, 2002). Thus, both organizers and participants will be able to prepare in advance an evaluation exercise (requiring several hours of work), so that the workshop itself can be devoted to the exploitation of those results.

The evaluation study that all participants are kindly required to carry on can be summarized as follows:

1. Select two evaluation metrics among those described below, preferably one "human-based" and one "automated" (more than two is welcome!).

2. Optionally, add one of the metrics that you have used before in MT evaluation, or any personal suggestion for a metric.

3. Using the test data provided by the organizers, apply the selected metrics and compute the scores of each translation, on a 0%–100% scale.
   The test data is described in the next document of the workbook and can be downloaded from `http://www.issco.unige.ch/projects/isle/mteval-may02/`. It consists in two source texts in French, each with a reference translation and about a dozen translations to be evaluated, from various systems and humans.

4. Send the results by email to the organizers (e.g., `Andrei.Popescu-Belis@issco.unige.ch`), together with any comments you believe useful.

5. Prepare a brief account of the evaluation (about 10–15 minute talk) to be presented at the workshop, for instance by first answering the question "what are the strongest and the weakest points in the measures that you used?"

1.3. Exploitation of the Results

The results of these evaluations will be discussed and highlighted at the workshop from the perspective of present research goals. Regarding individual metrics, the scores obtained by different evaluators using the same metric will inform the community about the reliability of that metric (cf. preceding document, 5.2), by computing standard deviation and inter-annotator agreement.

The other important result of the pre-workshop evaluations will be data on cross-metric correlation, i.e. the agreement between pairs of metrics. This is important both for metrics based on human judges (it illustrates how well the specifications are defined or how coherent the judges are) and for automated metrics (for which agreement with a reliable human judgement is almost the only proof of coherence). These meta-evaluation

considerations will be analyzed at the workshop by the organizers, based on the results sent to them by the participants. These considerations will constitute the basis for discussion and conclusions of the workshop.

2. Specifications of the Metrics

2.1. Preamble

The metrics that are proposed in this application illustrate a broad spectrum of those that were synthesized for the ISLE MT evaluation framework. The two categories identified below parallel of course the title of the workshop, "Human Evaluators Meet Automated Metrics". In the history of MT evaluation, given the difficulty of the task, most of the quality judgments, and later 'metrics', we carried on by humans. However, as explained in the previous chapter, the utility of automatic measures has always been clear: they provide cheap, quick, repeatable and objective evaluation. 'Objective' means here that the same translation will always receive the same score, as opposed to human judges that may have fluctuating opinions. However, since human judges are the final reference in MT evaluation, the results of automated metrics must correlate well with (some aspect of) human-based metrics.

The metrics specified below must of course be integrated in a broader view of evaluation, since none of them is sufficient to determine the overall quality of a system. As stated in the ISLE taxonomy, it is the desired context of use of the evaluated system that determines a 'quality model', namely a set of useful features, to which several metrics are associated. It is only the combination of these scores that provides a good view of the quality of the system in the given context.

Documentation about the metrics below (apart from the references quoted) can be found in several papers available over the Internet. The ISLE evaluation workgroup has a webpage at `http://www.issco.unige.ch/projects/isle/ewg.html`, with links to previous workshop material for MT Evaluation, and to electronic versions of Van Slype's (1979) report and of the MT Evaluation workshop held at the MT Summit VIII conference. The ISLE taxonomy can be found at `http://www.issco.unige.ch/projects/isle/taxonomy2/`.

Below is a synopsis of the metrics that will be described in the remaining part of this document.

(A1)	IBM's BLEU and the NIST version
(A2)	EvalTrans
(A3)	Named entity translation
(A4a)	Syntactic correctness
(A4b)	X-Score / parsability
(A5a)	Dictionary update / number of untranslated words
(A5b)	Translation of domain terminology
(A6)	Evaluating syntactic correctness from the implementation of transfer rules
(H1)	Reading time
(H2)	Correction / post-editing time
(H3)	Cloze test
(H4a)	Intelligibility / fluency
(H4b)	Clarity
(H5)	Correctness / adequacy / fidelity
(H6)	Informativeness: comprehension task

2.2. Automated/automatable metrics

2.2.1. IBM's BLEU and the NIST version (A1)

We mention first the most recent proposal of an automated metric for MT Evaluation, namely the BLEU algorithm proposed by a team from IBM (Papineni et al., 2001; Papineni, 2002). The principle of this metric, which was fully implemented, is to compute a distance between the candidate translation and a corpus of human "reference" translations of the source text. The distance is computed averaging *n*-gram similitude between texts, for $n = 1, 2, 3$ (higher values do not seem relevant). That is, if the words of the candidate translation, the bi-grams (couples of consecutive words) and tri-grams are close to one or more of those in the reference translations, then the candidate scores high on the BLEU metric.

Apart from intuitive arguments, the method to find out whether this metric really reflects translation quality is to compare its results with human judgements, on the same texts. In-house data (Papineni et al., 2001), as well as the DARPA 1994 data (Papineni et al., 2002), were used to test the coherence between human scores and BLEU scores, and this was found acceptable.

The metric was also adapted for the recent NIST MT Evaluation campaign (Doddington, 2001). The main changes were: text preprocessing, a differentiated weight associated to N-grams based on their frequency, and the use of tri-grams only. These modifications must still be discussed by the community, but the NIST provides yet the scripts implementing the BLEU metric as well as its adaptation, at: `http://www.nist.gov/speech/tests/mt/mt2001/resource/`.

We do not describe further this metric, but would like to refer the participants to the documentation quoted above, which provides enough resources to apply it.

2.2.2. EvalTrans (A2)

Automatic corpus evaluation extrapolation using EvalTrans (Niessen et al., 2000) gives statistics, such as the average Levenshtein distance standardized to the length of the target sentence. The tool can be downloaded at `http://www-i6.informatik.rwth-aachen.de/HTML/Forschung/Uebersetzung/Evaluation/`.

The first step is to load and save the human translations. For the present workshop, the reference translation as well as the other human translations of the same source text will constitute the "reference set". When the system is set up to work automatically, it will search this reference database for sentences which are most similar to the machine translated sentence that must be scored.

However, in order for the extrapolation to be performed, the Levenshtein distance algorithm needs to be seeded with scores for some (at least one) manually evaluated sentence. For this, a baseline machine translation (for instance) needs to be loaded and some sentence pairs need to be evaluated.

Next, the "test corpus" sentences need to be loaded. These are the machine translations for each source text. For each set of "test corpus" sentences, which comprise each machine translation of a source text, subjective sentence error rate (SSER) and multi-reference word error rate (mWER) will be calculated by the automatic metric.

- Several statistics of interest will be produced:
- Average number of "perfect" (scored 10) reference sentences per evaluation sentence pair (to indicate how reliable the mWER is).
- (average-score) / (value of all (evaluated/ extrapolated) sentence pairs)
- Standard deviation of the score
- Subjective sentence error rate (i.e., 100% * (1 – average-score)). An average score of 0.0 results in a SSER of 100%, an average score of 10.0 in a SSER of 0%.
- Subjective sentence error rate weighted by the length of the target sentences
- Average extrapolation distance: average Levenshtein distance (per target word) of all extrapolated sentences

The SSER indexes each sentence, then uses the mWER, the number of perfect reference sentences, the absolute Levenshtein distance to each sentence, and the Levenshtein distance to that sentence v. the length of current sentence.

The mWER is the word error rate against the most similar reference sentence which has been evaluated as "perfect" (i.e., has been assigned a score of ten). It is calculated as Levenshtein operations per reference word (and can thus exceed 100%). Average mWER for an evaluation corpus is calculated word-wise, not sentence-wise.

Another measure, the information item error rate, is not included because it relies heavily on manual scores, use of which would defeat the purpose of the automated metric.

2.2.3. Named entity translation (A3)

The NEE metric (Named Entity Evaluation) is described for instance in (Reeder et al., 2001). Since automated software to support this metric is available, it has been considered here an automated metric. Participants to the workshop may of course apply it manually, given the small amount of test data.

The process for utilizing this metric is relatively straightforward: a) identify the named entities within a given test corpus; b) pull unique entities from the document; c) find the entities in the system output text; and d) compare entities in the output text with those identified in the reference text (see Figure 1 below). Identifying the named entities in the reference translation requires human annotation, and is the only stage of the process to do so.

In a concrete example of this metric, to prepare the corpora for evaluation, two expert annotators used the Alembic Workbench (Day et al., 1997; see also `http://www.mitre.org/technology/alembic-workbench/`) annotation tool to tag occurrences of named entities according to the MUC annotation guidelines. After the named entities are tagged in the reference translation (designated here by ANNO), the metric can be applied.

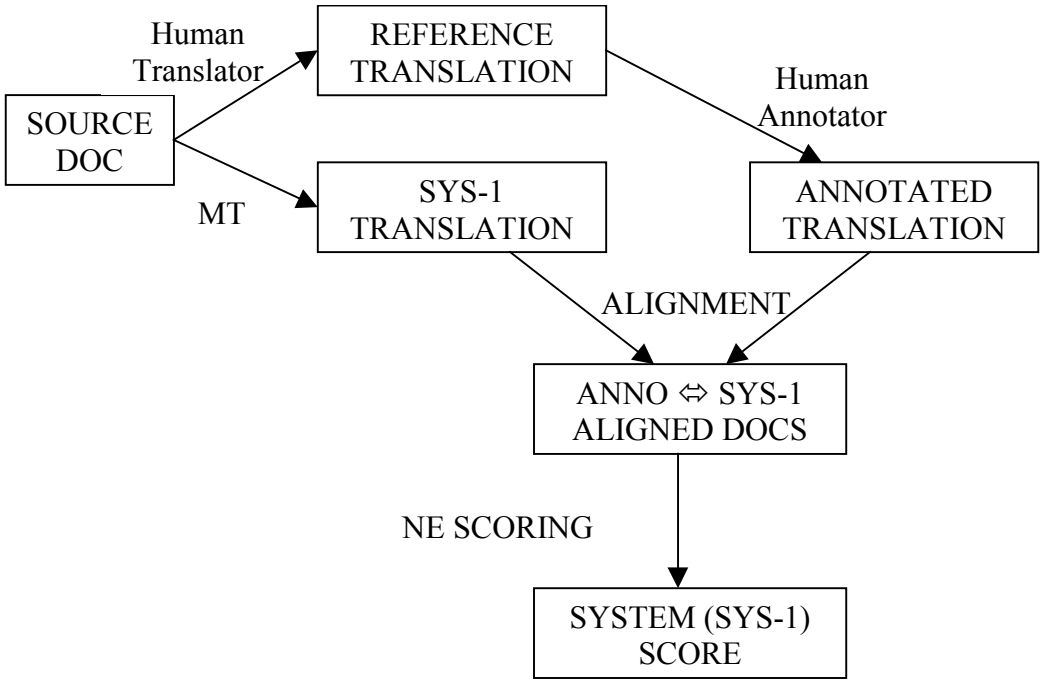


Figure 1. Scoring technique for the NEE metric.

The next stage is to align the ANNO translation text with the evaluation text (the output of the system SYS-1 for this example). To score the translation, for each article in the aligned pair, the tagged named entities are pulled from the ANNO and a list of unique names for the comparison unit (paragraph or article) is prepared. This is followed by normalization. At this time, the normalization steps applied are: (a) substitution of non-diacritic marked letters for the equivalent diacritic mark character for Romance languages (for instance **ã** becomes **a**); (b) down-casing; (c) the normalization of numeric quantities (particularly for numbers under 100) and (d) the removal of possessives. Other normalization steps may be needed, as well as the incorporation of partial match scoring (see Reeder et al., 2001). Once the named entity list and the SYS-1 tokens have been normalized, the search for named entities in the token lists is straightforward. Only exact matches given the normalization steps described are considered at this time and all results here reflect this.

2.2.4. Syntactic correctness (A4a)

The following describes a syntax metric based on the minimal number of corrections necessary to render an MT output sentence grammatical. Each evaluator must transform each sentence in the MT output into a grammatical sentence by making the minimum number of replacements, corrections, rearrangements, deletions, or additions possible. The syntax score for each sentence is then defined as the ratio of the number of changes for each sentence to the number of tokens in the sentence. For the purposes of this test, a token is defined as a whitespace-delimited string of letters or numbers. Additionally, individual punctuation marks, since they are subject to correction, are also counted as separate tokens. Each item of punctuation that occurs in pairs (e.g. brackets, braces, quotation marks, parenthesis) is counted as a separate token. Thus, in the following sentence, there are 24 tokens:

- *Mary, who had gone to see the fountain (in the center of town), said that it was turned off.*

It is important to remember that the final edited sentence need only be syntactically correct. That is, the final result may be semantically anomalous. Raters should endeavor to produce a syntactically correct sentence by making as few changes possible to the original MT output. Deletions, substitutions, additions, and rearrangements are counted by totaling the number of words deleted, substituted, added, or moved. In the event that there are combined operations, for example, moving a phrase consisting of four words, of which one has been deleted, the move is computed *after* the deletion is counted, thus the above-mentioned operation would result in one deletion and 3 moves. Finally, errors in inflectional morphology are not counted in the syntax metric. In applying this metric to test data, it was found that even when evaluators arrive at the same score for a given sentence (that is, they have the same total number of changes), they often choose a different combination of the four operations to arrive at their final grammatical sentence. The metric as it stands has not been automated, and would indeed be very difficult to automate; however, partial automation, such as automatic tracking and counting of necessary edit operations, would greatly assist in applying this metric in an efficient manner.

2.2.5. Automatic Ranking of MT Systems by X-Score (A4b)

Background: The X-Score metric aims to rank MT systems in the same order as would be given by a human evaluation of the Fluency of their outputs (Hartley & Rajman, 2001; Rajman & Hartley, 2002). The metric is especially adapted to rank machine translations relative to one another, rather than comparing human and machine translations. This metric was derived from experiments conducted on the French-English segment of the corpus used in the 1994 DARPA MT evaluation exercise. In that exercise, human evaluators scored translations of 100 source texts by 5 MT systems for their Fluency (among other attributes). To establish the present metric, the F-scores (Fluency scores) for individual texts were converted into rankings of systems using the aggregation technique of ranking by average ranks (average rank ranking or ARR). Using the same ARR technique, rankings were computed on the basis of the X-score for each document. The X-scores were found to represent a very good predictor of the ranking derived from the human evaluations (H-rankings). The distance between the H-ranking and the X-ranking is 1, corresponding to a similarity of 93.3%, a precision of 93,3% and a recall of 93.3%. If restricted to the most complete partial ranking, these values improve to a distance of 0.5, a similarity of 96.7%, a precision of 100% and a recall of 93.3%.

Computing the X-Score: The X-score is taken to measure the grammaticality of the translations. For any given document, the X-score is obtained as follows. First, the document is analyzed by the Xerox shallow parser XELDA in order to produce the syntactic dependencies for each sentence constituent. For example, for the sentence The Ministry of Foreign Affairs echoed this view, the following syntactic dependencies are produced: SUBJ (Ministry, echoed); DOBJ (echoed, view); NN (Foreign, Affairs); NNPREP (Ministry, of, Affairs).

On the corpus used in (Hartley & Rajman, 2001), XELDA produced 22 different syntactic dependencies, among which:

- RELSUBJ: for example, RELSUBJ(hearing, lasted) in “a hearing that lasted more than two hours”;
- RELSUBJPASS: for example, RELSUBJPASS(program, agreed) in “a public program that has already been agreed on ...”;
- PADJ: for example, PADJ(effects, possible) in “to examine the effects as possible”;
- ADVADJ: for example, ADVADJ(brightly, colored) in “brightly colored doors”.

After each document has been parsed, we compute its dependency profile (i.e. the number of occurrences of each of the 22 dependencies in the document). This profile is then used to derive the X-score using the following formula:

- *X-score = (#RELSUBJ + #RELSUBJPASS – #PADJ – #ADVADJ)*

Note that several formulae would have been possible for computing the X-scores. The above-mentioned one

was selected in such a way that, if applied to the average dependency profile, it correctly predicted the average rank ranking (ARR) derived from the F-scores. In this sense, one can say that the computation of the X-score was specifically tuned to the test data and so it was considered quite ad hoc in (Hartley & Rajman, 2001). However, this is not true of (Rajman & Hartley, 2002). This second experiment retained exactly the same formula for the X-scores, while completely changing the human evaluations – evaluators directly assigned rankings to series of translations instead of assigning individual scores to each of the translations. Moreover, a new MT system was added, not present at all in the data that was used for the tuning. Thus, there is no reason to believe the X-scores to be ad hoc, which strongly increases their chances of being highly portable to other experimental data.

Computing the Rankings: For each of the documents, the scores of the systems are first transformed into ranks and the average ranks obtained by the systems over all the documents are then used to produce the final ranking.

2.2.6. Dictionary update (A5a) and domain terminology (A5b)

Dictionary update (also known as *non-translated or untranslated words*) and *domain terminology* are two potentially automatable metrics. Although related, these two metrics are not identical, as can be seen from their descriptions below. There are many ways in which a dictionary update measure could be calculated, but it seems obvious to use two objective and easy to observe features of MT output:

- the number of words not translated;
- the number of domain-specific words that are correctly translated.

It is these two features that have been described in previous related work, including (Vanni & Miller, 2002), and that will be specified below.

2.2.7. Number of untranslated words (A5a)

This metric makes use only of the target text. It is based on the intuition that translation quality is linked to size of vocabulary. In its simplest form, the number of words left untranslated is counted. By untranslated, we mean simply that a word which should be translated is not, and is simply copied over untouched into the target text. (This reflects the behavior of many machine translation systems). There are, of course, words which should not be translated (most proper names are a good example): not translating these items is not counted as an error. A score is obtained by the following calculation:

- (number-of-untranslated-words) / (total-number-of-words-in-text) x 100 = percentage-of- untranslated-words… *high is bad*

One possible way to automate this metric would be to run a spelling checker over the target text and count the number of mistakes found. This would, of course, pick up any spelling mistakes in translated words which might exist, as well as finding words which were not legal words of the target language; however, this amount is probably low for translations programs, which generate words based on valid dictionaries. On the whole, this automatic measure might not invalidate the metric as an indicator of overall translation quality.

In discussing the automation of this measure, it is worth noting that some MT systems provide as ancillary output statistics concerning the numbers of untranslated words in the output. However, this is not the case for all systems. In these cases, other automated means must be developed for computing this measure. In cases of languages using a non-Roman script or containing characters outside the standard lower-ASCII range found in typical English text, one possible way of counting non-translated words (for systems that simply pass untranslated words through in the translation) would be to locate and count tokens containing these characters that do not appear in English text. However, even in the case of the Japanese-English systems, some systems did produce a romanization of the untranslated words, and did not leave them in the native script. The romanizations contained only characters found in the lower portion of ASCII.

Given that this metric is intended to compute the number of words that the MT system was unable to translate, another possibility would be to use a tool such as *ispell* in order to identify non-English strings within the output translation. Counting these strings and comparing with the output of a utility such as *wc* (Unix word count) could provide a ratio of untranslated words in the output text.

Two potential problems with this last approach could both lead to undercounting the number of untranslated words in a text. First, included in the untranslated word count for Japanese – English translation were Japanese particles and other bits of non-English material, which may or may not have been the result of romanization of text found in the source. Examples of this include *na*, *re*, *X*, and *inu*. Another Japanese particle, *no*, did not appear in this context in the translation, but had we relied on an automated spelling-based identification of untranslated words, words like *no*, which also happen to be valid English strings (although with a different meaning) would be left uncounted. Secondly, untranslated word scores would likewise be affected for languages that share a high number of cognates with English. For these languages, the string in the source and target language may be identical, and thus not counted as an untranslated word, regardless of whether the system actually translated the word or simply passed it through.

The application of this metric to translations produced by human translators is somewhat doubtful: human translators when faced by a gap in their lexical knowledge try to work round the problem, and do not, normally, simply transcribe the problematic word or leave a gap. It is possible though that the spelling mistake variation might be informative.

It is also worth noting that while untranslated words certainly have an impact on the usability of MT output, such output often contains sentences that are completely unintelligible, but in no way due to untranslated words. Thus, this test should clearly not be used in isolation to provide a picture of overall MT quality, whether quality is defined along the lines of clarity, fluency, adequacy, or coherence.

2.2.8. Translation of Domain Terminology (A5b)

The domain terminology score is calculated as the percentage of correctly translated pre-identified domain terms. The procedure for this test is as follows: First, a list of key term translations is extracted from the human translation. To accomplish this, raters individually select key terms from the human translation, and then the separate key term lists are reconciled before application of the test to the MT systems' output. This step is amenable to automation, but has not as yet been automated. During the test application, systems receive a point for each term for which the translation matches the human translation exactly, and no point otherwise. The final score is the percentage of exactly-matched translations of key terms.

There are two divergent directions in which this test could be developed in the future. First, it could be made more sensitive to acceptable variation in translation of key terms by application of the ACME Cloze test methodology as described for instance in Miller (2000). This methodology simulates basing lexical tests on multiple human translation, while sufficiently constraining the structure of the translation to enable automated comparison.

2.2.9. Evaluating syntactic correctness from the implementation of transfer rules (A6)

This metric proposal is the result of two previous studies. In the first former study, the authors chose to count the number of NPs (noun phrases) and VPs (verb phrases) in source text and target texts, a first indication being given by non parallel data (Mustafa El Hadi, Timimi, Dabbadie, 2001). Another study presented the results on the same corpus after terminological enrichment (Mustafa El Hadi, Timimi, Dabbadie, 2002).

Nevertheless, the use of finer grained criteria such as adjectives or prepositional phrases count could also be envisaged. Any overlap of this threshold might then be considered as an indication that MT system may have failed to analyze source syntactic structure and that therefore, the initial figures require further analysis. But this methodology is still imprecise and limited to a first indication of MT system's analysis failure, when a gap is observed on non parallel data. The use of this methodology also implies that the test is carried out on relatively syntactically isomorphic languages such as French and English. A methodology including a test tool that would implement source and target transfer rules might probably prove more accurate and also apply to non isomorphic languages.

We propose here the following steps for the application of the metrics:

1. Deduce a set of French / English transfer rules from the source text and the reference translation (this part involves manual processing).
2. Write a script (e.g., in Java or Perl) to implement these rules (if not, go to point n. 3)
3. Check that these rules apply through the various candidate translations from the test data (automatically with the script or manually).
4. Generate an output failure file (or else carry out a manual check) and work out syntactic correctness.

2.3. Human-based measures

2.3.1. Reading time (H1)

Reading time can be defined in one of two ways: oral reading time or closed reading time.

Oral reading time (Van Slype, 1979) tends to measure more closely with intelligibility and also tends to be more relevant to higher quality translations. Therefore, for each document, the evaluators should read out loud the first paragraph and time the length of time that it takes to read each sample. The number of words then can be used to calculate a words per minute (WPM) rate:

- *WPM = number-of-words / reading-time*

The closer the WPM rate is to the WPM of natural language (depending on the evaluator), the higher is the quality of the translation (on a scale to be defined by each participant).

Closed reading time relates to the amount of time that a user needs to read a document to a "sufficient" level of understanding. The sufficient level is often paired with other measurements such as comprehension score on a test. Still, the instructions can be given that the readers measure the amount of time necessary to arrive at an understanding they consider to be sufficient to answer basic questions about the text. Words-per-minute rate can be calculated in the same way.

2.3.2. Correction / post-editing time (H2)

This metric is based on the intuition that the time required to produce an acceptable translation from a raw translation (whether produced by a human or by a machine) is inversely proportional to the overall quality of the raw translation.

It can be measured fairly easily by noting when the person responsible for the revision/post-editing starts their task and when they finish it, normalizing the result by taking into account the size of the text measured in words, then multiplying by a fixed factor in order to obtain a number on a wider scale. For this exercise, the following calculation is suggested:

- (number-of-minutes-spent-in-correction) / (total-number-of-words-in-text) x 10 = correction-time… *high is bad*

Note that this metric can only sensibly be applied to a whole text: timing correction to smaller text elements is both annoying for the person doing the timing and difficult to do reliably.

A variation on this metric is to count not the overall time but the number of key strokes made by the corrector.

It should be noted that this metric is somewhat problematic both with respect to validity and reliability for a number of reasons:

- The amount of correction needed depends in part on the ultimate use to which the translation will be put: a text destined for publication will probably be treated with more care than a text intended for information assimilation, for example
- The errors corrected differ in their nature. There will be straightforward grammatical or lexical errors, as well as more complicated stylistic errors. This will affect the amount of time needed to carry out the correction. This would not matter

so much if those doing the correction always agreed on what corrections are needed. But, inevitably, where matters of style are concerned, no such agreement exists.
- There is considerable variety amongst correctors and the way they work. Some work quickly and decisively, others are more hesitant and sometimes change their minds.
- Correctors may be influenced by knowing whether they are dealing with a human produced translation or a machine produced translation. One anecdote tells of correctors correcting far more on machine produced translation but spending comparatively less time in doing so because they felt no need to take into account the computer's feelings.

Participants who choose to work with this metric are invited to reflect on these issues and on possible improvements to the simple metric defined here.

2.3.3. Cloze test (H3)

This metric is reported by Van Slype (1979) as a test of readability. It may however also be thought of as a test of fidelity or of intelligibility, since it is based on the ability of a reader to supply a missing word correctly, which intuitively relates both to readability and intelligibility when the target text alone is considered and to fidelity when the source text is taken into account.

The method is simple. Every *n*-th word in the translation is deleted (in the Van Slype Report (1979), $n = 8$, but other values appear also in the literature). The translation is then given to a group of readers, who are asked to supply the missing words. Two scores are normally computed, one based on the number of answers which comprise exactly the suppressed original word, the other based on the number of answers with a word close in meaning to the original word. The second score has to be interpreted partly in the light of the first score

- (number-of-exact-answers) / (number-of-deleted-items) x 100 = percentage-of-exact-items-supplied… *high is good*

- (number-of-close-answers) / (number-of-deleted-items – number-of-exact-items-supplied) x 100 = percentage-of-close-items-supplied… *high is good*

A possible weakness of this metric is that it potentially also tests the intelligence and wealth of vocabulary of the reader supplying the missing words. This weakness can be mitigated by controlling the size and type of the group of readers.

A second possible weakness appears if the translated text is technical in nature: the readers have to have sufficient knowledge of the subject matter to make it plausible that they should be able to supply the missing items.

Van Slype (1979) also points out that some texts are more redundant than others in the way they carry information, and that if translations of several texts are to be compared, it is important to take this factor into account. He suggests that this can be done by carrying out a Cloze test also on the original text.

2.3.4. Intelligibility / fluency (H4a)

Intelligibility is one of the most frequently used metrics of the quality of output. Numerous definitions (or protocols for measuring it) have been proposed for it, for instance in Van Slype's report or in the DARPA 1994 evaluations. We outline here the definition proposed by T.C. Halliday in (Van Slype, 1979, p. 70), which measures intelligibility on a 4-point scale (0 to 3).

Intelligibility or comprehensibility expresses how intelligible is the output of a translation device under different conditions (for instance, when the sentence fragments are translated while being entered, or after each sentence). Comprehensibility reflects the degree to which a complete translation can be understood. Intelligibility can be based on the general clarity of translation, or the output can be considered in its entirety or by segments out of context.

The following scale of intelligibility has been proposed, from 3 to 0, 3 being the most intelligible:

- 3 – Very intelligible: all the content of the message is comprehensible, even if there are errors of style and/or of spelling, and if certain words are missing, or are badly translated, but close to the target language.
- 2 – Fairly intelligible: the major part of the message passes.
- 1 – Barely intelligible: a part only of the content is understandable, representing less than 50% of the message.
- 0 – Unintelligible: nothing or almost nothing of the message is comprehensible

To apply the metric, the following steps are suggested:

1. Take the reference translation of a text (or the source if you are proficient in that language).
2. Separate and number the sentences.
3. Take a candidate translation and do the operation (2) on it. Match sentences with those in the reference/source translation.
4. Rate sentences from the candidate translation using the 0 to 3 scale described above.
5. Optional: to normalize scores, calculate intelligibility on a 0% to 100% scale, by averaging sentence ratings over the whole text.
6. Produce a final score for each translation

2.3.5. Clarity (H4b)

In work described in (Vanni & Miller, 2002) a metric called *clarity* is proposed that merges the ISLE categories of comprehensibility, readability, style, and clarity into a single evaluation feature. This measure ranges between 0 and 3. Raters are tasked with assigning a *clarity* score to each sentence according to the following criteria:

Score	Criterion
3	meaning of sentence is perfectly clear on first reading
2	meaning of sentence is clear only after some reflection
1	some, although not all, meaning is able to be gleaned from the sentence with some

effort

0 Meaning of sentence is not apparent, even after some reflection

Since the feature of interest is clarity and not fidelity, it is sufficient that some clear meaning is expressed by the sentence and not that that meaning reflect the meaning of the input text. Thus, no reference to the source text or reference translation is permitted. Likewise, for this measure, the sentence need neither make sense in the context of the rest of the text nor be grammatically well-formed, since these features of the text would be measured by tests proposed elsewhere, namely the *coherence* and *syntax* tests, respectively. Thus, the clarity score for a sentence is basically a snap judgement of the degree to which some discernible meaning is conveyed by that sentence.

2.3.6. Correctness / adequacy / fidelity (H5)

This evaluation metric reprises the DARPA 1994 *adequacy* test (Doyon, Taylor, and White, 1996). As with that test, the reference translation or "authority version" is placed next to each of the translations of the source text, to be used as a comparison against each one, human or machine. Before the test is performed, both the "authority version" as well as each of translations should be segmented, with each text separated into sentence fragments to appear next to the corresponding fragment in the translation.

Once each translation is lined up with its equivalent, evaluators grade each unit on a scale of one to five, where five represents a paragraph containing all of the meaning expressed in the corresponding text. The *Adequacy* scale is as follows:

- 5 – All meaning expressed in the source fragment appears in the translation fragment
- 4 – Most of the source fragment meaning is expressed in the translation fragment
- 3 – Much of the source fragment meaning is expressed in the translation fragment
- 2 – Little of the source fragment meaning is expressed in the translation fragment
- 1 – None of the meaning expressed in the source fragment is expressed in the translation fragment

2.3.7. Informativeness: comprehension task (H6)

There are two methods for testing comprehension. The most common of these is the reading comprehension exam (e.g., Somers & Prieto-Alvarez, 2000; DARPA-94; Tomita 1992). In this case, the evaluators design a set of questions, usually under 10, for the given texts. Sometimes, as in the case of Tomita, these tests are structured first and then applied to the translations. Tomita began with the Test of English as a Foreign Language (TOEFL) examinations which he then translated to Japanese and had students take. The theory being that the better scores on the exam will have resulted from the better translations. The big difficulty (Somers & Prieto-Alvarez, 2000) is that it is difficult to test only the reading without bringing a large amount of pre-existing world knowledge to the table. In addition, the design and structuring of such examinations is an art in and of itself.

The second method for a comprehension test takes instead the task of figuring out the kinds of questions that one might want to be able to answer from a translation and determining whether the translation can support answering said questions. For instance, one might want to know the people, places and organizations mentioned in an article. This is covered by the named entity metric. Yet, it is really only the first stage of measurement. The secondary measure would be to look to determine if the entity relationships are also preserved by the translation - that is, who belongs to what organization or who did what to whom. This is the question we began to study at MT Evaluation workshop organized at NAACL 2001, when we asked participants to fill in templates based on specific kinds of questions. The better systems would enable the successful template filling and scoring would follow Message Understanding (MUC) guidelines. It is this type of exercise you will be asked to do at this time. The previously identified named entities will be used here. You will fill out templates to answer specific details of events or relationships between parties.

3. References

D. Day, J. Aberdeen, L. Hirschman, R. Kozierok, P. Robinson, and M. Vilain. 1997. Mixed-Initiative Development of Language Processing Systems. In *Fifth Conference on Applied Natural Language Processing*, Washington, D.C.

G. Doddington. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. In *HLT 2002, Human Language Technology Conference*, San Diego, CA.

J. Doyon, K. Taylor, and J.S. White. 1998. The DARPA MT Evaluation Methodology: Past and Present. In *Proceedings of the AMTA Conference*, Philadelphia, PA.

A. Hartley and M. Rajman. 2001. Automatically Predicting MT Systems Rankings Compatible with Fluency, Adequacy or Informativeness Acores. In *Workshop on MT Evaluation "Who did what to whom?" at MT Summit VIII*, Santiago de Compostela, Spain, p.29-34. See `http://www.issco.unige.ch/projects/isle/MT-Summit-wsp.html`.

K. J. Miller. 2000. *The Machine Translation of Prepositional Phrases*. Unpublished PhD Dissertation. Georgetown University. Washington, DC.

W. Mustafa El Hadi, I. Timimi and M. Dabbadie. 2001. Setting a Methodology for Machine Translation Evaluation. In *Workshop on MT Evaluation "Who did what to whom?" at MT Summit VIII*, Santiago de Compostela, Spain, p.49-54. See `http://www.issco.unige.ch/projects/isle/MT-Summit-wsp.html`.

W. Mustafa El Hadi, I. Timimi, and M. Dabbadie. 2002. Terminological Enrichment for non-Interactive MT Evaluation. In *LREC 2002, Third International Conference on Language Resources and Evaluation*, Las Palmas, Canary Islands, Spain.

S. Niessen, F.J. Och, G. Leusch, H. Ney. 2000 An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research. In *LREC 2000, 2nd International Conference on Language Resources and Evaluation*, Athens, Greece, pp. 39-45.

K. Papineni. 2002. Machine Translation Evaluation: N-grams to the Rescue. In *LREC 2002, Third*

International Conference on Language Resources and Evaluation, Las Palmas, Canary Islands, Spain.

K. Papineni, S. Roukos, T. Ward, J. Henderson, and F. Reeder. 2002. Corpus-based Comprehensive and Diagnostic MT Evaluation: Initial Arabic, Chinese, French, and Spanish Results. In *HLT 2002, Human Language Technology Conference*, San Diego, CA.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. 2001. BLEU: a method for automatic evaluation of MT. Research Report, Computer Science RC22176 (W0109- 022), IBM Research Division, T.J.Watson Research Center, 17 September 2001. See `http://domino.watson.ibm.com/library/CyberDig.nsf/home`, and search for 'RC22176'.

M. Rajman and A. Hartley. 2002. Automatic Ranking of MT Systems In *LREC 2002, Third International Conference on Language Resources and Evaluation*, Las Palmas, Canary Islands, Spain.

F. Reeder, K.J. Miller, J. Doyon, and J.S. White. 2001. The Naming of Things and the Confusion of Tongues: an MT Metric. In *Workshop on MT Evaluation "Who did what to whom?" at MT Summit VIII*, Santiago de Compostela, Spain, p.55-59. See `http://www.issco.unige.ch/projects/isle/MT-Summit-wsp.html`.

H. Somers and N. Prieto-Alvarez. 2000. Multiple Choice Reading Comprehension Tests for Comparative Evaluation of MT Systems. In *Workshop on MT Evaluation at AMTA-2000*.

M. Tomita. 1992. Application of the TOEFL Test to the Evaluation of Japanese-English MT. In *MT Evaluation Workshop at AAMT*.

G. Van Slype. 1979. Critical Study of Methods for Evaluating the Quality of Machine Translation. Technical Report BR19142, Bureau Marcel van Dijk / European Commission (DG XIII), Brussels. See `http://issco-www.unige.ch/projects/isle/van-slype.pdf`.

M. Vanni and K. J. Miller. 2002. Scaling the ISLE Framework: Use of Existing Corpus Resources for Validation of MT Evaluation Metrics across Languages. In *LREC 2002, Third International Conference on Language Resources and Evaluation*, Las Palmas, Canary Islands, Spain..

Towards a corpus of corrected human translations

Andrei Popescu-Belis, Margaret King, Houcine Benantar

ISSCO/TIM/ETI, University of Geneva,
École de Traduction et d'Interprétation
40 Bvd. du Pont d'Arve
CH-1211 Geneva 4 – Switzerland
andrei.popescu-belis@issco.unige.ch
margaret.king@issco.unige.ch
benanta4@etu.unige.ch

Abstract

This section of the workbook describes the test data that is proposed to the participants. The data is part of a broader-scope corpus containing translations produced by students and corrected by their professors. Such a corpus will be used in automatic evaluation of MT systems. This section describes the structure of the corpus and provides some sample data. The full workshop data can be downloaded from: `http://www.issco.unige.ch/projects/isle/mteval-may02/`.

1. Introduction

Several automatic measures for MT evaluation have been proposed, and computational tools to carry them on effectively are now available. From Henry Thompson's (1992) proposal to IBM's BLEU, through Niessen et al.'s (2000) proposal and NIST's 2001 MT Evaluation, all of these measures make heavy use of large sets of reference data (or golden standard).

It is indeed acknowledged that, while a unique "correct translation" of a source is insufficient for evaluation (since another perfectly acceptable translation can differ substantially from the first one), the solution may reside in the use of a set of reference translations, which will hopefully encompass the range of possible variations among acceptable translations. Once such a set available, the quality of candidate translations can be judged with respect to it, by automatically computing a similarity distance between the candidate and the set. Evaluation is thus greatly accelerated.

However, producing such resources is quite expensive. A team of professional translators must be hired and asked to translate a number of reference texts. The quality of the reference translations thus produced would be high, but maybe some more simplistic formulations, acceptable from an MT system, would not be present in the corpus, thus biasing the results.

We propose here to build a corpus of translations using translations exams from the Ecole de Traduction et d'Interprétation (University of Geneva). These translations are encoded using markup, together with the corrections made by professors, and most important, with the *grade* that has been decided. We describe below this construction effort, than describe the data that will be used in the LREC 2002 MT Evaluation Workshop.

2. Description of the corpus

2.1. Structuring the data

One of the principles underlying the encoding of the data is to encode the most part of the information present on the paper version of the exam. This includes mainly the text produced by each student, the corrections added by the professors grading the exam, and the final grade.

We chose an XML-based annotation format, with one file per translation. Each file has a header containing useful data (except the name of the student, who is never typed in), and a <content> element with the translation. Instead of giving the DTD that was written, here is an example of exam file.

```
<?xml version="1.0"
      encoding="iso-8859-1"
      standalone="no" ?>
<!DOCTYPE exam SYSTEM "exam.dtd">
<exam>
  <header>
    <index>101</index>
    <author>101</author>
    <date>11/02/2002</date>
    <source-language>en</source-language>
    <target-language>fr</target-language>
    <level>2e cycle (years 3-4)</level>
    <exam-title>Traduct. FR/EN</exam-title>
    <comments>Exam graded by two
independent reviewers. This is a non-native
English speaker. Teacher's comments: "Your
style was confident, your English
idiomatic. Only minor mistakes appear in
the flow of your translation. Good work."
    </comments>
    <grade max="6.0" pass="4.0">5.0</grade>
  </header>
  <contents>
    <title-zone>
      <s>...</s>
    </title-zone>
    <p>
    <s>...</s>
    ...
    </p>
  </contents>
</exam>
```

Figure 1. Example of translation header.

Together with the DTD, we also use tools to validate each XML file, as well as a simple XSL file (stylesheet) that extracts the original text and discards the markup (this stylesheet is used to produce the workshop data described in the next section).

The innovative part of this corpus of "imperfect" translations is the encoding of the mistakes, together with their corrections. This requirement renders the typing of the data a bit more tedious, but increases the value of the resource, since the erroneous fragments of the texts can be discarded (or given a lower weight) when computing the distance between a candidate translation and the corpus.

Several conventions have been used to encode the mistakes and their correction: the <*m*> tag denotes a mistake, and the attributes encode its correction. The 't' attribute encodes the type, as noted by the professor ('–' means a fragment to be deleted), while the 'w' attribute encodes the replacement string. Missing parts are encoded as an empty <*m/*> element, with t="miss" and w="the missing string". A sample corrected paragraph is shown below.

```
<p>
   <s>Just like you, we feel convinced
   that the prevention of drug addiction
   <m t="-" w="none">s</m> starts at
   home, through <m t="-">the</m> <m
   t="miss" w="a good"/> <m t="w"
   w="relationship">relation</m> between
   adults and children, by strengthening
   self-esteem.</s>
   <s>The findings of recent studies
   clearly show that the earlier the
   prevention, the <m t="gr" w="more">
   most</m> efficient it is.</s>
</p>
<p>
   <s>You do not necessarily need to be a
   specialist in drug addiction <m t="-">
   s</m> to talk over this issue with
   your children.</s>
   <s> The most important thing <m t="-"
   w="is">lies in</m> dialog, <m t="-">
   in</m> attentive listening, <m t="-">
   in</m> reciprocal confidence.</s>
</p>
```

Figure 2. Translated paragraph and annotated mistakes.

2.2. Present state of the corpus

The corpus presented above is still under construction. As members of the Translation Faculty at the University of Geneva, we have been granted access to the written examinations of translations students (anonymized). We are focusing, for this corpus, on pure translations: the students are required to produce, in a limited amount of time and without dictionary, a translation of a piece of text – in general an excerpt from an article or essay, broadly speaking with a "general" vocabulary (through more specific exams, such as law translation, do exist).

Several language pairs are tested for at our faculty. The best represented ones, in terms of number of translations, are translations from English into French. However, given that a majority of researchers focuses on translation *into* English, we collect also French-to-English translations (less numerous).

The quality level of these translations is quite variable, as well as the difficulty of the source text. A considerable part of the corpus comes from entry-level examinations, but there are also translations from students that are close to graduation; in this case, the source texts are more "difficult" (a notion that must still be quantified).

The corrections are done on the paper version by two graders, teachers of the faculty. Their annotations are by no means standardized, but we attempt to grasp them in the most precise manner using the annotation format described above. The encoding principle is that *stripping a text from its XML annotation must yield exactly the text produced by the candidate.* The consistency and correction of the typed texts are checked by a second annotator, and the validity of the XML mark-up is checked against the DTD using a parser (Xalan-Java).

For the time being, a total of about 50 translations of two texts have been encoded. The public distribution of this data is still under consideration.

2.3. Possible uses of the corpus

The construction of this corpus is part of a long-term effort in MT evaluation at ISSCO/TIM/ETI, University of Geneva. The main use of the corpus is as a resource for automatic evaluation, where the cost of the resource lies in typing and encoding the data, rather than asking professional translators to translate a given source text. Given that this is a corpus of "imperfect" translations, we must encode also the corrections that were made by the graders (teachers). This increases the reliability of the corpus when used for automatic evaluation, since the erroneous fragments of the student translations can be discarded or given less confidence. The grades obtained by each translation can also be used to modulate the confidence attributed to each translation.

The corpus can also be used, of course, to extract statistics about the types of translations mistakes, and the correlation between the distribution of mistakes in a translation and the grade scored by that translation. Of course, the corpus could serve also to explore automatic techniques to grade human translations, which differ quite strongly from machine translations (translation quality, proximity to source structures, etc.).

3. Description of test data for the workshop

For the present workshop, the organizers provide test data consisting in two sets of translations extracted from the corpus, enriched with machine translations of the same text. The test data is available at the workshop's site: `http://www.issco.unige.ch/projects/isle/mteval-may02/`.

- The source texts (*10S.txt* and *20S.txt*) are excerpts from two longer essays, originally in French – the source is of course provided, as well as a reference translation for each text (*10A.txt* and *20A.txt*) constructed from the best student translations, using also the teacher's corrections. Of course, these aren't meant to be "the perfect translation", but only correct translations that are close enough to the source text to help evaluators that do not understand French

For each of the two source texts, we provide about a dozen translations in English, some of them by translation students and some by commercial systems available over the Internet. Translations are numbered *101.txt* through *113.txt* and *201.txt* through *213.txt* (three numbers are missing from the second list, for technical reasons). There is no particular order, and in particular *1XY.txt* vs. *2XY.txt* are not necessarily translated by the same translator (human or system).

The human translators were not instructed to use either of the particular varieties of English (British vs. American), hence some slight spelling variations. The systems were simply those made available over the Internet by various providers, as listed for instance on the following page, compiled by Laurie Gerber: `http://www.lim.nl/eamt/resources/`. We do not wish to disclose the names of the systems that produced the various translations, since the evaluations produced in this workshop do not claim commercial-level reliability.

A sample of the translations produced for the first text (including source and reference) is provided for visual comparison in the table below.

Subject to availability, and depending on decisions that will be made after the time of writing, extra data will be made available at the workshop's website (`http://www.issco.unige.ch/projects/isle/mteval-may02/`), and the participants will be informed as soon as possible about updates.

<table>
<tr><td colspan="2">Source text
Comme vous, nous sommes convaincus que la prévention des toxicomanies commence dans la famille, dans la relation entre adultes et enfants, à travers le renforcement de l'estime de soi.
Les résultats d'études récentes le démontrent clairement : plus la prévention commence tôt, plus elle est efficace.
Il n'est pas forcément nécessaire d'être un spécialiste des toxicomanies pour aborder ce sujet avec vos enfants.
L'essentiel est ailleurs, dans le dialogue, dans l'écoute, la confiance réciproque.</td></tr>
<tr><td colspan="2">Reference translation
Like you, we are convinced that the prevention of dependence begins at home, through the relationship between adults and children. This is done through reinforcing the child's self-esteem.
The findings of recent studies clearly show that the earlier prevention starts, the more efficient it will be.
You do not necessarily need to be an expert in drug dependence to talk about this issue with your children.
What really matters is talking together, listening to each other, and having mutual confidence in one another.</td></tr>
<tr><td>Translation 101
Just like you, we feel convinced that the prevention of drug addictions starts at home, through the relation between adults and children, by strengthening self-esteem.
The findings of recent studies clearly show that "the earlier the prevention, the most efficient it is."
You do not necessarily need to be a specialist in drug addictions to talk over this issue with your children.
The most important thing lies in dialog, in attentive listening, in reciprocal confidence.</td><td>Translation 108
As you, we are convinced that the prevention of the drug addiction begins in the family, in the relation among adults and children, through the intensification of the respect of one.
The results of recent studies demonstrate him(it) clearly: the more the prevention begins early, the more it is effective.
It is not necessarily necessary to be a specialist of the drug addiction to approach this subject with your children.
The main part is somewhere else, in the dialogue, in the listening, the mutual confidence.</td></tr>
<tr><td>Translation 102
One thing is sure, we both agree: prevention of drug addiction starts at home, through the relationships between adults and children where the self-esteem has to be strengthened.
Outcomes of recent studies carried out recently, clearly demonstrate that the sooner the prevention begins, the better and the more successful it will be.
You needn't be a specialist in drugs to talk about it with your children.
It is necessary to listen to them, you must establish a real dialogue based on reciprocal confidence.</td><td>Translation 109
As you, we are convinced that the prevention of the drug addiction begins in the family, in the relation between adults and children, through the intensification of the self-respect.
The results of recent studies demonstrate him(it) clearly: the more the prevention begins early, the more it is effective.
It is not necessarily necessary to be a specialist of the drug addiction to approach this subject with your children.
The main part is somewhere else, in the dialogue, in the listening, the mutual confidence.</td></tr>
<tr><td>Translation 103
Like you, we are convinced that drug prevention begins within the family, in the relationship between grown-ups and children, through the encouragement of</td><td>Translation 110
Like you, we are convinced that the prevention of drug-addiction starts in the family, in the relation between adults and children, through the reinforcement</td></tr>
</table>

self-esteem. Recent studies have clearly shown that the earlier the prevention begins, the more efficient it is. It is not unavoidably necessary to be a specialist in drug addictions to talk about this subject with your children. What matters more is discussion, attentive listening and mutual trust.	of the regard of oneself. The results of the recent studies show it clearly: the more the prevention starts early, the more it is effective. It is not inevitably necessary to be a specialist in drug-addiction to tackle this subject with your children. Essence is elsewhere, in the dialogue, in listening, reciprocal confidence.
Translation 104 Like you, we are convinced that the prevention of dependences begins at home, through the relationship of parents with their children. This is done through the reinforcement of the child's self-esteem. As recent studies have clearly shown, the earlier prevention starts, the more efficient it will be. You do not necessarily need to be an expert in dependences to talk about this issue with your children. What really matters is talking together, listening to each other, and having confidence in one another.	*Translation 111* Like you, we are convinced that the prevention of drug-addiction starts in the family, in the relation between adults and children, through the reinforcement of the regard of oneself. The results of the recent studies show it clearly: the more the prevention starts early, the more it is effective. It is not inevitably necessary to be a specialist in drug-addiction to tackle this subject with your children. Essence is elsewhere, in the dialogue, in listening, reciprocal confidence.
Translation 105 Like you, we are convinced that prevention starts at home: the relationship between parents and children as well as the child's self-esteem are of great importance. Recent studies have shown very clearly that the earlier prevention starts, the more effective it will prove. You do not necessarily need to be an expert in addictions to talk about that issue with your children. Exchanging thoughts, listening to each other as well as mutual trust is much more important.	*Translation 112* As you, we are convinced of the prevention of the drug addictions beginning in the family, in the relationship between adults and children, through the reinforcement of the esteem of themselves. The results of recent studies demonstrate it clearly : the earlier the prevention begins, the more efficient it is. Him n ' is not inevitably necessary of to be a specialist of the drug addictions to approach this subject with your children. The essential is elsewhere, in the dialogue, in the listening, the reciprocal trust.
Translation 106 Like you, we are convinced that the prevention of drug addiction begins within the family, in the relationship between adults and children, through the reinforcement of self-confidence. Recent study results show this clearly: the earlier the prevention starts, the more efficient it is. It is not completely necessary to be a specialist on drug addiction to discuss this subject with your children. The importance is elsewhere: it is in the discussion, in the listening, in the mutual confidence.	*Translation 113* Like you, we are convinced that the prevention of drug-addiction starts in the family, in the relation between adults and children, through the reinforcement of the regard of oneself. The results of the recent studies show it clearly: the more the prevention starts early, the more it is effective. It is not inevitably necessary to be a specialist in drug-addiction to tackle this subject with your children. Essence is elsewhere, in the dialogue, in listening, reciprocal confidence.
Translation 107 As you, we are convinced that the prévention of the toxicomanies begin in the family, in the relation between adults and children, through the reinforcement of the esteem of oneself. The results of recent studies show it clearly: more the prévention begin early, more she is effective. It is not necessarily necessary be a specialist of the toxicomanies to approach this subject with your children. The essential is elsewhere, in the dialog, in the listen, reciprocal confidence.	

Figure 3. Excerpt from the test data: source text (French), reference translation, candidate translations from humans and from commercial systems available over the Internet.

The references of the two source texts are the following:

- Excerpts from the brochure “Prévenir ses enfants des problèmes de drogue”, Institut Suisse de Prévention de l’Alcoolisme et Autres Toxicomanies (ISPA), 24 p., 1999. (Free, order at *http://www.sfa-ispa.ch*
- Micheline Centlivres-Demont, “Hommes combattants, femmes discrètes : aspects des résistances subalternes dans le conflit et l’exil afghan” (p.169-182, excerpt at p. 178). In “Hommes armés, femmes aguerries : rapports de genre en situations de conflit armé”, Fenneke Reysoo, editor, DDC/Unesco/IUED, Geneva, 2001, 250 p.
  Proceedings of a colloquium held at the Institut Universitaire des Études du Développement, Geneva, 23-24 January 2001.
  Available freely at the IUED’s press service or at: http://www.unige.ch/iued/new/information/publications/yp_tm_hommes_armes_femmes.html).

4. References

G. Doddington. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. In *HLT 2002, Human Language Technology Conference*, San Diego, CA.

K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. 2001. BLEU: a method for automatic evaluation of MT. Research Report, Computer Science RC22176 (W0109-022), IBM Research Division, T.J.Watson Research Center, 17 September 2001. See `http://domino.watson.ibm.com/library/CyberDig.nsf/home`, and search for ‘RC22176’.

H. S. Thompson, ed., 1992. The Strategic Role of Evaluation in Natural Language Processing and Speech Technology. Record of a workshop sponsored by DANDI, ELSNET and HCRC, University of Edinburgh, Technical Report, May 1992.

Preface to the Workshop Proceedings

The ability to identify and analyse temporal information is important for a variety of natural language applications, such as information extraction, question answering, and multi-document summarisation. Nevertheless, this area of research has been relatively unexplored. It became evident during an ACL 2001 workshop on temporal and spatial information processing that that some research institutions have started to work on different aspects of temporal information, but no consensus has been achieved yet on what and how temporal information should be identified in text.

This workshop aims to provide a forum for researchers to present their work in this field and to discuss future developments such as building shared resources e.g. temporally annotated corpora. It is timely to coordinate the effort being undertaken in the community at this stage of research into temporal information.

We received 11 submissions and accepted 8 of them. The quality of the papers was very high and the topics diverse within the boundaries of the workshop theme. The selection process was therefore difficult. We would like to thank the members of the programme committe for their time and effort during the evaluation stage and for their valuable comments to the authors of the submitted papers.

The workshop will comprise presentation of each author and two talks by invited speakers. This will be followed by a panel discussion session at the end of the workshop. We would like to thank the invited speakers and the authors in advance for their participatation and we are looking forward to a hopefully enjoyable and enriching workshop which will contribute to this interesting and important emerging field within the natural language processing community.

Andrea Setzer, University of Sheffield
Robert Gaizauskas, University of Sheffield
(proceedings editors)

The Workshop Programme

09:15 - 09:30	Introduction
09:30 - 10:00	Andrew Salway
	Temporal Information in Collateral Texts for Indexing Movies
10:00 - 10:30	David Bree
	Features required to distinguish between temporal uses of the preposition FOR
10:30 - 11:00	Eleni Galiotou
	A Representational Scheme for temporal and causal Information Processing
11:00 - 11:30	coffee break
11:30 - 12:00	Joel Tetreoult
	Tense and Implicit Role Reference
12:00 - 12:30	Estele Saquete
	Recognising and Tagging Temporal Expressions in Spanish
12:30 - 13:00	Invited speaker: Stephanie Strassel
	Temporal Annotation and Relation Tagging for Automatic Content Extraction
13:00 - 14:30	lunch
14:30 - 15:00	Rich Campbell
	A Language-Neutral Representation of Temporal Information
15:00 - 15:30	Andrea Setzer
	On the Importance of Annotating Temporal Event-Event Relations in Text
15:30 - 16:00	Jerry Hobbs
	Towards an Ontology for Time for the Semantic Web
16:00 - 16:30	Invited speaker: James Pustejovsky
	TimeML: Time and Event Recognition for Question Answering
16:30 - 17:00	coffee
17:00 - 18:00	panel discussion

Workshop Organisers

Bill Black, UMIST, UK
Rob Gaizauskas, University of Sheffield, UK
Graham Katz, University of Osnabrueck, Germany
Andrea Setzer, University of Sheffield, UK
George Wilson, the MITRE Corporation, USA

Workshop Programmme Committee

Bill Black, UMIST, UK
Rob Gaizauskas, University of Sheffield, UK
Graham Katz, University of Osnabrueck, Germany
Inderjeet Mani, the MITRE Corporation and Georgetown University, USA
Allan Ramsay, UMIST, UK
James Pustejovsky, Brandeis University, USA
Frank Schilder, University of Hamburg, Germany
Andrea Setzer, University of Sheffield, UK
Beth Sundheim, SPAWAR Systems Center, USA
Janyce Wiebe, University of Pittsburgh, UK
George Wilson, the MITRE Corporation, USA

Table of Contents

Papers

Author Index

Features required to distinguish between temporal uses of the preposition FOR

David S. Brée

Computer Science Department, Manchester University
Oxford Road, Manchester M13 9PL, UK
bree@cs.man.ac.uk

Abstract

What are the features of a sentence that enable a reader to distinguish between the various different temporal uses of prepositions? To answer this question, we analyzed all the sentences in the million word Brown corpus containing a temporal use of FOR. The preposition FOR has been taken as a case study as it is both highly frequent and has several different uses. The results show that many different aspects of the FOR PP itself and of its matrix play a role. The most important are: the temporal nature of the noun in the PP as this indicates the temporal use, the definite v. indefinite nature of the determiner in the PP (in particular if there is no determiner, then the distinction between absence due to a proper noun v. a plural), the nature and aspect of the verb in the matrix of the PP, the position of the PP relative to other components of the matrix and finally the presence of certain qualifiers and post-modifiers in the PP which attach durations to the time axis. It is recommended that attention be paid to these features when tagging a text for temporal analysis.

1. Introduction

We use a dozen or so prepositions to convey temporal information. A few of do so in several different ways. How do listeners determine which of these ways the speaker intended? They use other features in the sentence, such as the aspect of the verb. Determining what these features are is the central concern of this talk.

We focus on just one of these prepositions: FOR. FOR is the most prolific of the temporal prepositions after IN, which I have already dealt with elsewhere (Brée, in press). And, like IN, but unlike the other high frequency temporal preposition AT, it is used in several different ways. The main distinction is between temporal durations (lengths of time not directly attached to the time line, e.g. FOR *five minutes*) and named temporal intervals, which are attached to the time line, e.g. FOR *1961*.

The most frequent temporal use of FOR is to indicate that a state given by the matrix in which the FOR Prepositional Phrase (PP) is embedded, holds for a length of time, i.e. FOR acts as a universal quantifier over a duration:

f32:084 She helped with teaching as well as office work FOR a few years . . .

During any sub duration of the duration *a few years*, the claim is being made that *she helped with teaching*. Indeed, the possibility or otherwise of adding a durational FOR phrase to a verb phrase is a well-known test of a state versus an event verb.

The duration may be attached to the time axis to indicate an interval, with one end or the other being the time of reference (`tor`) (Reichenbach, 1947):

n09:100b Tom had been laying for Aaron McBride FOR a long time . . .
p14:189 I won't be in town FOR a couple of days . . .

One of our tasks is to discover how to determine whether the duration indicated by the FOR prepositional phrase (PP) is attached to the `tor` or not.

FOR, unlike IN, cannot in general be used with a temporal interval, c.f.:

g14:001 There were fences IN/*FOR the old days when we were children.

So how is universal quantification over a temporal interval indicated? By using IN, even though with durations it is an existential quantifier, or, occasionally and for emphasis, *throughout*.

g32:061 Little enough joy was afforded Wright *in/*for/throughout* the spring of 1925 . . .

However, in certain circumstances FOR can be used with a temporal interval:

j66:011 "Evil, man, evil", he said, and that's all he said FOR *the rest of the night.*
b01:038 DeKalb's budget FOR *1961* is a record one . . .
h29:018b I look for TV sales and production to be approximately equal at 5.7 million sets FOR *the year* . . .

There are obviously restrictions on the use of FOR with an interval rather than a duration, and another of our tasks will be to determine what these restrictions are.

As with other temporal prepositions, FOR can be used not only to indicate a duration or an interval, but it also has special and idiomatic uses which will need to be recognized as such:

k19:060 Rector asked him to move it FOR the time being; . . . [*special*]
n23:015 And FOR the hundredth time that week, he was startled at her beauty. [*idiomatic*]

We will now look in more detail at these different ways in which FOR is used temporally before searching for the features of the context that may discriminate between these different uses. Then, by examining sentences with a temporal FOR PP in a corpus, we will develop a set of heuristic rules for actually making such discriminations using these features, and others if necessary.

The corpus that will be used is the Brown University corpus of a million words taken from 499 sample texts of American English (Kučera & Frances, 1967) selected randomly from 15 genres, labelled *a . . . r*. Sentences from the corpus are indicated by the use this letter, followed by a number to indicate the text, then a semi-colon and finally the sequential number of the sentence within its text. For example, n19:100b indicates the 100^{th} sentence in the 19^{th} text within the genre *n*. The final *b* indicates that the FOR being analyzed is its second occurrence in that sentence.

There is, in the Penn Tree Bank, a tagged version of the Brown corpus which was originally used. However, for the level of analysis that is to be undertaken here, the tagging proved to be insufficient. Since this analysis was made, a newly tagged and parsed version of the Brown corpus has been made available by the Language Data Consortium (Penn Tree version 3), but only 8 of the 15 genre have been parsed. This tagging includes an identification of the temporal use of prepositions. However, a check of the temporal uses of IN against a hand analysis showed that there were about 80% more instances tagged as the temporal use of IN (821) compared to those found by hand (483) in these same genres (Brée, in press). Consequently this new version of the Brown corpus is not suitable as a basis for the level of analysis that we are about to conduct.

2. Uses

2.1. Duration

We begin with cases in which FOR specifies a duration. It is usually indicated by an indefinite determiner in the FOR PP. In its simplest form the duration is unattached in any way to the time axis. Otherwise the duration is attached to the time axis at the `tor`, to indicate an interval either beginning or ending at the `tor`. Taken together these are the Duration uses.

The simplest Duration use is to specify a pure duration during which the matrix proposition holds, either just once or on a regular basis:

a24:035 One house was without power FOR about *half an hour* ...

k14:074 He worked FOR *two hours a day* with each model sent by the rabbi.

However, the duration may also be attached to the time axis just after or before the `tor`. The matrix state then lasts for the whole of the interval with the `tor` at one end and lasting the duration specified in the FOR phrase:

p14:189 I won't be in town FOR *a couple of days*, ...[tor+]

f34:092 Cereal grains have been used FOR *centuries* to prepare fermented beverages. [tor−]

Combinations with PPs introduced by other temporal prepositions give interesting possibilities here:

After: updates the `tor`, and the duration follows the `tor` (tor+):

n13:034 ...FOR a good minute *after* they rounded the bend ...Matilda could not speak at all.

Before: updates the `tor`, and the duration precedes the `tor` (tor−):

f13:080 ...usually you would do better to rent a place FOR a year or two *before* you buy.

Since: marks the beginning of the duration ending at the `tor` (tor−):

f20:073 FOR three years, *since* the liquor territorial conference, Torrio had ...tolerated O'Banion's impudent double-crossing.

Until: marks the end of the duration beginning at the `tor` (tor+):

d15:075 They went along the pass ...FOR a short while *until* they came to a river ...

Occasionally an indefinite duration will be found with a special use of FOR (see Section 2.3.):

p20:162 We went to the Louvre FOR *a* few hours ...

Note that the *going to the Louvre* did not last a few hours; it was the subsequent stay in the Louvre that lasted that time.

2.2. Interval

FOR phrases with intervals, usually indicated by a definite determiner, can be used in the same three ways as for IN phrases: attached to the VP, attached to an NP and as an interval over which a count in the matrix is to be taken. Taken together these are the Interval uses. We will look at each in turn.

2.2.1. Attached to the VP

As we have seen already, only certain intervals can be used in a FOR PP whose matrix is a VP. From the Brown corpus we find that there are just three types of interval that may be used.

The specification of the interval includes its duration. The interval in the FOR phrase is a duration but the duration is qualified or modified so that it becomes unique and definite:

- The duration is attached to the `tor` and made definite by using a suitable qualifier such as *past/last/next*:

  g08:010 But the South is, and has been FOR *the past* century, engaged in ...

  n19:053 FOR *the last* half hour Mary Jane had criss-crossed half the length of the Gardens ...

  l07:116 FOR *the next* hour he scrambled happily up and down the ladder, ...

- The duration is attached to some other time interval using a suitable post-modifier such as *following/ending/up to* ...:

  g54:071 ...FOR *the* weeks *following* it Tom did not know whether his return to Harvard could be arranged.

  a43:036 Operating revenues were ...up FOR *the* 12 months *ending* in March.

  g46:040 The plan is admirably fulfilled FOR *the* period *up to* 1832.

The interval is part of a larger definite interval. The part may be the beginning, the end or somewhere in the middle of the interval and is given either by an adjective or a complex NP:

- The beginning of a definite interval, using *the first*:

  f16:061 FOR *the first* three weeks, the ship skirted up the east coast of Great Britain, then turned westward.

- The end of a definite interval, using the qualifiers *the rest of/remainder of/remaining*:

  r08:022 FOR *the remainder of* the movie, Chancellor Neitzbohr proceeds to lash the piano stool ...

j79:009a Moreover, by ... always using the optimal Af-stage policy FOR *the remaining* stages, ...

- Some other part of a definite interval, using, e.g. *most/much/whole (duration)/all of*:

a18:027 FOR *most of* the 25 years the operation was under feminine direction.
a18:039 Col. Clifton Lisle ... headed the Troop Committee FOR *much of* its second and third decades ...
f15:021b ... it is possible to be exempt from the normal obligation of parenthood for a long time and even FOR *the whole duration of* married life ...

The `tor` is included in the interval using a noun phrase such as *now, the present, the moment*:

m03:016 "FOR *now*, it is clear that we were in the wrong.
g15:025 FOR *the present* it is enough to note that ...
l14:068 Try to forget motive FOR *the moment*.

In summary, the interval in a FOR phrase whose matrix is a VP must be either:

- a duration made definite by a qualifier or modifier linking it to the `tor` or another interval; or
- part of a larger interval; or
- an interval that includes the `tor` specifically, e.g. *now, the present, the moment*.

Note, in particular, that proper nouns indicating clock times and calendar dates, such as *7 p.m., May, 1996*, are not found.

2.2.2. FOR modifying a noun phrase

The matrix of FOR phrase may itself be a noun phrase referring to a regularly recurring event or object. The FOR phrase indicates a unique interval which serves to pick out a specific occurrence of this event or object. Here there is no restriction on the type of interval specified in the FOR PP:

- Regularly recurring events, e.g. plays

k05:140 The *play* FOR Saturday night was to be a benefit performance of The Octoroon.

- Regularly recurring financial objects, e.g. tax returns, budgets, costs, revenues etc.:

h24:009 Since ... your *return* FOR the calendar year 1961 will be timely filed.
b01:038 DeKalb's *budget* FOR 1961 is a record one ...
a28:013 FOR the year to date, *sales* ... still lag about 5 % behind 1960.

- Other regularly recurring objects, e.g. newspapers, journals, appointments, horoscopes, rates and even obligations:

g38:040 The final issue of the *Englishman*, No. 57 FOR February 15, ran to some length ...
j56:055 Eber L. Taylor of Manchester Depot recorded ... in his *diary* FOR 1906.
g49:003 ... having made a formal *engagement* by letter FOR the next week, ...
p21:082 I may settle on some makeshift *arrangements* FOR the summer.
r09:078 ... I'd say that your *horoscope* FOR this autumn is the reverse of rosy.
j07:030b ... to give reliable impact *rates* FOR the periods of exposure.
b17:009a Our only *obligation* FOR this day is to vote, ...

It could be argued that the use of a FOR phrase to qualify a noun is not essentially temporal. In many of the above examples the time in the FOR phrase could be replaced by an appropriate proper noun, e.g. FOR *IBM*, FOR *John*. The FOR phrase would then indicate possession or purpose, e.g. *the budget of IBM, the arrangements for John* (i.e. the special use of FOR, see Section 2.3.). We have included it here as we have chosen to include as temporal all uses of FOR that have a temporal noun in the PP.

2.2.3. There is a count over an interval given by the FOR PP

This is closely related to the above, but there is no NP which the FOR PP is modifying. Rather there is a count over an interval given by the FOR PP:

a43:038 FOR the year, the road earned 133 per cent of its interest costs ...

Normally IN is used for indicating the interval over which a count is taken, but when the interval is given by a duration made definite by the determiner *the*, then FOR is used.

The remaining examples of the use of a definite noun phrase with FOR are non-standard. These have been categorized as a special use (Section 2.3.) and an idiomatic use (Section 2.4.).

2.3. Special use

The special use is one in which the FOR phrase gives an interval but not one in which the matrix proposition holds. Rather it is an interval, later than the `tor`, in which some event, that is related to the matrix proposition in some way, will take place. The relationship may be a simple prediction, planning or purpose:

h29:008 I believe a further gain *is in prospect* FOR 1961. [*prediction*]
a10:045 The dinner is sponsored by organized labor and *is scheduled* FOR 7 p.m.. [*plan*]
l24:118 We *decided to* leave the third one intact FOR tomorrow. [*purpose*]

Note how in all these examples, the matrix proposition holds at the `tor` rather than in the interval given by the FOR PP. In:

h29:008 the *gain* is already *in prospect* at the `tor`; the actual gain will be realised later, in *1961*.

a10:045 the *dinner* is already *scheduled* by the `tor`; the actual dinner will take place later, at *7 p.m.*.

l24:118 the *third one* is *left intact* at the `tor`; the reason for this is so that it can serve some purpose *tomorrow*.

It is the matrix that is indicative of this special usage. It needs to indicate that an action has been undertaken that will lead to some event to take place or state to be the case at a future interval that is given by the FOR PP. How this is indicated is not simple. The matrix verb, plus particle if present, gives the indication: *be in prospect, be scheduled*,

decide to X. A list of verbs that were found in the corpus to indicate special use will be given in Section 4.2.

While the FOR PPs in the purposeful sentences such as l24:118 clearly have a temporal function, they are not only temporal. There is also a sense of purpose which is another use of FOR:

a01:060 Vandiver opened his race FOR governor in 1958 . . .

Should they be included as a temporal use of FOR? With prepositions with both temporal and spatial uses, such as AT or IN, the temporal use can be detected from the temporal nature of the noun in the PP (Brée & Pratt, 1997). If we wish to use this general heuristic to identify when a PP is being used temporally rather than spatially, we need to keep these as a special temporal use of FOR.

Note also that, as with NP attachment of the FOR interval, proper clock and calendar nouns can be used in the FOR PP when there is a special use.

2.4. Idiomatic

For the Nth time, where *Nth* is any ordinal, generally *first*, but including *last*, was classified as idiomatic use:

p22:041 It was weeks before we even kissed FOR *the first time*.

19:008 And FOR *the thousandth time*, I answered myself.

The reason for considering this an idiomatic use is that the noun *time* is here not being used in its temporal meaning but as an indication of position in a sequence. Confirmation of this difference is given by the different translations given in other languages. In Dutch, for instance, the normal word for time is *tijd*; but in the expression *for the Nth time*, it is *keer*.

This concludes the list of uses for temporal FOR phrases. We turn now to an examination of the components of the phrase in the expectation that they will distinguish between these different uses.

3. Components of the prepositional phrase

Our task now is to code features of the sentences which may indicate which of the temporal uses of the FOR PP was intended by the writer. The following information was coded for each sentence in the Brown corpus in which the preposition FOR was being used temporally. The features of the PP, being those specified in a standard grammar (Quirk et al., 1985), arranged in order of occurrence in the PP as in *Except for most of the first glorious year of his Presidency*, were:

- Any pre-modifier of FOR (16):[1] *not, except, only, save*.
- Any pre-determiner of the NP in the FOR PP (68): a fraction (*of*), a cardinal number *of, all, only, even, most of, much of, longer than, more than, about, approximately, nearly, almost, up to, at (the) least, over*.
- Any determiner of the NP: definite, indefinite, demonstrative, possessive, quantifier, none or zero. When there was no determiner a choice was made, by hand, whether to give a 'none' or a 'zero' entry. A 'none' entry was chosen if the noun was a noun, usually proper, which could not take any determiner, i.e. a particular year, month or date. Otherwise a 'zero' entry was chosen. The reason for this distinction is that the 'none' entries behave in the same way as definite determiners, whereas the 'zero' entries behave in the same way as indefinite determiners.[2]

  The distinction between 'definite' and 'indefinite' was important in distinguishing between Duration and Interval uses of FOR. The 'demonstrative' and 'possessive' entries all had the same effect as the 'definite' and 'none' determiners. We will refer to them as Definite determiners (302). The 'quantifiers' were the existentials *some* and *any*; they occurred in FOR phrases which were given a duration meaning. Therefore, they were grouped with 'indefinite' and 'zero'; this group will be referred to as Indefinite determiners (704).
- Any post-determiner of the NP (541): (*total of*) *N* [a cardinal number], *Ns of*, an ordinal number, *single, couple of, dozen, number of, few, several, (so) many, matter of, another, more, past, preceding, last, next, only*.
- Any qualifier of the head noun in the NP, e.g. *glorious*.
- The head noun in the NP, sub divided by type (Brée & Pratt, 1997):

  Measure of a temporal duration (865), e.g. *year*;

  Calendar interval (47), e.g. *1966, future*;

  Cyclical (22), e.g. *morning, spring*;

  Part of a longer event (10), e.g. *act, inning, movement, session, spell, stage*;

  Adverbials (25): *ever, long, now, once, tomorrow, tonight*;

  Life (2): *birthday, lifetime*;

  Complex, always with a post-modifier (2): *length, remainder*
- If there was a phrase post-modifying the NP, then the preposition or conjunction introducing this post-modifying expression (126): *of, when, in (advance), during*, any demonstrative, *ended, ending, up to, after(ward), since, following, commencing, starting, or, and, at, each, every, a(n), now, being, to (come), longer, forward, as (a whole), than.*

Certain properties of the verb in the matrix were also noted as these were required to make the distinction between different uses:

Tense (only coded when required): past, present, past/present participle, infinitive;

Aspect (174): perfect (146), progressive (2), perfect progressive (26);

[1] The numbers in parentheses give the number of occurrences in the total of all 1006 examples of temporal use of FOR in the Brown corpus.

[2] Two 'nouns', actually adverbials being used as nouns, never take a determiner, but the determiner was coded as 'Zero' rather than 'None' as they always occurred with Durations rather than Intervals: *long* (17), *ever* (1).

Modal (92): *will* (46), *shall* (5), *can* (30), *may* (6), *have to* (3).

Also with Interval uses, the position of the FOR PP relative to other components of the matrix, in particular its verb.

The categories that turned out to be of most help in discriminating between different uses of FOR were the determiner, the noun itself and the post-modifiers, which we now need to examine in some detail.

3.1. Post-modifiers

As we have seen, the noun in the FOR phrase is frequently post-modified (126). There are of several kinds of post-modifiers: adverbials, conjunctions, demonstratives and other temporal prepositions followed by a noun phrase. Each can be used for different functions depending on whether the determiner is Definite or Indefinite. We now will look in detail at which post-modifiers indicate which function. This section can be skipped without loss of continuity; a summary is given in Section 3.1.3..

3.1.1. With an Indefinite determiner

When the determiner was Indefinite (82), usually associated with one of the Duration uses (see Section 4.3.), the post-modifying phrase has one of the following functions:

- It can simply add information about the duration (22), usually just to say that a *period* is temporal (*period of time*, 7), but also to indicate more precisely its length (using *period of*, 4; *as long as*, 4) or even its nature (*of*, 5):

  j11:081 ... many have been made FOR a very short *period of time*.
  j08:054a All samplers were operated FOR a *period of* two hours ...
  j16:012b ... until used they were stored FOR *as long as* 2 weeks.
  f11:076 An average national figure FOR two to three years *of* treatment would be $ 650 to $ 1,000 .

- It can signal that the duration is generic (18), either directly (*at a time*, 2) or by giving the cyclical frequency (*a(n)*, 13; *every*, 3; *each*, 2):

  k14:074 He worked FOR two hours *a* day ...
  g37:010 ... the missionaries skipped FOR hours *at a time*.
  f04:117 Rugged outdoor exercise FOR an hour and a half *every* day

- It can link the duration to the `tor` (14), either as preceding it (*now*, 10) or following it (*to come*, 6; *longer*, 3; *forward*, 1)

  b08:066 She's been in and out of my house FOR a dozen years *now* ...
  b15:005b The demand for these lots can be met FOR some time *to come*.
  p07:005 Spencer was quiet FOR a moment *longer* ...
  h02:081a ... they must plan their own complex investment programs FOR at least 5 years *forward* ...

- It can link the duration to an interval other than the `tor` (20), either being in the interval (*in*, 6; *during*, 2; a demonstrative, 2) or beginning at the interval (*after(ward)*, 3; *since*, 1; *following*, 3; *starting*, 1; *commencing*, 1):

  a28:031 Except FOR a few months *in* late 1960 and early 1961 ...
  b23:087a A truth-revealing crisis erupted in Katanga FOR a couple of days *this* month ...
  k29:016 FOR many nights *afterward* ...
  j56:088a FOR a time *following* the abandonment of the local plant ...

- It can even alter the extent of the duration given by the head noun in the FOR PP (11), either by giving an alternative duration (using *or*, 5), by including another type of duration (*and*, 1) or by setting the given duration as a maximum (*as long as*, 4) or a minimum (*more ... than*, 1):

  f43:021 FOR a moment *or* two, both scenes are present simultaneously ,
  r06:063 ... her husband hasn't been home FOR two days *and* nights."
  j16:012b ... until used they were stored FOR *as long as* 2 weeks.

3.1.2. With a Definite determiner

When the determiner was Definite (44) the post-modifying phrase has one of the following functions:

- To specify the interval over which the count FOR *the* N^{th} *time* is to be taken (21), using *in* (14), a demonstrative (4) or *since* (3):

  l12:154 FOR the first time *in* his life ...
  n23:015 And FOR the hundredth time *that* week, he was startled at her beauty.
  r01:035 Then, FOR the first time *since* his arrest ...

- To specify the larger interval from which a part is selected by the head noun, using *of* (7):

  p03:078 FOR the first few months *of* their marriage ...
  r08:022 FOR the remainder *of* the movie ...

- When the head noun is a Measure noun, to specify the end of the duration thus selecting an interval and making the FOR NP Definite (5), using *ending/ed* (3), *up to* (1), *following* (1):

  h27:001 Sales and net income FOR the year *ended* December 31, 1960 ...
  h29:011 FOR the year as a whole, retail sales of TV sets ...

- To describe a further attribute of the interval which enables it to be identified uniquely (6), using *of* possessively (5), or *when* with a clause (1):

  a05:025b ... come up with recommendations for possible changes in time FOR the next session *of* the General Assembly.
  h15:081 A flashlight or electric lantern also should be available FOR those periods *when* a brighter light is needed.

- Other (5): relating an imprecise interval to the `tor` (*to date*, 1; *time being*, 1), giving a range (*to*, 1) or giving emphasis (*as a whole*, 2):

  a28:013 FOR the year *to date* ...

h13:008 ...total annual rail commutation dropped 124 million FOR 1947 *to* 1957.
h29:011 FOR the year *as a whole*, retail sales of

3.1.3. In summary

Most of the post-modifiers can be used with both a Definite and an Indefinite determiner:

- *of*:
  Indefinite: property of the duration,
  Definite: possessive, after a temporal part noun, to indicate which larger interval it is a part of;
- *in, during*: and the demonstratives *this, that*:
  Indefinite: linking the duration to an interval,
  Definite: to give the interval over which the count in *the N^{th} time* is to be taken;
- *ending, up to, after, since, following, commencing, starting*: after a Measure noun, indicating that the duration ends or begins at the interval in the post-modifier PP (the FOR PP may be either Definite or Indefinite);
- *or, and*: coordination of either a duration or an interval.

Certain post-modifiers only appear after an Indefinite determiner:

- *at a time, each, every, a(n)* indicate that the duration is generic (and give its frequency);
- *now, to come, longer, forward* attach a duration to the `tor`;
- *as long as, more ... than* affect the extent of a duration.

This concludes the description of the various features in a temporal FOR PP, and its matrix that are likely to indicate which of the different possible uses of temporal FOR was intended by a writer. Most attention has been given to the post-modifying phrase as this is the least well understood part of the FOR PP. While the post-modifier is important, it is by no means the only feature of significance. We will see that the determiner, the qualifier and the verbal aspect also influence the use.

4. Distinguishing uses

We turn now to the task of distinguishing between the different uses of temporal FOR phrases, using the features introduced in the previous section.

There are 9482 instances of FOR in 7710 sentences (out of a total of 52355 sentences) in the Brown corpus. If the noun following the FOR was one that is known to indicate temporal use (Brée & Pratt, 1997), then it was included in the sample. Correspondingly, if the noun clearly indicated another use, e.g. purpose, then it was discarded. For all the remaining instances, the complete sentence was inspected to determine whether or not there was a temporal use. All the temporal instances were classified by use. Finally, the features, as given in Section 3., were coded. The numbers of instances of each temporal use of FOR in the Brown corpus are shown in Table 4.

We begin with the two most distinctive uses, idiomatic and special. Then we will see how to distinguish Durations from Intervals. Finally we will look at the different types of Duration and Interval use.

Use	Instances	Sub-totals
pure duration	212	
`tor−`	168	
`tor+`	265	
All Durations		677
interval: Count	8	
interval: Identify	94	
interval: Universal	73	
All Intervals		175
Special		97
Idiomatic		57
Total		1006

Table 1: Temporal uses of FOR PPs in the Brown corpus

4.1. The idiomatic use

The idiomatic use of FOR (57) in which the noun *time* indicates a position in a sequence rather than the passage of time is readily detected. All occurrences of the singular noun *time* with a post-determiner were idiomatic use. The post-determiner was usually an ordinal number (54) but could also be *last* (2) or *only* (1):

j64:033 FOR *the only time* in the opera, words are not set according to their natural inflection ...

4.2. The special use

Detecting the special use of FOR (97) to give the interval in which a consequence of the matrix will take place, is not simple. Simple features turn out not to be useful. For example, usually (78/97) the determiner is Definite, but Definite determiners are more frequently used (167) for simple intervals.

It is the matrix verb that indicates the special use of FOR. These verbs all have some element of purpose, planning or prediction. Here are examples of such uses grouped by these features. Note that there is no hard and fast way to separate these verbs into these three categories.

planning i.e. carrying out an activity at the `tor` that sets up a later event or state in the interval given by the FOR PP (46), e.g. *be available, have, issue* something, *invite* somebody, *plan, schedule, set*:

a18:081 Their Majesties ... have jointly *issued* invitations FOR Shrove Tuesday evening at midnight.
a42:006 ...Mr. Kennedy *invited* Stevenson to Cape Cod FOR the weekend.
a43:051a Higher tolls are *planned* FOR July 1, 1961, ...
l12:101 Andy's performance was *scheduled* FOR eleven o'clock.
a18:048c A preview party for sponsors of the event and for the artists is *set* FOR April 8.

predicting from what is known at the `tor` that some event or state will occur at a later time given by the FOR PP (8): *auger well/badly, be in (prospect), be out, continue, leave, realize, suggest* something (once each):

b20:076 He has, moreover, another qualification which *augurs* well FOR the future.
b10:048 To me, Brandt looks as though he could *be in* FOR a fine year.
h29:008 I believe a further gain *is in prospect* FOR 1961.
a23:004 ...Multnomah, as of Aug. 22, had spent $ 58,918 out of its budgeted $ 66,000 in the category, *leaving* only $ 7,082 FOR the rest of the month.
j39:091b ...and to appraise the forecast that its interpretation *suggests* FOR the future of farm prices over the years immediately ahead.

purpose in which some activity is conducted at the `tor` in order to achieve some state in a time period given in the FOR PP (33), e.g. *be at, be out, bed down, develop* something, *go (back/to), live, prescribe, protect, shut up, stash (away)*:

p11:030 I made a lemon sponge, ..., so there *would be* something nice in the icebox FOR the weekend.
k12:002 Once, they *were at* Easthampton FOR the summer.
l02:107 ..., but when you ...know that you can't eat until he's *bedded down* FOR the night, ...
e28:104b Does your company have a program for selecting and *developing* sales and marketing management personnel FOR the longer term?
p20:162 We *went* to the Louvre FOR a few hours.
l11:082-4 "You *live* in the present?" "In the present', Felix proclaimed. "FOR the future. ..."
f34:027 The artist who paints in oil uses drying oils to carry the pigments and to *protect* his finished work FOR the ages.
k26:027 The women ...sounded like chickens *shut up* in a coop FOR the night.
l24:144 I've got a little *stashed* FOR a rainy day, and I guess this is rainy enough.

Clearly this list of verbs is incomplete. Nor is it obvious what test should be applied to verbs to decide whether or not they can indicate a special use of a temporal FOR phrase. They should admit of prediction, purpose or planning, but how to develop this insight into a test is not clear.

4.3. Duration versus Interval

Having distinguished idiomatic and special uses of temporal FOR, we turn to Duration and Interval uses (852). Both uses can be one-off or generic. Since distinguishing between one-off and generic sentences is a general problem whose solution does not depend on the FOR PP, we will not attempt to make the distinction here. Instead we now examine how to make the main distinction, that between Duration uses on the one hand and Interval uses on the other. In general this is easy: an Indefinite determiner indicates a Duration use (8/685), a Definite determiner indicates an Interval use (4/167). However, there are exceptions.

Very occasionally (8/685) an Indefinite determiner does not indicate a Duration use. This is when:

- There is the universal quantifier *all* as a pre-determiner (4):

j19:044 The managers stay the same, so that A[fj] is the same FOR *all* weeks.
f30:044 Probably a lawyer once said it best FOR *all* time in the Supreme Court of the United States.

These four sentences are the only ones in which *all* occurs as a pre-determiner. Hence it is a reliable if infrequent indicator of an interval use.

- A duration is made a particular interval by being directly attached to the `tor`, using the post-determiners *last, next* (2):

b06:083 South Viet Nam's rice surplus FOR *next* year ...may have been destroyed.
a27:024 A substantial rise in new orders and sales of durable goods was reported FOR *last* month.

This is similar to, but different from, the attachment of a duration to the `tor`, indicated by a perfect aspect or modal (see Sections 4.4. and 4.5.). The difference arises as the noun phrase itself is sufficient to detect that there is attachment to the `tor`. In fact, while there is no determiner in these examples, a definite determiner is acceptable: FOR *the next/last month*.

- A duration is made a particular interval by being directly attached to a point of time, using certain post-modifying expressions, such as *commencing, following* (4):

h07:095 ...they could levy taxes FOR an interim period of nine months, *commencing* with September 30 and ending with June 30.
a03:054 Full payment of nursing home bills FOR up to 180 days *following* discharge from a hospital.
d08:007 ...which succeeded in reuniting China and keeping it together FOR a longer period (*from* 202 B.C. *to* A.D. 220).
a23:002 FOR a second month *in a row*, Multnomah County may be short of general assistance money ...

This is somewhat different from the previous two cases, as here the determiner can't be definite. Moreover, several other gerundive post-modifiers would have this same effect, e.g. *beginning, starting, ending*.

- There is no determiner, but a definite determiner could have been used (2):

a26:086b ...estimated sales of domestic cars in the U.S. FOR first three months of 1961 were ...
j37:022a ...the rumors of election dates appeared once again, first FOR spring of 1958 and later for the summer.

In a26:086b it could be argued that a definite determiner should have been included. But j37:022a is quite acceptable as it stands. In fact, a definite determiner may be dropped with a season, and the season then acts as a proper noun.

Only 4 of the FOR PPs with a Definite determiner turned out to have a Duration rather than an Interval use, all but one with a superlative in the FOR PP.[3] Superlatives can be definite but still indicate a Duration use, here always `tor+`:

[3]The one exception was:

p25:068 Richard's dark eyes came up and seemed FOR *the tiniest moment* to reflect sharp light.

All the superlative qualifiers indicated tor+ use.

In summary, our simple rule for distinguishing between Durations and Intervals needs to be elaborated to:

- If the determiner is Indefinite, then there is a Duration use, unless:
  - there is the pre-determiner *all*;
  - *last* or *next* are qualifiers, indicating that the duration is attached to the tor;
  - there is a post-modifier that attaches the duration to some other time point, e.g.: *beginning, commencing, following, starting, ending*;
  - there is no determiner and the noun is a season;

  when there is an Interval use.
- If the determiner is Definite, then there is an Interval use, unless there is a superlative qualifier when there is a Duration use.

We now turn to distinguishing between the different Interval and Duration uses.

4.4. Distinguishing tor− from other Duration uses

The duration can be attached to the time axis to indicate an interval ending at the tor (tor−) (168). Can this tor− use be distinguished from other Durations? A likely indicator is the perfect aspect as it is used to place an event or state before the tor. While perfect aspect was an indicator of tor− use with IN PPs other features of the matrix, sometimes in conjunction with the perfect aspect, were better indicators: a negative, superlative or ordinal in the matrix (and sometimes if there was a cardinal number and perfect aspect), or if the IN PP was topicalized (Brée, in press). Turning to the data for FOR PPs we find, by contrast:

- The perfect aspect frequently (150/165) did indicate a tor− use:

  f31:080 They *have* not been friendly FOR years".
  n09:100b Tom *had* been laying for Aaron McBride FOR a long time ...

  However, 15 FOR PPs occurring in a matrix with a perfect aspect weren't tor− use. One reason is that a perfect aspect with a past tense is used to indicate a time earlier than the tor rather than the time just prior to the tor (10/15):[4]

  k28:165 After that they *had* sat FOR five minutes without saying a word.

  If the past perfect aspect is being used consistently, e.g. as it is in the text preceding k28:165:

  k28:161 "I'll give you ...", Miss Ada *had* said.

  then it is likely that the past perfect in the matrix is also setting the time of the event to before a past tor, rather than to the time immediately preceding the tor. The past perfect aspect occurs much more frequently with the tor− use than with the other Duration uses (65/10), so does not in itself provide an exclusion heuristic.

- The FOR PP was embedded in an *after* phrase (9), generally with a present participle (7):

  a17:001 *After being* closed FOR seven months, the Garden of the Gods Club will have its gala summer opening Saturday, June 3.
  f16:118 Once, *after* the Discovery lay FOR a week in rough weather ...

  Inserting a perfect aspect in these after phrases is quite possible, but clearly not necessary. All occurrences of an Indefinite FOR PP in a *after* phrase were tor− use.

- If the verb in the matrix was a past participle then there was usually (5/7) a tor− meaning:

  a12:074 Halfback Bud Priddy, *slowed* FOR almost a month by a slowly-mending sprained ankle, ...

  It is not possible to have a perfect aspect with a past participle:

  a12:074′ Halfback Bud Priddy, *had* slowed FOR almost a month by a slowly-mending sprained ankle, ...

  The 2 instances of a past participle in the matrix which were not tor− meaning were both generic durations. Distinguishing generic from episodic sentences is non trivial!

These rules explain 164 of the 168 instances of tor− use, but without any means for detecting that the past perfect had been established as the tense of story, they would also include 17 of the other 509 Duration uses (see Table 4.4.). So there is no hard and fast rule for distinguishing tor− use from other Duration uses, but the following heuristics would be most useful. The duration of FOR PP is likely to end at the tor if:

- there is a perfect aspect in the matrix (unless the perfect past has been established as the tense of use); or
- the matrix is an *after* phrase (generally one with a present participle rather than a full tense); or
- the tense verb in the matrix is a past participle.

Rule	When use is: tor−	all	Cumulative: OK	miss
Matrix is an *after* phrase	9	9	9	0
Perfect aspect in matrix	150	165	159	15
Matrix verb is past participle	5	7	164	17
Out of a maximum of:			168	509

Table 2: Evaluation of the heuristic rules for tor− use

j04:095 The temperature was maintained to within about A[fj] FOR *the* period of time required to make the measurement (usually about one hour).

[4] In all the remaining 5 examples it is a present tense perfect aspect that fails to indicate a tor− use. It seems unlikely that these examples could be excluded without resorting to domain knowledge.

4.5. Distinguishing tor+ from other Duration uses

Attachment of the duration to after the tor, the tor+ use, is also possible and indeed very frequent, occurring in almost half the Duration cases (297). With temporal IN PPs, the rules for distinguishing tor+ use from pure durations were heuristic, so the same is to be expected with FOR PPs. The indicators of the tor+ use with FOR PPs that were found are:

- Topicalization of the temporal PP, without a perfect aspect, always signalled tor+ use with IN (Hitzeman, 1997; Brée, in press). The tor+ use of IN is unusual in that the time of the matrix event is set to be at a time interval which is at the end of a duration, given by the IN PP, after the tor, rather than within an interval starting at the tor and lasting this duration, as one might expect. But even so topicalization without a perfect aspect also usually indicated tor+ use with FOR PPs (62/80):[5]

  l10:172 FOR exactly one week, she was able to continue in this manner.

  but not always (18):

  a19:034 FOR a number of years the board used a machine to keep a permanent record but abandoned the practice about two years ago.

- *Then* is a marker for the tor. When *then* is in the matrix (15) or begins the clause that follows the matrix (which may be the next sentence) (53) then tor+ is almost always (68/73) indicated:

  g67:080 *Then* he kept Blackman awake FOR more than an hour ...
  f31:098a She worked as a domestic, first in Newport FOR a year, and *then* in ...

- The vague noun *time*, without a qualifier or post-modifier, was an indicator of tor+ use with IN, as in the expression *in time*. With FOR, several nouns were found more frequently with a tor+ use than with a pure duration (177/203), e.g.: *instant* (10/11), *moment* (56/63), *second* (14/16), *minute* (50/63), *(a)while* (28/29), and *time* when there was no qualifier (19/21):

  n29:174 The kid showed FOR an *instant*, and his arm was cocked back.
  n07:110 Barton waited FOR a long *moment* ...
  n12:049 He studied the problem FOR a few *seconds* and thought of a means by which it might be solved.
  p20:079 She was silent FOR a *while* ...
  l14:048 Detective Pearson, Eighteenth Precinct, thought FOR *a time* he might be on to something.

  Why these nouns? *While*, like *time*, refers to only a vague duration, so is not useful for specifying the duration of an event unless the duration is attached to the tor. The other nouns (*instant, second, moment, minute*) refer to short or very short durations, almost too short for any ordinary event to take place within; so they are not often used for that purpose. When they are, it is usually in a generic context, where there is no explicit tor:

  k25:040 He looked at her out of himself, she thought, as he did only FOR *an instant at a time* ...
  k25:048a *Every* few minutes she would awaken FOR a *moment* to review things ...

  or there is a pre-modifier:

  g35:004 *Not* FOR a *moment* do we forget that our own fate is firmly fastened to that of these countries ...

- The post-determiner *another* (6) and the qualifier *additional* (1), always indicated tor+:

  f28:005b ...it would be best for all if the plantation were operated FOR *another* year.
  j27:054b ...the group felt a topic under study should not be dropped FOR an *additional* week ...

- Some post-modifiers, such as *to come* (6), *longer* (3), *afterward* (1), also always indicate tor+:

  h03:027 May the Divine Speaker in Heaven bless this country with Sam Rayburn's continued service here FOR years *to come*.
  l12:102 He stalled FOR a half-hour *longer* ...
  k29:016 FOR many nights *afterward*, the idea of ...would return

Rule	When use is: tor+	all	Cumulative: OK	miss
another, additional	7	7	7	0
to come, longer, afterward	10	10	17	0
Noun: *time* (not qualified)	19	21	35	2
Nouns: *instant, second*	52	56	85	5
then in matrix or following	68	73	135	10
Noun: *moment*	56	63	176	15
Noun: *minute*	50	63	210	25
FOR PP topicalized	62	80	219	42
Out of a maximum of:			297	212

Table 3: Evaluation of the heuristic rules for tor+ use

The affect of applying these rules together, rather than individually, is shown in Table 4.5.. The rules are applied in order of most effective first, starting with the post-determiners, qualifiers and post-modifiers that always indicate a tor+ use, then adding in those nouns that are most indicative, then the presence of *then*, then the remaining nouns and finally topicalization. The additional effect of each rule can be seen in the two rightmost columns. Note in particular that, although topicalization was a good indicator of tor+ use with IN PPs and that it frequently occurs with this use of FOR PPs, it detracts from the effectiveness of the rule set, adding only 9 tor+ uses but 17 pure duration uses; so topicalization, while a good heuristic in itself, is not a good heuristic in conjunction with the other rules in the set.

[5] The numbers in parentheses indicate the number of tor+ occurrences out of the total of tor+ plus pure duration occurrences. Occurrences with tor− are not included as tor− usage is almost always indicated by a perfect aspect or a matrix which is an NP.

To explain the remaining instances of tor+ use, further heuristics are necessary, for example the nature of the verb in the matrix. One of the heuristics useful for detecting tor+ use with IN PPs was the presence of one of the modals *will, shall* or *may*. However, this heuristic is not specific enough (25/ 51) to be useful with FOR.

We have selected the following heuristics (in addition to there being no perfect aspect, the matrix not being an *after* phrase and the adverbial *now* not being in the matrix, all of which indicate tor− use) as indicating tor+ rather than other Durative uses:

- there is a post-determiner or qualifier in the class of words such as *another*, or
- there is a post-modifier in the category of words such as *to come*.
- the noun in the FOR phrase is very short (*instant, second*) or vague (*while*, or
- the noun in the FOR phrase is *time*, unqualified, or
- *then* appears in the matrix or begins the following clause (which may be the next sentence), or
- the noun in the FOR phrase is shortish: *moment, minute*.

This concludes the rules for distinguishing between different Duration uses. The curious reader can check them for himself by taking any simple sentence in which there is a temporal FOR PP with a pure duration use and altering it in each of the above ways. They will find that the pure duration use switches to a tor+ use. The skeptical might like to put this hypothesis to a psychological test.

4.6. Distinguishing between different interval uses

We now turn to the distinguishing between the three different Interval uses. The main distinction to be made is whether the FOR PP is attached to a VP (73) or an NP (94). There are only a few (8) count uses.

Detecting NP attachment is a classic problem. In this context the noun to which a FOR PP is attached is usually a regularly recurring event or object, such as financial figures (see Section 2.2.2.). However, there are many such types of noun and it is not clear how to use this information to detect that a FOR PP is attached to an NP. It can sometimes (63/94) be detected by the following syntactic heuristics:

- Always whenever the FOR PP occurs after a noun but before the verb (41):

  h24:024 His return FOR the period January 1 to June 20, 1961, is due April 16, 1962.

- The FOR PP occurs not only after the verb but also after another PP (18):[6]

  b27:081 Sir Robert Watson-Watt wrote, *on* page 50 of SR Research FOR 4 March 1961: ...

- Whenever the FOR PP follows one of a sequence of nouns after the verb (4):

  h30:035 An alert dean will confer all through the year on personnel needs, plans FOR the future, qualifications of those on the job , and ...

There are various other clues, but there will remain about a third of the sentences, in which the FOR PP follows a single NP after the verb but which should be classified as NP attachment:

e28:098 Have you estimated your sales manpower needs FOR the future ...

Of course, if the verb in the matrix is an event, as in e28:098, then the FOR PP can't be given VP attachment. This would require classifying the verbs by Vendler's types (Vendler, 1957), a difficult but not impossible task. However, recall that attachment to the VP can only occur under rather restricted circumstances (Section 2.2.1.):

- The specification of the interval, signalled by a Definite determiner, includes its duration, attaching the duration to the tor using the post-determiners *past, last, next* (25/35):

  b16:038 My husband's hours away from home FOR the *past* years have been ...

- The FOR NP picks out a particular duration from part of a longer interval (24/31),[7] using:
  - the N^{th} interval of that duration, usually the *first* (11/16);
  - the end, using *rest of, remainder of* (6/ 7);
  - (almost) all, using *all, most, much, whole of* (8/ 9)

- The temporal noun is one that specifically includes the tor (13/13): *moment, present, now*:

  b02:067 ...this approach might be expected to head off Mr. Khrushchev FOR the *moment*.

These three heuristics pick out 70 of the 81 VP attachments, but also include 16 cases of NP attachment. This is more restrictive than the conditions for attachment to an NP, so it would be simpler to look for VP attachment rather than NP attachment. Fortunately, the heuristics for detecting NP attachment detect all but 6 of these 16 cases. So once it has been signalled that VP attachment is likely, a check should be made that after all it isn't NP attachment.

Finally, the occasional use (8) to indicate an interval over which a count is taken is easily spotted by there being a cardinal in the matrix giving the count. Clearly there can also be a cardinal in the matrix with VP attachment, but this happened not to be present in this FOR extract from the brown corpus.

An overview of the effect of each rule, starting with the rule for the count use, is shown in Table 4.6.. Note how adding the simple rule that a FOR PP before the verb must be NP attachment reduces considerably the misclassification as VP attachment. Adding the two additional rules for identifying miss classifications of NP attachment as VP

[6] There are 2 exceptions which are VP attachment:

n26:046 Blue Throat, who had ruled the town *with* his six-shooter FOR the last six months ...

a43:036 Operating revenues were off in the first three months of 1961, but up FOR the 12 months ending in March.

[7] The number after the forward slash gives the number of occurrences in all the interval readings taken together, here 31.

Use	Rule	When use is: this use	all	Cumulative: OK	miss
Count	Cardinal in matrix	8	8	8	0
VP attachment	Noun: *moment, present, now*	13	13	21	0
	Qualifier: *rest/remainder of*	6	7	27	0
	Pre-determiner: *all, much/most/whole of*	8	9	35	1
	Post-determiner: Nth	11	16	45	6
	Post-determiner: *past, last, next*:	25	35	70	16
Count and VP attachment		In total:		81	94
NP attachment	FOR PP before verb	41	41	111	9
	Sequence of nouns	4	4	115	8
	FOR PP after another PP	18	20	133	6
	NP attachment as default			164	6
All Interval uses		In total:		175	175

Table 4: Evaluation of the heuristic rules for Interval Uses

attachment helps only marginally. So the only rule that I propose for identifying NP attachment is the obvious one that the FOR PP is before the matrix verb. Then anything already identified as some Interval use but that isn't count use or VP attachment is, by default, NP attachment.

4.7. Summary

Bringing this all together we see that some uses of temporal FOR phrase can be readily distinguished:

- The expression *for the Nth time* indicates the idiomatic use;
- The special use is identified by the presence of predictive, planning or purposeful verbs in the matrix.
- *A Definite determiner indicates an Interval use:
  - provided that the adverbial 'nouns', such as *long, ever* are not classified as proper nouns, even though they can never take a determiner;
  - *unless there is a superlative qualifier.
- *An Indefinite determiner indicates one of the Duration uses, unless:
  - there is the pre-determiner *all*; or
  - *there is a qualifier that attaches the duration to the `tor`, e.g.: *last, next*; or
  - *there is a gerundive post-modifier that attaches the duration to some time point other than the `tor`, e.g.: *beginning, commencing, following, starting, ending*; or
  - there is no determiner and the noun is a season;

  in which cases there is an Interval use.

Other uses are difficult to distinguish, but the following heuristics would be of help. For distinguishing between the different Duration uses:

- Attachment of a duration to before the `tor` rather than pure duration is indicated by:
  - *a perfect aspect in the matrix, *unless the perfect aspect has been established as the tense of use; or
  - the matrix is an *after* phrase (generally with its verb in present participle form); or
  - the verb tense in the matrix is a past participle.
- Attachment of a duration to after the `tor` rather than pure duration is indicated by the following heuristics:
  - there is a post-determiner or qualifier in the class of words such as *another*; or
  - there is a post-modifier in the category of words such as *to come*; or
  - the noun in the FOR phrase is very short (*instant, second*) or vague (*while*); or
  - the noun in the FOR phrase is *time*, unqualified, or
  - *then* appears in the matrix or begins the following clause (which may be the next sentence); or
  - the noun in the FOR phrase is shortish: *moment, minute*.

To distinguish between the different Interval uses:

- Use of the FOR PP to specify an interval over which a count is to be made is indicated by there being a cardinal in the matrix.
- Attachment of an interval to the VP rather than an NP is indicated by:
  - the specification of the interval includes its duration which is attached to the `tor` using the post-determiners *past, last, next*; or
  - the FOR NP picks out a particular duration from part of a longer interval; or
  - the temporal noun is one that specifically includes the `tor`: *moment, present, now*;

  *unless the FOR PP occurs after a noun but before the verb, when NP attachment is signalled.
- *Otherwise there is attachment of the interval to an NP.

Those rules above that are marked with an asterisk (*) are also those used for distinguishing between the different

uses of IN in (Brée, in press). The rules for detecting the idiomatic and special uses are obviously going to be different between these two prepositions. The rules for discriminating Duration from Interval uses show a large overlap. There are some minor differences, most probably due to the size of the samples, that I believe could be avoided by the inclusion of exceptions under both prepositions. Discriminating the `tor−` use from other Duration uses is again largely the same. However, the heuristics for detecting the `tor+` use are very different. (With IN topicalization and modals were the main clues.) The `tor+` use with IN indicates that the matrix holds at a time after the `tor` by the duration indicated by the NP, rather than between the `tor` and this time, as is the case with FOR, so it is unlikely that a common set of heuristic rules will serve to detect the `tor+` use for both prepositions. For the Interval uses, the heuristics for detecting NP attachment and count use are very similar and could be generalised. (No heuristics were provided for detecting VP attachment for IN PPs.)

Clearly providing a common set of (heuristic) rules for discriminating between the different uses (except for `tor+`) for both FOR and IN is the next order of the day. There are also two further prepositions, both infrequent, that have both Duration and Interval uses and so should be examined with the same heuristics in mind: WITHIN, OVER.

5. Conclusions

For such a fine grained analysis as we have made here, a million word corpus is about the maximum size that can be managed by hand; but it is clearly not large enough to give testable results. A corpus of the order of 10 million words, such as the British National Corpus, is needed. To perform this type of analysis such a corpus would have first to be suitably coded. What features would be need to be coded to make the distinctions by the method we have developed?

Some features are easily coded:

- First and foremost the temporal use of FOR, and other prepositions, can be readily detected from the noun in their PP. A list of about 100 such nouns is already available (Brée & Pratt, 1997). Some nouns are marginally temporal, e.g. *in history*, but these are mercifully few. This should overcome the problem, in the latest Penn Tree version of the Brown corpus, of tagging as temporal many non-temporal uses of prepositions .
- The distinction between different types of temporal noun, as given in Section 3., would be useful, in particular, recognising seasons, short time periods, such as *now, present, instant, second, moment, minute* and that some nouns, such as *a while*, give vague times, would be useful.
- The idiomatic use of temporal prepositions needs to be recognised, here simply *for the N^{th} time*.
- The contrast between Definite and Indefinite determiners needs to be recognised. Mostly this is easy, but the distinction between a 'Zero' and a 'None' entry needs to be made on the basis of the nature of the noun. Perhaps this is best left to the user, but then a provision has to be made for recognising proper nouns.
- The presence of a superlative in the PP as, while this is Definite, it is not a signal for an Interval use, but for a Duration use.
- The aspect of the verb in the matrix.
- The distinction between pre-determiners, determiners and post-determiners needs to be made along the lines in (Quirk et al., 1985).
- The position of the FOR PP relative to other components in its matrix, in particular the verb.

Other features are more difficult to code:

- To recognise the special use of FOR, the verb in the matrix must be coded as one of prediction, planning or purpose.
- Recognising those qualifiers, such as *past, last, next*, which attach a duration to the `tor`.
- Recognising those post-modifiers, such as *beginning, commencing, following*, which attach a duration to the `tor`.
- Distinguishing the temporal use of *then* to signal the `tor` from other uses, such as inferential.
- It would also assist the analysis if the entire matrix could be categorised as a state, process or event, or something similar to Vendler's types, although this feature has not been explicitly used here.

This is some wish list. But most features would be useful not only for distinguishing between the different temporal uses of FOR but for all the other temporal prepositions.

6. References

D. S. Brée. in press. The semantics of temporal IN. In C. Zelinsky, editor, .

D. S. Brée and I. E. Pratt. 1997. Using prepositions to tell the time. In M. G. Shafto and P. Langley, editors, *Proceedings of the Nineteenth Conference of the Cognitive Science Society*, page 873, New Jersey. Erlbaum.

J. Hitzeman. 1997. Semantic partition and the ambiguity of sentences containing temporal adverbials. *J. Natural Language Semantics*, 5:87–100.

H. Kučera and W. N. Frances. 1967. *Computational analysis of present-day American English*. Brown University Press.

R. Quirk, S. Greenbaum, G. Leech and J. Svartvik. 1985. *A comprehensive grammar of the English language*. Longmans.

H. Reichenbach. 1947. *Elements of symbolic logic*. Macmillan.

Z. Vendler. 1957. Verbs and times. *The Philosphical Rev.*, 66:143–160.

A Language-Neutral Representation of Temporal Information

Richard Campbell*, Takako Aikawa, Zixin Jiang, Carmen Lozano, Maite Melero and Andi Wu

Microsoft Research
One Microsoft Way, Redmond, WA 98052 USA
{richcamp, takakoa, jiangz, clozano, maitem, andiwu}@microsoft.com
*to whom inquiries should be addressed

Abstract

We propose a framework for representing semantic tense that is language-neutral, in the sense that it represents what is expressed by different tenses in different languages in a shared formal vocabulary. The proposed framework allows the representation to retain surface distinctions for particular languages, while allowing fully semantic representations, such as a representation of event sequence, to be derived from it. The proposed framework also supports the incorporation of semantic tense information that does not derive from grammatical tense, but derives instead from other expressions such as time adverbials. The framework is currently implemented in NLPWin, a multi-lingual, multi-application natural language understanding system currently under development at Microsoft Research, but the representational framework is in principle independent of any particular system.

1. Introduction[1]

Multilingual applications face (at least) two problems in the domain of semantic tense: First, there is the problem that grammatical, or morphological, tenses in different languages do not mean the same thing. In English, for example, grammatical past tense situates an event prior to the utterance ("speech time" in Reichenbach's (1947) terminology), and grammatical present tense situates an event simultaneous with the utterance. In contrast Japanese past tense situates an event prior to some reference time, and present tense situates an event simultaneous with some reference time, where the reference time may or may not be the utterance time. Neither language has a tense that expresses exactly what is expressed by past or present in the other. This poses a problem for applications such as machine translation (MT), since a given grammatical tense in one language does not automatically translate into the same surface tense in another language:

(1) 彼女は病気だと言った。
kanozyo-wa [byooki **da**] to itta
she -Top sick be-**Pres** that say-Past
'she said [she **was** sick]'

In (1), for example, the grammatical present tense in the embedded clause (indicated by brackets) translates into English as grammatical past tense, both of which allow the interpretation that the event described in the embedded clause is simultaneous with that described in the main clause.

Another problem is that what is expressed as grammatical tense in one language is sometimes only expressible as an adverbial construction in another language. For example, Chinese has no grammatical tense per se (see Section 3.3 for more details); consequently a single form can in principle express past, present or future; this is illustrated in the following examples:

(2) 昨天他来看我
zuotian ta lai kan wo
yesterday he come see me
'Yesterday he came to see me.'

(3) 明天他来看我
mingtian ta lai kan wo
tomorrow he come see me
'Tomorrow he will come to see me.'

In (2) and (3), the adverbials *zuotian* 'yesterday' and *mingtian* 'tomorrow' are all that indicate that these sentences are set in the past and future, respectively.

In this paper, we propose a framework for representing semantic tense, by which we mean information about the sequence of events. Our framework is *language-neutral*, in the sense that it represents surface tense marking of various languages using a shared formal vocabulary. Our framework also allows the incorporation of semantic tense information that is not expressed as grammatical tense, for example, that (2) is about a past time. Also, since a large part of what is expressed by tenses concerns the sequence of events and states, one aspect of our framework is enabling an explicit representation of temporal sequence. The analyses reported here are currently implemented in the NLPWin system under development at Microsoft Research (Heidorn, 2000).

Most (if not all) other proposals for a language-neutral representation of tense, such as Van Eynde (1997), are explicit attempts to represent the semantics of tense directly. However, the kind of semantic representation of tense may vary considerably depending on application. For example, some applications may require tense to be represented in first-order predicate calculus, perhaps incorporating Davisonian event arguments (Davidson, 1980), while others might require only an explicit sequence of events, as in Filatova and Hovy (2001).

The novelty of our approach lies in the fact that it does not attempt to be a particular semantic representation. Our goal is to preserve syntactic information about semantic tense so that various semantic representations of

[1] We would like to thank three anonymous reviewers and our colleagues in the Natural Language Processing group at MSR for their helpful comments and discussion, especially Michael Gamon, Marisa Jimenez, Jessie Pinkham and Hisami Suzuki.

tense can be constructed if necessary for a particular application. For example, our representation is compatible with both the referential theory of tense (e.g. Enç, 1987) and the quantificational theory of tense (e.g. Ogihara, 1995). Also, although it does not express sequence of events directly, a representation of such a sequence can be derived from our representation.

Our framework owes much to Reichenbach (1947); but while a strictly Reichenbachian approach to tense may work well for European languages, such an approach becomes unwieldy when faced with a set of languages with more typologically diverse tense systems, including Japanese and Chinese, aspects of which are discussed below. We therefore do not rely on the Reichenbachian notions of reference and event times, as does e.g. Van Eynde (1997), but adapt what we take to be Reichenbach's essential insights to a wider range of tense systems.[2]

Before proceeding, it is necessary to say something about the terms *tense* and *aspect*, and to lay out what the scope of the paper is. By *semantic tense*, we mean information about how events or situations are sequenced; this includes some of what in some traditions is called aspect, such as the interpretation of the English perfect, etc. It also includes information that may not be recorded by grammatical tense, as shown in (2) and (3). By *aspect*, we mean temporal information that goes beyond temporal sequence, such as (im)perfectivity, progressive, stative, habitual, and the like. In this paper, we are concerned with semantic tense, not primarily with aspect, though some aspectual features are considered in Section 3.3.2, below.

The paper is organized as follows: In Section 2 we outline the general framework of Language-Neutral Syntax (LNS) (Campbell, 2002; Campbell & Suzuki, 2002), within which we situate the current proposal; in Section 3, we lay out our proposal for the representation of semantic tense; in Section 4, we compare our system to other proposals for representing semantic tense; Section 5 offers a conclusion.

2. Language-neutral syntax

In this section we describe the basic properties and motivation for LNS. For more detailed descriptions, the reader is referred to Campbell (2002) and Campbell & Suzuki (2002).

LNS is a level of representation that is more abstract than a surface-syntactic analysis, yet not as abstract as a fully-articulated semantic analysis; rather, it is intermediate between the two. The basic design principle of LNS is that it be close enough to the surface syntax of individual languages to allow reconstruction of the surface structure of a given sentence (i.e., LNS can serve as the input to a language-particular generation function), yet abstract and language-independent enough to allow derivation of deeper semantic representations, where necessary, by a language-independent function. The role of LNS is illustrated schematically in Figure 1.

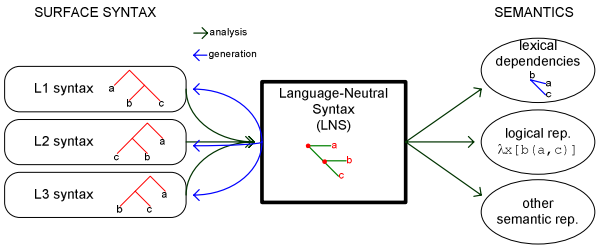


Figure 1: Language-Neutral Syntax

The primary motivation for such an intermediate representation is to mediate between languages in multi-lingual applications, given that fully articulated semantic representations are typically not needed in most such applications. For example, the Adjective + Noun combinations *black cat* and *legal problem* have identical surface structures, but very different semantics: the first is interpreted as λx[black(x) & cat(x)], i.e., as describing anything that is both a cat and black; the second, however, does not have the parallel interpretation as a description of something which is both a problem and legal: rather, it typically describes a problem having to do with the law. To accurately analyze this distinction would require extensive and detailed lexical annotation for adjective senses and, most likely, for lexicalized meanings of particular Adj + Noun combinations; such extensive annotation, if it is even possible, would make a system that depends on it very brittle. For most applications, however, this semantic difference is immaterial, and the extensive and brittle annotation unnecessary: for example, all that we need to know to translate these phrases into French *chat noir* lit. 'cat black' and *probléme legal* lit. 'problem legal' is that the adjective modifies the noun in some way. LNS is a representation in which *black cat* and *legal problem* have the same structure, despite their deep semantic difference, and in which *black cat* and *chat noir* have the same structure, despite their superficial syntactic difference.

An LNS representation is an annotated tree, in which constituents are unordered, and linked to their parent by labeled arcs, the labels corresponding to semantically motivated grammatical functions such as semantic head, logical subject, time, etc. The LNS tree is annotated with semantically motivated features and relations expressing long-distance dependencies (such as binding and control) and discourse-oriented functions (such as topic and focus). An example (somewhat simplified, and with tense not represented for the time being) is given below; this figure represents the LNS for this noun phrase before the implementation of the framework for tense representation presented below.

(4) the cat that was seen yesterday

```
NOMINAL1 (+Def +Sing)
|_SemHeads--cat1
|_L_Attrib--FORMULA1 (+Pass +Proposition)
                |_SemHeads--see1
                |_L_Sub---_X1
                |_L_Obj---NOMINAL2
                          |_SemHeads--that1
                          |_Cntrlr: cat1
                |_L_Time-- yesterday1
```

[2] However, we do use the terms "reference time" and "event time" informally below.

The root node (NOMINAL1) is in the upper left; the daughters of a given node are indicated by labeled arcs such as *SemHeads* (semantic head), *L_Attrib* (logical attributive modifier), *L_Obj* (logical object), and the like. In addition to these attributes indicating deep grammatical relatons, there are other attributes which express additional relations among nodes in the tree. For example, the relative pronoun NOMINAL2 has a *Cntrlr* attribute, whose value is *cat1*, and indicates that *cat1* is the antecedent of the relative pronoun. The Cntrlr attribute is not part of the LNS tree per se; that is, the value of Cntrlr must be part of the LNS tree independently of the Cntrlr relation (in this case, as the semantic head of NOMINAL1). We refer to attributes such as Cntrlr as *non-tree* attributes. For display purposes only in this paper, we display non-tree attributes as labeled arcs, even though they are not part of the LNS tree per se; they will be displayed slightly differently, however, in that the value of the attribute is introduced by a colon, instead of by a dashed line.

In this example we see also that passives are normalized in terms of their argument structure, but the fact that the relative clause is passive is recorded in the feature +Pass on FORMULA1. This reflects a basic design principle of LNS: The basic structure is normalized for variation both within and among languages, but surface distinctions (such as the active/passive distinction) are retained as much as possible.

Thus an LNS representation needs to be close enough to the surface syntax to indicate meaningful distinctions, yet abstract enough to normalize meaningless cross-linguistic variation.

3. Framework for semantic tense

The LNS representation of semantic tense must therefore satisfy the following design criteria:

(5) *Design criteria for LNS representation of tense*:
a. Each individual grammatical tense in each language is recoverable from LNS.
b. The explicit sequence of events entailed by a sentence is recoverable from LNS by a language-independent function.

Criterion (5)(a) says that we must be able to reconstruct, by a distinct generation function for each language, how the semantic tense was expressed in the surface form of that language; this criterion will be satisfied if the LNS representation is different for each tense in a particular language. Criterion (5)(b) says that we must be able to derive an explicit representation of the sequence of events from an LNS representation by means of a language-independent function. This criterion will be satisfied if the representation of each tense in each language is truly language-neutral. In this section we detail a framework for semantic tense that meets the design criteria in (5). We begin by giving the details of the basic formalism (which we will add to in subsequent subsections), followed by a discussion of the motivation and function of its various aspects.

3.1. Basic framework: simple tenses

3.1.1. Tense features and relations

In our proposal each tensed clause contains a distinct Tense node, which is in the *L_Tense* ("logical tense") relation with the clause, and which is specified with semantic tense features, representing the meaning of each particular tense, and attributes indicating its relation to other nodes (including other Tense nodes) in the LNS tree. Semantic tense features can be either *global* or *anchorable*.[3]

The basic tense features, along with their interpretations, are given in the following tables; Table I shows the global features, and Table II the anchorable ones ('U' stands for the utterance time: 'speech time' in Reichenbachian terms):[4]

Feature	Meaning
G_Past	before U[5]
G_NonPast	not before U
G_Future	after U

Table I: Global tense features

Feature	Meaning
Befor	before Anchr if there is one; otherwise before U
NonBefor	not before Anchr if there is one; otherwise not before U
Aftr	after Anchr if there is one; otherwise after U
NonAftr	not after Anchr if there is one; otherwise not after U

Table II: Anchorable tense features

The tense features of a given Tense node are determined on a language-particular basis according to the interpretation of individual grammatical tenses. For example, the simple past tense in English is [+G_Past], the simple present is [+G_NonPast +NonBefor], etc.

Additional features may turn out, on further analysis, to be necessary; for example, many languages make a grammatical distinction between immediate future and general future, or between recent past and remote or

[3] The distinction between global and anchorable tense features is very similar to Comrie's (1985) distinction between 'absolute' and 'relative' tenses. We have adopted the different terminology to emphasize that the global/anchorable distinction is for features, not for tenses per se, as in Comrie's taxonomy.

[4] Note that, given their meanings, some pairs of Tense features are semantically incompatible with each other, and cannot occur on the same node. For example, a given Tense cannot be [+G_Past +G_NonPast].

[5] Strictly speaking the meaning of the global tense features is to express a relation between a given time t and a globally specified reference time, G. Conceivably, the value of G could vary, depending on various factors including genre, discourse context, etc. However, we currently have no theory as to how G might be set to any value other than U, so we will assume throughout that the global referene time is always the same as the utterance time.

general past. We have nothing to say about these specific contrasts, however, other than to note that the framework we propose is flexible enough to accommodate new tense features, if necessary.

A Tense node T will also under certain conditions have a non-tree attribute called *Anchr*, which indicates a relation that T bears to some other Tense node (the value of the Anchr attribute must be another Tense node). Like other non-tree attributes such as Cntrlr, Anchr should be thought of as an annotation on the basic tree, not as part of the tree itself; that is, the value of the Anchr attribute must fit into the LNS tree in some independent way. A Tense node has an Anchr attribute if (a) it has anchorable tense features; and (b) meets certain structural conditions. For simple tenses, the structural condition that it must meet to have an Anchr is that the clause containing it is an argument (i.e., logical subject or object) of another clause; in this case the value of Anchr is the Tense node in the governing clause. In the discussion of compound tenses below we will augment the set of sufficient structural conditions for having an Anchr.[6]

3.1.2. Past tense in English and Japanese

As indicated in Table II, if a Tense node with anchorable features has no Anchr, then it is interpreted as if anchored to the utterance time U. This means that, for example, a [+G_Past] Tense and an unanchored [+Befor] Tense have the same interpretation, all else being equal. Consider the following English and Japanese sentences, with the relevant parts of their LNS structure shown:[7]

(6) She was sick.
FORMULA1
|_SemHeads----sick1
|_L_Tense----_Tense1 (**+G_Past**)

(7) 彼女は病気だった。
kanozyo-wa byooki datta
she -Top sick be-Past
'She was sick.'
FORMULA1
|_SemHeads----病気1 (sick)
|_L_Tense--_Tense1 (**+Befor**)

The English and Japanese past tenses are represented differently because they are semantically different, though in these simple examples that difference is neutralized. The English simple past tense is [+G_Past], indicating that it denotes a time that is before U. The Japanese simple past tense on the other hand is [+Befor], indicating that it denotes a time that is before its Anchr. However, in this simple root sentence, there is no Anchr, so it is interpreted as if anchored to U; hence the interpretation is before U. Thus the design criterion (5)(b) is met, at least for these simple cases: a simple language-independent function would yield the correct sequence *be_sick* < *U* for both these examples.

The semantic difference between the English and Japanese past tenses comes into play when the Anchr attribute is present, which for simple tenses is in clauses that are arguments of a higher clause. Consider the following English and Japanese examples, in which the tense in question (in boldface) is in an embedded sentence (indirect speech), represented in LNS as the logical object (L_Obj) of the matrix clause:

(8) She said she was sick.
FORMULA1
|_SemHeads--say1
|_L_Tense--_Tense1 (+G_Past)
|_L_Obj--FORMULA2
|_SemHeads--sick1
|_L_Tense--**_Tense2** (+G_Past)

(9) 彼女は病気だったと言った。
kanozyo-wa byooki datta to itta
she -Top sick be-Past that say-Past
'she said she was sick'
FORMULA1
|_SemHeads--言う1 (say)
|_L_Tense--_Tense1 (+Befor)
|_L_Obj--FORMULA2
|_SemHeads--病気1 (sick)
|_L_Tense--**_Tense2** (+Befor)
|_Anchr: _Tense1

Since the embedded tense in (8) is +G_Past, its interpretation is before U; left unspecified is whether the situation described by the embedded clause (FORMULA2) is reported to have occurred before, or simultaneous with, the situation described by the matrix clause. In fact, both interpretations are possible in this case: her reported sickness may be simultaneous with her saying that she was sick (i.e., she said "I am sick"), or it may have preceded it (i.e., she said "I was sick").[8] The structure we assign to it captures that underspecification succinctly.

In (9), on the other hand, the embedded tense, *_Tense2*, is +Befor; since it has an anchorable feature, and its clause is the logical object of another clause, it must be anchored to the tense of that matrix clause, i.e., to *_Tense1*. Consequently, it denotes a time that is before the time denoted by *_Tense1* (which, like *_Tense1* in (7), denotes a time before U). So the only interpretation (9) has is that her reported sickness is prior to her saying that she was sick; i.e., it can only mean 'she said "I was sick"'; it cannot mean 'she said "I am sick"'. This construction illustrates the essential difference between the English and Japanese past tense forms: the former directly expresses a

[6] We have not ruled out the possibility of language-particular anchoring conditions, but so far have not encountered any need for them.

[7] In this paper we show only the parts of the LNS necessary to illustrate the treatment of tense; for example, we leave out logical subject, etc., unless otherwise necessary. Note also that the copula is regularly omitted from the LNS (see Campbell, 2002).

[8] A third logical possibility, consistent with the interpretation of G_Past, is that her sickness was in the past (i.e., before U), but after her saying that she was sick; i.e., she said "I will be sick". But this kind of interpretation seems to be universally disallowed without some kind of irrealis marking on the clause (such as a modal), and therefore does not need to be separately indicated.

relation to U, while the latter directly expresses a relation to some "reference" time, which may or may not be U.

Examples such as (8) and (9) illustrate precisely why the English and Japanese grammatical past tenses have different representations in the current framework. Suppose for example that the Japanese past tense were [+G_Past] (like the English past), instead of [+Befor]; then Japanese (9) should have the same range of interpretations as English (8), in particular it should be able to serve as a description of an event in which she said "I am sick"—i.e., where the time of her being sick coincides with the time that she said she was sick. As noted, however, this interpretation is not available for (9), as it is for (8).

Our analysis of the English and Japanese past tenses differs from the approach taken by e.g. Ogihara (1995), who claims that English and Japanese past tenses mean the same thing, and that differences such as that between (8) and (9) below are due to the optional application in English of a rule that deletes the embedded past tense from the logical form component. Our analysis gives a uniform description to both the English and Japanese grammatical past tenses.

It is important to note that there is only one sense of the feature Befor (the same holds true for all the anchorable features in Table II), and hence only one meaning for Japanese past tense, in our system. This is a crucial point which is easily overlooked: phrased in strictly Reichenbachian terms, we may appear to be saying that the Japanese past tense means *either* E<R (if it is anchored) *or* E<S (if not anchored). But this appearance of bi-vocalism is due, we believe, to an overly rigid adherence to Reichenbach's notation; our own notation is more flexible, allowing us to characterize the Japanese past tense as univocal, while still retaining what we regard as Reichenbach's essential insights, namely that some tenses relate to U and others to a structurally determined "reference" time.

3.1.3. Present tense in English and Japanese

Another good illustration of the differences between global and anchorable tense features is provided by the English and Japanese present tenses. As in the case of past tense, the two tenses receive the same interpretation in simple sentences:

(10) She is sick.
FORMULA1
|_SemHeads—sick1
|_L_Tense--_Tense1 (**+G_NonPast** +NonBefor)

(11) 彼女は病気だ。
kanozyo-wa byooki da
she-Top sick be-Pres
'She is sick'
FORMULA1
|_SemHeads—病気1 (sick)
|_L_Tense--_Tense1 (**+NonBefor**)

Since the English present tense in (10) is [+G_NonPast] (as well as [+NonBefor]; see below), it must denote a time that is not before U. The Japanese present tense is just [+NonBefor], so it denotes a time that is not before its Anchr; since it lacks an Anchr, in this case, it must denote a time that is not before U. Consequently (10) and (11) receive the same interpretation.

Note that nothing in these representations directly expresses anything about the "present": G_NonPast is interpreted as "not before" U, but does not have to be simultaneous with U. This is by design: the English grammatical present tense allows a future interpretation as well as a "present" one, as in *We speak tomorrow* (see Section 4, below). Our assumption is that present-time reference is the default denotation for any Tense whose features and relations to other time expressions are consistent with that interpretation. Similar comments hold for the Japanese present tense, which is [+NonBefor] in our analysis. As in English, the Japanese present tense also allows a future-time construal (see Section 3.3.3, below).

As in the case of the past tenses, the difference between the English and Japanese present tenses shows up when there is an Anchr:

(12) She said she is sick.
FORMULA1
|_SemHeads--say1
|_L_Tense--_Tense1 (+G_Past)
|_L_Obj--FORMULA2
|_SemHeads--sick1
|_L_Tense--**_Tense2 (+G_NonPast +NonBefor)**
|_ Anchr: _Tense1

(13) 彼女は病気だと言った。
kanozyo-wa byooki da to itta
she -Top sick be-Pres that say-Past
'she said she was sick'
FORMULA1
|_SemHeads--言う1 (say)
|_L_Tense--_Tense1 (+Befor)
|_L_Obj--FORMULA2
|_SemHeads--病気1 (sick)
|_L_Tense--**_Tense2 (+NonBefor)**
|_ Anchr: _Tense1

In this case, both embedded tenses are anchored, since both have the anchorable feature [+NonBefor]. The English present tense is [+G_NonPast], however, so *_Tense2* denotes a time that is not before U; it is also [+NonBefor], so it also denotes a time that is not before the (past) time denoted by *_Tense1*. Consequently, the period of her sickness must overlap both the time of her saying that she was sick and the utterance time U (see also Note 8); in fact, as Enç (1987) notes, this construction has exactly that meaning. This example also illustrates the fact that a given tense may have any collection of mutually-compatible tense features, including both global and anchorable ones.

In contrast, the Japanese example (13) (the same as (1)) does not imply that the period of her sickness includes the utterance time; instead, the possibility that she is still sick at the present moment is left open, unlike (12). In our framework, this is because the Japanese present lacks a global tense feature. *_Tense2* is [+NonBefor] and not [+G_NonPast] like (12), so its only requirement is that it denote a time that is not before the time denoted by its Anchr, *_Tense1*. As indicated in the gloss, the best English translation of (13) is with the past tense. Examples like (12) and (13) illustrate precisely why the

English and Japanese present tenses are to be represented differently.

3.2. Compound tenses

One of the great insights of Reichenbach's (1947) analysis of tense is his treatment of compound tenses, such as the English present- and past-perfect. In this subsection, we outline our representation of compound tenses, which, despite notational differences, is essentially Reichenbachian.

We begin by making a formal distinction between *primary* and *secondary* tenses, the latter being tenses, such as English *have* + past participle, which require an Anchr within the same clause, the former being all others. Thus each language-particular tense must be specified as to its features, and whether it is primary or secondary. Consider the following example of the past perfect in English:

(14) He had arrived.
FORMULA1
|_SemHeads—arrive1
|_L_Tense--_Tense1 (+G_Past)
--_Tense2 (**+Befor**)
|_ **Anchr: _Tense1**

We treat English perfect constructions as consisting of two tenses: a secondary tense that is [+Befor], anchored to a primary tense, in this case simple past (hence [+G_Past]). There is no principled upper limit to the number of Tense nodes in a given clause (though particular grammars presumably impose de facto limits), though the following conditions must be met for well-formedness: (1) each clause has one and only one Tense that is not anchored within the clause (though it may be anchored outside the clause); this is the Tense that designates the "reference" time; and (2) each clause has one and only one Tense which is not the Anchr of another Tense in the same clause (though it may be the Anchr of another Tense in another clause); this is the Tense that designates the "event" time. In (14), the first condition is satisfied by *_Tense1*, and the second condition is satisfied by *_Tense2*. In the simple tense examples discussed in Section 3.1, both conditions are satisfied by the same Tense node.

The advantages of treating the perfect construction as a compound tense, instead of as a simple tense, are two-fold: (1) it allows us to distinguish English present perfect and simple past without additional features (thus helping to satisfy the design criterion (5)(a)); and (2) it captures the fact that the perfect construction co-occurs with every simple tense in English, with the same interpretation.

3.3. Survey of tenses across languages

The framework described above is not a theory of tense, in that it does not uniquely determine a representation for each grammatical tense in each language, but provides a language-neutral vocabulary for expressing differences among grammatical tenses across typologically diverse languages. To implement the framework in an NLP system, then, we need to have actual analyses of specific tenses. In this section we provide such analyses for several tenses in several languages.

3.3.1. English

The discussion above gives examples of the past, present and perfect tenses in English and their combinations. Here we give two more examples of English grammtical tenses: the future with *will*[9] and the past with *used to*.

Future: Though an argument might be made that the future with *will* is actually a compound tense, we take the simpler route here and analyze it as a distinct primary tense with the feature [+G_Future], as in the following example:

(15) You will be sick.
FORMULA1
|_SemHeads—sick1
|_L_Tense--_Tense1 (+G_Future)

Past with used to: The past tense formed with *used to*, as in *he used to work here*, like the simple past tense is [+G_Past], but differs from the simple past not only in aspectual properties (not treated here), but also in that it has the anchorable feature [+Befor]. Consider the following example:

(16) He said he used to work here.
FORMULA1
|_SemHeads—say1
|_L_Tense--_Tense1 (+G_Past)
|_L_Obj—FORMULA2
|_SemHeads—work1
|_L_Tense--_Tense2 (**+G_Past +Befor**)
|_ **Anchr: _Tense1**

Since the embedded *_Tense2* is [+Befor], it denotes a time that is not only before U, but also before the (past) time denoted by *_Tense1*. This reflects the fact that in (16), the time that he worked here must be before the time that he said he used to work here (compare to (8), above); that is, it can only mean that he said "I used to work here", and cannot mean that he said "I work here".

3.3.2. Other European languages

Apart from aspectual differences, the tense systems of Western European languages such as French, German and Spanish are very similar to that of English. The aspectual differences are of course important, and must be represented in LNS. Although a complete discussion of aspect goes beyond the scope of the present paper, we include a brief discussion of some diferences between English and Spanish here.

One notable difference between Spanish and English is that Spanish has two distinct grammatical past tenses, the perfective, or preterite, and the imperfective. The

[9] Needless to say, this is not the only way to express future-time reference in English. The simple present can sometimes be used, and there are at least two other constructions that are future only: *be going to* + infinitive, and *be about to* + infinitive. The latter construction has a different meaning from the others (immediate future), and should be distinguished, perhaps with a feature. The difference between *will* and *be going to* is hard to detect, if it exists at all, but in keeping with design criterion (5)(a) they should be distinguished in some way.

difference is entirely aspectual, and does not appear to affect the interpretation of sequence of events per se. Another notable difference between English and these other languages is that most of them use the simple grammatical present tense to refer to an event ongoing at the utterance time, as in the following Spanish example:

(17) Llueve.
rain-Pres
'It's raining.'

The simple present in English, however, cannot be used this way; English *it rains* has only a generic or habitual sense.

For both of these distinctions, a feature indicating the aspectual difference is used; in our system, the relevant features are *Discrete* and *NonDiscrete*; the former indicating that events are viewed in their entirety, the latter that events are subdivided into arbitrarily small subintervals. Thus the Spanish preterite is [+Discrete], while the imperfect is unmarked for either of these features. Also, the simple present in English is [+Discrete], while the simple present in e.g. Spanish is umarked for this feature.

Aside from such aspectual differences, the most notable tense difference between Spanish and English is that the Spanish present progressive, in contrast to the simple present, is incompatible with future time reference:

(18) Vuelvo mañana.
return-1sg tomorrow
'I return/am returning tomorrow'

(19) Estoy volviendo (*mañana).
be-1sg returning tomorrow
'I am returning tomorrow.'

This is handled by assigning the present progressive the features [+G_NonPast +NonBefor +NonAftr] (in addition to aspectual features), which differs from the simple present in being [+NonAftr]. In (19) there is no Anchr, so the [+NonAftr] feature dictates that the time referred to is not after U; i.e, is not in the future; this accounts for this tense's incompatibility with a future time adverbial.

3.3.3. Japanese

The discussion above gives some examples of the simple past and present in Japanese, analyzed in our framework as [+Befor] and [+NonBefor], respectively. Since there is no separate future tense in Japanese, future time reference is normally achieved with the simple present tense, as in the following example:

(20) 明日雨が降る。
ashita ame-ga furu
tomorrow rain-Nom fall-Pres
'Tomorrow, it will rain.'
FORMULA1
|_SemHeads—降る1 (fall)
|_L_Time—明日1 (tomorrow) (+G_Future)
|_L_Tense--_Tense1 (+NonBefor)

The feature [+NonBefor] on *_Tense1* is compatible with future time reference, as discussed in Section 3.1.3, above. The future, as opposed to present, reading of (20) comes from the presence of the adverbial *ashita* 'tomorrow'. In Section 4, we discuss how semantic tense information from adverbials is incorporated into our framework.

3.3.4. Chinese

Unlike the other languages discussed above, Chinese has no grammatical tense. As noted in the introduction vis-a-vis examples (2) and (3), semantic tense, when expressed, is often expressed via adverbials, and not with grammatical tense; this is discussed in more detail in Section 4, below. However, Chinese does have a limited number of particles, traditionally referred to as aspect markers, which, besides indicating aspect, also indicate semantic tense information. The aspectual meaning of these particles is beyond the scope of this paper, but we will discuss a few examples to show how they express semantic tense, and how that information is represented in our framework.

We discuss here the particles *le*, *guo* and *jiang*, as in the following examples:

(21) 他说他买**了**书
ta shuo ta mai **le** shu
he say he buy Aspect book
'He says/said that he has/had bought books.'
FORMULA1
|_SemHeads--说1 (say)
|_L_Tense--_Tense1
|_L_Obj—FORMULA2
|_SemHeads--买1 (buy)
|_L_Tense--**_Tense2 (+Befor)**
|_Anchr: _Tense1

(22) 他说他买 **过**书
ta shuo ta mai **guo** shu
he say he buy Aspect book
'He says/said that he has/had (once) bought books.'
FORMULA1
|_SemHeads--说1 (say)
|_L_Tense--_Tense1
|_L_Obj—FORMULA2
|_SemHeads--买1 (buy)
|_L_Tense--**_Tense2 (+Befor)**
|_Anchr: _Tense1

(23) 他说他**将**到美国去
ta shuo ta **jiang** dao meiguo qu
he say heAspect to US go
'He says/said that he will/would go to the US.'
FORMULA1
|_SemHeads--说1 (say)
|_L_Tense--_Tense1
|_L_Obj—FORMULA2
|_SemHeads--买1 (buy)
|_L_Tense--**_Tense2 (+Aftr)**
|_Anchr: _Tense1

In all these examples, the tense of the main clause (*_Tense1*) has no features; we take this to be the default case in Chinese, in which an unmarked clause can be interpreted as past, present or future (see the discussion of examples (2) and (3) in the Introduction, and Section 4,

below). However, aspectual particles such as *le*, *guo* and *jiang* can also contribute semantic tense information, which we represent as if it were grammatical tense.

The particles *le* and *guo* are both [+Befor] (their difference is aspectual, not represented here); in (21) and (22), the embedded clause Tense is anchored to the matrix, indicating that the buying of books took place before his saying. In contrast, *jiang* in (23) is [+Aftr], so this example means that the going to the US takes place after his saying.

4. Deriving semantic tense from syntactic context

It is often the case that semantic tense information is not represented as grammatical tense per se, but can come, at least in part, from adverbials or other features of the syntactic environment. We have seen that this is one of the main sources of semantic tense information in Chinese; an example from English is *We speak tomorrow*, which is grammatically present tense (hence [+G_NonPast +NonBefor], but semantically is unambiguously about the future. To deal with this situation, we propose to augment the framework outlined in Section 3 with an additional non-tree attribute *Spcfrs*, which indicates, for a given Tense node, any other temporal expressions in the clause that contributes to the semantic tense of that clause. Like Anchr, Spcfrs is not part of the LNS tree per se, but is an annotation on the tree. The representation is given below:

(24) We speak tomorrow.
FORMULA1
|_SemHeads—speak1
|_L_Time—tomorrow1 (+G_Future)
|_L_Tense--_Tense1 (+G_NonPast +NonBefor)
|_ Spcfrs: tomorrow1

_Tense1 has only the features of any present tense, so the representation satisfies the first design criterion (5)(a); but its Spcfrs is the adverb *tomorrow1*, which itself has the feature [+G_Future], since tomorrow is unambiguously in the future. This relation indicates to the language-independent function that derives the explicit sequential representation that the temporal reference of the clause is to a time that is after U, thus satisfying the second design criterion (5)(b).

Note that design criterion (5)(a) is satisfied in another way, as well: the structure of (24) is different from the structure of a sentence with a future tense, which presumably makes use of the feature [+G_Future] (see below); thus the distinction between the "scheduled" future (Comrie, 1985) in (24) and the more basic future of *We will speak tomorrow* is preserved.

The need for the Spcfrs relation is much more prevalent in languages that make little or no use of grammatical tense, such as Chinese. Consider the following examples:

(25) 昨天他来看我
zuotian ta lai kan wo
yesterday he come see me
'Yesterday he came to see me.'
FORMULA1
|_SemHeads--来1 (come)
|_L_Time--昨天1 (yesterday) (+G_Past)
|_L_Tense--_Tense1
|_Spcfrs: 昨天1

(26) 明天他来看我
mingtian ta lai kan wo
tomorrow he come see me
'Tomorrow he will come to see me.'
FORMULA1
|_SemHeads--来1 (come)
|_L_Time--明天1 (tomorrow) (+G_Future)
|_L_Tense--_Tense1
|_Spcfrs: 明天1

The Spcfrs relation thus permits specification of semantic tense features that are not expressed as grammatical tense.

5. Comparison to other frameworks

Our proposal is for a system of representation of semantic tense that is language-neutral; i.e., that represents the tense distinctions of different languages in a formal vocabulary that has the same meaning in all languages. As such, our proposal is very different from proposals to represent the semantics of tense in a particular language such as English, both in the obvious respect that we consider other languages, and in the less obvious respect that our proposal is not a semantic one in any deep sense, but rather a syntactic representation that is language-neutral, as sketched in Section 2 (Campbell & Suzuki, 2002).

As such, the nearest thing to a comparable proposal that we have encountered in the computational literature is Van Eynde (1997), which explicitly provides a Reichenbachian semantic framework for multiple languages, and incorporates information from temporal adverbs in addition to grammatical tense. Unlike our proposal, however, Van Eynde's framework is explicitly Reichenbachian, characterizing tenses in terms of three possible values for sTENSE, expressing the relation between the reference and speech times, and six values for sASPECT, expressing the relation between the event and reference times. Although our framework encodes the same essential insight, it does so without rigidly adhering to the reference time/event time distinction, which leads to a simpler representation in our view.

6. Application

Having a language-neutral representation of semantic tense has clear implications for multi-lingual applications such as MT. Consider again the Japanese example (13), in which an embedded present tense is to be translated into past tense in English. A simple transfer of the language-particular present tense yields the wrong result, since *She said she is sick* (=(12)) means something very different from (13). Instead, what needs to be transferred is the whole temporal structure of *_Tense2*, including its features and its Anchr, since this is the information that

determines that it denotes a time that is before U. Such context-sensitive transfers are possible in an MT system such as that described by Richardson, *et al*. (2001).

Similarly, consider the Chinese example (25), in which there is no grammatical tense specified. A Chinese-English MT system must transfer not the grammatical tense (which yields no information whatsoever), but rather the whole temporal structure, which in this case includes its Spcfrs, in order to give the English generation system the information it needs to generate past tense.

7. Conclusion

We have presented and exemplified a framework for representing semantic tense in a language-neutral fashion, which meets the competing design criteria in (5): that each language-particular tense be reconstructible by a generation function, and that an explicit representation of temporal sequence be derivable by means of a language-independent function.

The framework we have proposed allows us to get semantic tense information from grammatical tense, or from adverbial modifiers, and represents this information in a semantically motivated, language-neutral fashion.

8. References

Campbell, R., 2002. *Language-neutral syntax*. MSR Tech Report (in preparation).

Campbell, R. & H. Suzuki. 2002. A language-neutral representation of syntactic structure. SCANALU-2002.

Comrie, B., 1985. *Tense*. Cambridge University Press.

Davidson, D. 1980. The logical form of action sentences. In D. Davidson, ed., *Essays on actions and events*, 105-122. Clarendon Press, Oxford.

Enç, M., 1987. Anchoring conditions on Tense. *Linguistic Inquiry* 18, 633-657.

Filatova, E. & E. Hovy. 2001. Assigning time-stamps to event-clauses. In Proceedings of ACL-EACL.

Heidorn, G.E. 2000. Intelligent writing assistance. In R. Dale, H. Moisl and H. Somers, eds., *Handbook of natural language processing*. Marcel Dekker, New York.

Ogihara, T. 1995. The Semantics of Tense in Embedded Clauses. *Linguistic inquiry* 26, 663-679.

Reichenbach, H. 1947. *Elements of symbolic logic*. The Free Press, New York, and Collier-Macmillan, London.

Richardson S., W. Dolan, A. Menezes and J. Pinkham. 2001. Achieving commercial-quality translation with example-based methods. In *Proceedings of the VIIIth MT summit*, Santiago de Compostela, Spain. 293-298.

Van Eynde, F. 1997. Mood, Tense & Aspect. In F. Van Eynde and P. Schmidt (eds.), *Linguistic specifications for typed feature structure frameworks*. EU Commission, Luxembourg.

A Representational Scheme for Temporal and Causal Information Processing

Eleni Galiotou[1,2], Gérard Ligozat[3]

[1]Department of Informatics, TEI of Athens
Ag. Spyridona, GR-122 10 Egaleo, Greece,
[2]Dept. of Informatics and Telecommunications, University of Athens
Panepistimiopolis, GR-157 84 Athens, Greece
egali@di.uoa.gr

[3]LIMSI-CNRS, Paris-Sud University
Bldg. 508, P.O.Box 133, F-91403, Orsay, France
ligozat@limsi.fr

Abstract

In this paper, we propose a representational scheme for temporal and causal information processing which is based on the exploitation of linguistic elements of narrative texts having a temporal and/or causal interest. We show that the framework of generalized intervals provides an adequate representational device for temporal information as it is conveyed from descriptions in natural language. The linguistic elements which are used in order to determine the temporal structure of the text are: temporal adverbials, verbs, derived nouns, temporal and/or causal connectives. Reasoning over the resulting temporal constraint network allows the disambiguation of ambiguous temporal relations and provides answers to questions concerning the temporal order of events in the text, their relative positioning and their causal interpretation.

1. Introduction

The extraction and processing of temporal information from natural language texts requires an effective representational scheme which facilitates the answer to questions such as: «What happened»?, «Why?», «Did event A take place before event B?», etc. In this paper, we propose such a scheme and we focus on its theoretical justification.

We show that the framework of generalized intervals (Ligozat 1991; Ligozat 1997; Bestougeff and Ligozat 1992) which extends Allen's interval framework (Allen 1983; Allen 1984) provides an adequate representational device for temporal information as it is conveyed from descriptions in natural language. The formalism of generalized intervals allows a direct representation of qualitative temporal relations possessing an internal structure. Therefore, it is suitable for the representation of events which are classified according to the tripartite ontology of events (Moens 1987; Moens and Steedman 1988). This representation allows reference to their internal structure and proves to be well adapted to the representation of temporal information

In our approach, the linguistic elements which are used in order to determine the temporal structure of the text are: temporal adverbials, verbs, derived nouns, temporal and/or causal connectives. The above mentioned elements contribute to determining the order of events in the text, their relative position, their degree of completion and their duration.

Reasoning over the temporal constraint network which reflects the temporal structure of the text allows the disambiguation of ambiguous temporal relations and provides answers to questions concerning the temporal order of events in the text, their relative positioning and their causal interpretation.

The structure of the paper is as follows: In section 2 the theoretical assumptions of the approach are presented. Section 3 is concerned with the representational scheme while in section 4 the final conclusions are presented.

2. Theoretical assumptions

Our choice of a representation language is motivated by the complexity and richness of the linguistic information required for the processing of temporal and causal phenomena. Indeed, most of the well known formalisms such as the interval-based framework proposed by Allen (1983; 1984) prove to be inadequate for the representation of complex linguistic information. In the following we briefly present the formalism of generalized intervals as proposed by Ligozat (1991; 1996; 1997) and Bestougeff and Ligozat (1992).

2.1. The generalized interval framework

In this approach, the main temporal object is a a poly-typed string (henceforth PTS) which consists of a generalized interval, i.e. an increasing sequence of typed boundaries (time points) corresponding to an event, and its associated linguistic information. The typing of the boundaries (opening, closing, undefined) allows to reason over the degree of completion of an action and therefore proves to be adequate for the representation of the grammatical aspect. A sequence of n typed boundaries is called an n-string and it defines a partitioning of time into 2n+1 zones numbered from 0 to 2n. Odd numbers correspond to boundaries and even numbers correspond to open intervals defined by these boundaries. Using this numbering we are able to define all the combinatorial possibilities of relations between two n-strings: Consider a p-string X and a q-string Y. The relations between X and Y is determined by specifying for each boundary of X which zone of Y it belongs to. We thus obtain a non-decreasing sequence of p integers between 0 and 2q.

A sequence of polytyped strings reflecting a consistent piece of discourse is called a temporal site. This temporal site is represented by a temporal constraint network, that is, a graph where the edges correspond to typed strings and the arcs to relations between them.

2.2. Events in the generalized interval framework

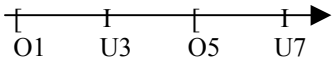
The processing of temporal and causal information as it is conveyed by natural language requires an ontology which is constituted on the cognitive basis of relations such as causation and enablement (grouped under the more general notion of consequentiality) between events and states rather than on purely temporal primitives. An ontology which satisfies the above requirements and is also able to capture most of the relevant linguistic phenomena is the tripartite ontology of events (Moens 1987; Moens and Steedman 1988). The basic element of this ontology is an «event nucleus» which is composed by a preparatory process, leading to a culmination point followed by a consequent state. In this way, we can explain the fact that the same event (for example, the construction of a bridge) can be used by referring to its preparatory process (drawing the plans), to its culmination point (the bridge was inaugurated) or to its consequent state (traffic problems were solved). Temporal and aspectual categories as conveyed by natural language utterances are classified into states and events. Following (Parsons 1990) we will use the term *eventuality* in order to describe an event or a state. Events are further classified into culminated processes, processes, culminations and points (punctual events which are not associated to a preparatory process or consequent state). Reference to the appropriate part of the nucleus determines the exact nature of the event or state at hand.

3.2.3. Representing temporal knowledge

In the generalized interval framework, different types of processes are associated to typical schemata in terms of sequences of typed boundaries.

In the following we sketch the representation of temporal knowledge in the generalized interval framework (Galiotou 1999; Galiotou and Ligozat 1997).

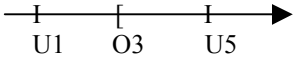
A culminated process (the nucleus) situated in the past will be represented by a sequence of typed boundaries numbered by odd numbers from 1 to 7.



Preparatory process: (1 3), Culmination point: 3, Consequent state: (3 5), Speech time: 7, Aspect: Perfect, Tense: Past

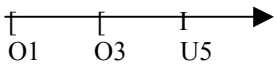
The boundaries take their values among the following: **[** (**O**pening boundary), **]** (**C**losing boundary), **U** (**U**ndefined boundary)

A culmination situated in the past will be represented by a sequence of three boundaries as in the following:



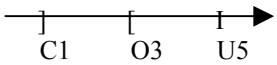
Culmination point: 1, Consequent state: (1 3), Speech time: 5, Aspect: Perfective, Tense:Past

A process situated in the past will represented as:



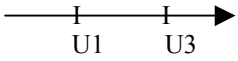
Process: (1 3), Speech time: 5 , Aspect: Imperfective, Tense: Past

A state will represented as:



State: (1 3), Speech time: 5, Aspect: Imperfective, Tense: Past

A punctual event will be represented as:

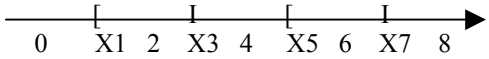


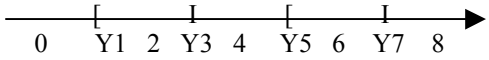
Point: 1, Speech time: 3, Aspect: Perfective, Tense: Past

3.2.3. Building the temporal site

Using this schematic representation we try to express the corresponding temporal relations (see also Galiotou 1999, Galiotou and Ligozat 1997). In the following, we give an example of a temporal relation expressed by means of relations between typed boundaries:

Suppose that we consider two 4-intervals X and Y each referring to an event nucleus. X is defined by the sequence of boundaries x_1, x_2, x_3, x_4 and Y is defined by the the sequence of boundaries y_1, y_2, y_3, y_4. The variables x_i and y_i , $i \in \{1,3,5,7\}$ have a type among the following: {O(pening), C(losing), U(ndefined)}. The open intervals defined by the boundaries will be numbered by even numbers. So, the representation of the two 4-intervals will be:





The relation of temporal precedence will be represented by the following:

([0,2], [0,2], [0,6], 7)

The culmination point of X (x_3) is related to the culmination point of Y (y_3) by a relation of temporal precedence while the position of x_5 is not completely determined.

Based on this representation, we build a temporal constraint network reflecting the temporal structure of a text. Actually, the arcs of the network correspond to two types of relations: on the one hand, purely temporal relations, and on the other hand discourse relations reflecting the hierarchical structure of the text and having a temporal or causal significance (Asher 1993): narration, elaboration, explanation, background, and result. We also take into account the discourse relation of contrast which has a rather non-temporal character but proves to be useful in establishing contextual constraints for the processing of causal information.

At this point of the representation we can anticipate different ways of processing (Bestougeff and Ligozat 1992):

- Restrict *a priori* the number of possible relations between two n-intervals in a natural language utterance.
- In case of an incomplete graph, calculate the possible relations between two generalized intervals using the formula of composition of relations between two generalized intervals (Ligozat 1991). An application of this procedure can be found in (Galiotou 1999).
- Verify that the set of temporal relations in the temporal constraint network is consistent.
- Verify that updating the network does not introduce any inconsistency.

Every time a new set of constraints between two generalized intervals is introduced in the temporal constraint network, the set of relations is updated. For the propagation of constraints in the graph we use a variant of Allen's constraint propagation algorithm (Allen 1983; Allen 1984) suitably adapted for the generalized interval framework (Ligozat 1991; Galiotou 1999).

3. A proposal for a representational scheme

Before proceeding with the definition of the scheme, we have to specify the linguistic elements of the text which contribute to determining the temporal relations.

3.1. Linguistic elements in the text

In our experimentation we have used a corpus of short newspaper articles in Modern Greek describing car accidents. The study of temporal and causal phenomena in such a corpus had two goals:

- extract the events which have led to an accident as well as its consequences;
- provide all plausible explanations of the causes of the accident.

In general, the causes of the accident are explicitly mentioned in the text using causal connectives and/or causal expressions. This explicit causal information contributes to the determination of the temporal order and the relative positioning of events in conjunction with the «Causes precede Effects» rule.

Our corpus consists of narrative texts composed of simple clauses containing temporal adverbials, subordinates introduced by temporal and/or causal connectives, verb phrases, derived nouns and adjectives. In this type of narration, events are situated in the past so, verbal forms of the future do not appear in the text. The most frequent verbal forms are those of the preterite, while the present tense is used in order to describe consequent states. As far as grammatical aspect is concerned, all possible values for Greek are present, i.e. imperfective, perfective and perfect (Mackridge 1985). The lexical aspect follows the tripartite ontology of events as already stated in section 2.

Temporal adverbials are classified (Tzevelekou 1995) into:

- those which require a different starting point from the starting point of the narration: kat' ar`xin[1] (at first), sti si`nexia (afterwards) etc.
- those which establish their reference either from the starting point of the narration or form another starting point: ta ksime`romata tis kiriakis (on Sunday morning), `liγo prin ta me`sanixta tis paraske`vis (short before Friday midnight) etc.
- deictic adverbs which establish their reference uniquely from the starting point of the narration: xθes (yesterday), pro`xθes (the day before yesterday), etc.

Temporal clauses are introduced by temporal connectives which on the one hand provide information on the temporal order of events and on the other hand help determine implicit causal relations (i.e. causal relations which are not expressed by causal markers in the text). Consider the case of the temporal connective `otan (when). Its use in the corpus is in accordance with the observation that its basic meaning is not temporal in the first place (Moens and Steedman 1988). Indeed, `otan (when) establishes a consequentiality relation between the main clause event and the subordinate clause event provided that the eventualities in the text follow the organizational scheme of the tripartite ontology of events. Therefore, it is a candidate for expressing causality in the text. Take for example, the case of the construction

Q(Past-perfective) `otan P(Pat-Perfective)

In this case the processes Q and P are related with a consequentiality relation as in the following example:

tris `nei `exasan ti zo`i tus otan to afto`kinito tus ana`trapike
Three young men lost their lives when their car overturned.

Explicit causal information in the text is expressed by causal connectives such as *epi`δi (because)* of causal expressions such as *me apo`telesma (with result*) or *`loγo (because of).*

It is also expressed by noun phrases which are introduced in simple sentences such as:

e`pisimi e`tia tu δisti`ximatos ine i olisθi`rotita tu `δromu
The official cause of the accident is the slipperiness of the road.

or, verbal phrases with a causal sense such as:

to δi`stixima o`filete stin ipervoli`ki ta`xitita tu aftoki`nitu
The accident is due to the excessive speed of the car.

The combination of a causal expression with a derived noun phrases such as *i olisθi`rotita tu `δromu (the slipperiness of the road), i api`ria tu oδi`γu (the inexperience of the driver), I aprose`ksia ton pe`zon (the inattention of the pedestrians)* etc is used in order to

[1] Greek words are transcribed according to the characters of the International Phonetic Alphabet. When necessary, stress is indicated with symbol « ` » before any stressed syllable.

establish discourse relations and to build the temporal structure of the text. For example:

to δi`stixima o`filete stin apì`ria tu oδi`γu a`la ke stin aprose`ksia ton pe`zon

The accident is due to the inexperience of the driver but also, to the inattention of the pedestrians.

Here, a relation of explanation holds between the propositions and therefore, a relation of consequentiality between the corresponding eventualities. The derived nominal represents a state. So, the consequentiality relation is specified as an enablement relation between the state represented by the derived nominal and the main clause event which in this case is the accident. Thus we obtain an overlap relation between the related temporal intervals.

3.2. The representational scheme

The theoretical assumptions described in section 2 lead to the definition of a representational scheme taking into account the linguistic elements of the text which are described in the previous subsection. As already stated, our treatment of temporal information in the text is based solely on the exploitation of linguistic knowledge.

Therefore, we propose a single-dimensional scheme consisting of a list of labels relative to the temporal and/or causal knowledge as it is expressed by the linguistic elements of the text.

Note that, since the starting point of our research was the study of temporal and causal phenomena in Modern Greek, we had to take into account the particularities of the temporal and aspectual system of the language in elaborating a suitable tag set. Yet, in our opinion, this does not constitute a major problem since the set of values is easily extendable in order to capture particularities of other languages provided that the general guidelines are kept.

The text segments to be annotated are taken at the clause level.

3.2.3. The tag set

The proposed tag set takes into account the following classes:

BOUNDARY_LIST : The generalized interval expressed in terms of a list of typed boundaries for example,

<U1, 03, U5>

TENSE: Values are limited to

<past>, <non_past>

since in Modern Greek future is considered as modal.

ASPECT: <imperfective>, <perfective>, <perfect>

LEXASP: The values follow the tripartite ontology of event and they are limited to:

<state>, <culm_process>, <process>, <culmination>, <point>.

The number of categories could easily be modified provided that the principles of the ontology are respected.

CONNECTIVE: We take into account causal and/ or temporal connectives therefore, the proposed tags are:

<cn_causal>, <cn_temp>, <cn_ct>

TEMPADV: Temporal adverbials play a crucial role in our processing of temporal and causal information but there was no need for a more fine-grained characterization. So, the only possible tag is:

<tempadv>

In the current state of the scheme we have decided to encode dates under the <tempadv> tag.

CUE_PHRASE: Cue_phrases contribute to the determination of temporal and explicitly causal information, so they enter our scheme with the tag:

<cue_phrase>

3.2.3. The decision rules

Following (Klein, 1999) we provide a decision tree aiming at giving an overview of all possible tags and how they are related to each other. We also provide rules that help to mutually constraint the tags between categories and ensure the coherence of the annotation:

In figure 1, we give an example of the decision rules :

if the segment is tagged with <culmination> and <past >
then label with <U1 O3 U5>
….
if the segment is tagged with <past> and <perfect>
then label with <culm_process>
…
if the segment is tagged with <past> and <imperfective>
then label with <state>
…

Figure 1. Example of decision rules

3.2.3. Examples and evaluation

In table 1 we give an example of the application of the scheme to a fragment of a text describing a car accidents (in Modern Greek).

A quick overview this application has shown that as far as the temporal entities are concerned there was an approximately even distribution of culminated processes and culminations. and to a lesser extent states and processes. The fifth category of temporal entities, namely that of punctual events, did not appear in any text.

As for the verbal forms, there was no occurrence of the future tense. The most frequent forms were those of the preterite, while the present tense was used in order to describe the consequent states of events and conclusions.

The distribution of temporal and causal connectives indicated that the most frequent appearance is that of the *`otan (when)* connective used both as a temporal and as a causal one.

We must also report an extensive use of cue_phrases for the expression of causal knowledge in the text. Contrary to what one may have expected in a corpus of short newspaper articles describing car accidents, causal connectives were not the most frequent elements used in order to express causality. Explicit causal information was mostly put forward using cue phrases in simple sentences.

The scheme proved to be quite easy to apply by the human annotator. Yet, the tagging procedure, as far the boundary list was concerned proved to be more difficult to implement. This has led us to the conclusion that the annotation procedure should be automated at least as far as the boundary_list category is concerned.

ORIGINAL TEXT	ANNOTATED TEXT
tris `nei `exasan ti zo`i tus otan to afto`kinito tus ana`trapike (Three young men lost their lives when their car overturned).	<U1 O3 U5> <culm> tris `nei <past><imperfective>`exasan </past></imperfective> ti zo`i tus </culm> </U1 O3 U5> <O1 U3 O5 U7> <culm_process> <cn_temp>otan </cn_temp> to afto`kinito tus <past><perfective>ana`trapike</past></perfective> </culm_process></O1 U3 O5 U7>

Table 1: Application of the scheme to a fragment of a text

4. Conclusions

In this paper, we proposed a representational scheme which provides the background for the processing of temporal and causal information.

We have insisted on the theoretical assumptions of the scheme concerning:

- the temporal positioning of events;
- the appropriate temporal ontology;
- the interaction of temporal and causal information in texts;
- the causal interpretation of events.

We also discussed the nature of the linguistic elements of a text which can be used to determine the temporal structure of the text.

We then briefly described a representational scheme based on the theoretical assumptions mentioned above.

This representational scheme was applied to a corpus of short newspaper articles describing car accidents in Modern Greek. These articles were selected with respect to their informational interest and the temporal and causal relations they contained. Therefore, the particularities of the temporal and aspectual system of the Greek language were taken into account in the proposed tag set. Nevertheless, this tag set could easily be modified to take into account the particularities of other languages provided that the general approach is followed.

The scheme has proved to be quite reliable in its application to the particular corpus. Yet, the task of the human annotator would be greatly facilitated if at least certain procedures such as the tagging with the boundaries_list notation were automated.

5. References

Allen, J.F., 1983. Maintaining Knowledge about Temporal Intervals. *Communications of the ACM 26,* 11:832-843.

Allen, J.F., 1984. Towards a General Theory of Action and Time. *Artificial Intelligence, 23,* 123-154.

Asher, N., 1993. *Reference to Abstract Objects of Discourse*, Dordrecht:Kluwer.

Bestougeff, H. and G. Ligozat, 1992. *Tools for Temporal Knowledge Representation*, Academic Press.

Galiotou, E., 1997. Temporal and Causal relations in Greek Texts. In A. Ralli, M. Grigoriadou, G. Philokyprou, D. Christodoulakis and E. Galiotou (eds.) *Working Papers in Natural Language Processing*, Athens: Diavlos, 109-122.

Galiotou, E., 1999. *Modélisation de la Causalité à partir de l'Analyse des Phénomènes Temporels dans les Langues Naturelles: Application au Grec Moderne,* Ph.D. Thesis, Université Paris-Sud, (=Publication Notes et Documents LIMSI 2000-11, LIMSI-CNRS).

Galiotou, E. and G. Ligozat, 1997. Représentation du Temps et de l'Aspect en vue de la Génération d'Explications Causales. *Actes GAT'97*, Université Stendhal, Grenoble, 85-96.

Kitis, E., 1995. Connectives and Ideology. *Proceedings of the 4th International Symposium on Critical Discourse Analysis: Language and Social Thought*, University of Athens.

Klein, M., 1999. Standardisation Efforts on the Level of Dialogue Act in the MATE Project. *Proceedings of the ACL Workshop: Towards Standards and Tools for Discourse Tagging,* University of Maryland, 85-41.

Lascarides, A., and N. Asher, 1993. *Temporal Interpretation, Discourse Relations and Common Sense Entailment*, Research Paper HCRC/RP-16, University of Edinburgh.

Ligozat, G., 1991. On Generalized Interval Calculi. *Proceedings of AAAI'91*, 234-240.

Ligozat, G., 1996. A New Proof of Tractability for ORD-Horn Relations. *Proceedings of AAAI'96,* 395-401.

Ligozat, G., 1997. Time, Space and Natural Language Processing. In Ralli et al. (eds.), Athens: Diavlos, 155-174.

Mackridge, P., 1985. *The Modern Greek Language*, Oxford University Press.

Moens, M., 1987. *Tense, Aspect and Temporal Reference*, Ph.D. Thesis, University of Edinburgh.

Moens, M. and M. Steedman, 1988. Temporal Ontology and Temporal Reference. *Computational Linguistics, 14*, 2: 15-28.

Parsons, T., 1990. *Events in the Semantics of English*, MIT Press.

Soria, C. and V. Pirrelli, 1999. A Recognition-based Meta-scheme for Dialogue Acts Annotation,

Proceedings of the ACL Workshop: Towards Standards and Tools for Discourse Tagging, University of Maryland, 75-83.

Teufel S., and M. Moens, 1999. Discourse-level argumentation in scientific articles: human and automatic annotation, *Proceedings of the ACL Workshop: Towards Standards and Tools for Discourse Tagging,* University of Maryland, 84-93.

Tzevelekou, M., 1995. *Catégorisation Lexicale et Aspect: Le Système Aspectuel du Grec Moderne*, Ph.D. Thesis, Université Paris 7.

Toward an Ontology of Time for the Semantic Web

Jerry R. Hobbs

Artificial Intelligence Center
SRI International
Menlo Park, California 94025
USA
hobbs@ai.sri.com

Abstract

In connection with the DAML project for bringing about the Semantic Web, an ontology of time is being developed for describing the temporal content of Web pages and the temporal properties of Web services. The bulk of information on the Web is in natural language, and this information will be easier to encode for the Semantic Web insofar as community-wide annotation and automatic tagging schemes and the DAML time ontology are compatible with each other.

1. Introduction

The DARPA Agent Markup Language (DAML) project is an effort aimed at bringing into reality the Semantic Web, in which Web users and automatic agents will be able to access information on the Web via descriptions of the content and capabilities of Web resources rather than via key words. An important part of this effort is the development of representative ontologies of the most commonly used domains. We are beginning to develop such an ontology of temporal concepts, for describing the temporal content of Web pages and the temporal properties of Web services. This effort is being informed by temporal ontologies developed at a number of sites and is intended to capture the essential features of all of them and make them easily available to a large group of Web developers and users.

The bulk of information on the Web is in natural language, and this information will be easier to encode for the Semantic Web insofar as community-wide annotation and automatic tagging schemes and the DAML time ontology are compatible with each other.

In this paper I outline the temporal ontology as it has been developed so far, in order to initiate a dialog between the two communities. Five categories of temporal concepts are considered, and for each the principal predicates and their associated properties are described.

A note on notation: Conjunction ($\wedge$) takes precedence over implication($\supset$) and equivalence ($\equiv$). Formulas are assumed to be universally quantified on the variables appearing in the antecedent of the highest-level implication. Thus,

$$p_1(x) \wedge p_2(y) \supset q_1(x,y) \wedge q_2(y)$$

is to be interpreted as

$$(\forall x,y)[[p_1(x) \wedge p_2(y)] \supset [q_1(x,y) \wedge q_2(y)]]$$

2. Topological Temporal Relations

2.1. Instants and Intervals

There are two subclasses of temporal-entity: $instant$ and $interval$.

$$instant(t) \supset temporal\text{-}entity(t)$$
$$interval(T) \supset temporal\text{-}entity(T)$$

(In what follows, lower case t is used for instants, upper case T for intervals and for temporal-entities unspecified as to subtype. This is strictly for the reader's convenience, and has no formal significance.)

$start\text{-}of$ and $end\text{-}of$ are functions from temporal entities to instants.

$$temporal\text{-}entity(T) \supset instant(start\text{-}of(T))$$
$$temporal\text{-}entity(T) \supset instant(end\text{-}of(T))$$

For convenience, we can say that the start and end of an instant is itself.

$$instant(t) \supset start\text{-}of(t) = t$$
$$instant(t) \supset end\text{-}of(t) = t$$

$inside$ is a relation between an instant and an interval.

$$inside(t,T) \supset instant(t) \wedge interval(T)$$

This concept of $inside$ is not intended to include starts and ends of intervals, as will be seen below.

Infinite and half-infinite intervals can be handled by positing time instants at positive and negative infinity, and using them as start and end points.

It will be useful in characterizing clock and calendar terms to have a relation between instants and intervals that says that the instant is inside or the start of the interval.

$$in\text{-}interval(t,T) \equiv [start\text{-}of(T) = t \vee inside(t,T)]$$

$interval\text{-}between$ is a relation among a temporal entity and two instants.

$$interval\text{-}between(T,t_1,t_2) \supset temporal\text{-}entity(T) \wedge instant(t_1) \wedge instant(t_2)$$

The two instants are the start and end points of the temporal entity.

$$interval\text{-}between(T,t_1,t_2) \equiv start\text{-}of(T) = t_1 \wedge end\text{-}of(T) = t_2$$

The ontology is silent about whether the interval from t to t, if it exists, is identical to the instant t.

The ontology is silent about whether intervals *consist of* instants.

The ontology is silent about whether intervals are uniquely determined by their starts and ends.

We can define a proper interval as one whose start and end are not identical.

$$\begin{array}{l}\mathit{proper\text{-}interval}(t) \equiv \\ \quad \mathit{interval}(t) \wedge \mathit{start\text{-}of}(t) \neq \mathit{end\text{-}of}(t)\end{array}$$

The ontology is silent about whether there are any intervals that are not proper-intervals.

2.2. Before

There is a $before$ relation on temporal entities, which gives directionality to time. If temporal entity T_1 is before temporal entity T_2, then the end of T_1 is before the start of T_2. Thus, before can be considered to be basic to instants and derived for intervals.

$$\begin{array}{l}before(T_1, T_2) \\ \quad \equiv before(\mathit{end\text{-}of}(T_1), \mathit{start\text{-}of}(T_2))\end{array}$$

The end of an interval is not before the start of the interval.

$$\begin{array}{l}\mathit{interval}(T) \\ \quad \supset \ before(\mathit{end\text{-}of}(T), \mathit{start\text{-}of}(T))\end{array}$$

The start of a proper interval is before the end of the interval.

$$\begin{array}{l}\mathit{proper\text{-}interval}(T) \\ \quad \supset before(\mathit{start\text{-}of}(T), \mathit{end\text{-}of}(T))\end{array}$$

If one instant is before another, there is an interval between them.

$$\begin{array}{l}\mathit{instant}(t_1) \wedge \mathit{instant}(t_2) \wedge before(t_1, t_2) \\ \quad \supset (\exists T)\mathit{interval\text{-}between}(T, t_1, t_2)\end{array}$$

The ontology is silent about whether there is an interval from t to t.

If an instant is inside a proper interval, then the start of the interval is before the instant, which is before the end of the interval. The converse is true as well.

$$\begin{array}{l}\mathit{instant}(t) \wedge \mathit{proper\text{-}interval}(T) \\ \quad \supset [\mathit{inside}(t, T) \\ \quad\quad \equiv before(\mathit{start\text{-}of}(T), t) \\ \quad\quad\quad \wedge before(t, \mathit{end\text{-}of}(T))]\end{array}$$

Intervals are contiguous with respect to the $before$ relation, in that an instant between two other instants inside an interval is inside the interval.

$$\begin{array}{l}before(t_1, t_2) \wedge before(t_2, t_3) \\ \quad\quad \wedge \mathit{inside}(t_1, T) \wedge \mathit{inside}(t_3, T) \\ \quad \supset \mathit{inside}(t_2, T)\end{array}$$

The $before$ relation is anti-symmetric and transitive.

$$\begin{array}{l}before(T_1, T_2) \supset \neg before(T_2, T_1) \\ before(T_1, T_2) \wedge before(T_2, T_3) \\ \quad \supset before(T_1, T_3)\end{array}$$

The relation $after$ is defined in terms of $before$.

$$after(T_1, T_2) \equiv before(T_2, T_1)$$

The ontology is silent about whether time is linearly ordered.

2.3. Interval Relations

The relations between intervals defined in Allen's temporal interval calculus (Allen and Kautz, 1985) can be defined in a straightforward fashion in terms of $before$ and identity on the start and end points.

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}equals}(T_1, T_2) \\ \quad\quad \equiv \mathit{start\text{-}of}(T_1) = \mathit{start\text{-}of}(T_2) \\ \quad\quad\quad \wedge \mathit{end\text{-}of}(T_1) = \mathit{end\text{-}of}(T_2)]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}before}(T_1, T_2) \equiv before(T_1, T_2)\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}after}(T_1, T_2) \equiv after(T_1, T_2)\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}meets}(T_1, T_2) \\ \quad\quad \equiv \mathit{end\text{-}of}(T_1) = \mathit{start\text{-}of}(T_2)\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}met\text{-}by}(T_1, T_2) \\ \quad\quad \equiv \mathit{int\text{-}meets}(T_2, T_1)]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}overlaps}(T_1, T_2) \\ \quad\quad \equiv before(\mathit{start\text{-}of}(T_1), \mathit{start\text{-}of}(T_2)) \\ \quad\quad\quad \wedge before(\mathit{start\text{-}of}(T_2), \mathit{end\text{-}of}(T_1)) \\ \quad\quad\quad \wedge before(\mathit{end\text{-}of}(T_1), \mathit{end\text{-}of}(T_2))]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}overlapped\text{-}by}(T_1, T_2) \\ \quad\quad \equiv \mathit{int\text{-}overlaps}(T_2, T_1)]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}starts}(T_1, T_2) \\ \quad\quad \equiv \mathit{start\text{-}of}(T_1) = \mathit{start\text{-}of}(T_2) \\ \quad\quad\quad \wedge before(\mathit{end\text{-}of}(T_1), \mathit{end\text{-}of}(T_2)]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}started\text{-}by}(T_1, T_2) \\ \quad\quad \equiv \mathit{int\text{-}starts}(T_2, T_1)]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}during}(T_1, T_2) \\ \quad\quad \equiv (before(\mathit{start\text{-}of}(T_2), \mathit{start\text{-}of}(T_1)) \\ \quad\quad\quad \wedge before(\mathit{end\text{-}of}(T_1), \mathit{end\text{-}of}(T_2))]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}contains}(T_1, T_2) \\ \quad\quad \equiv \mathit{int\text{-}during}(T_2, T_1)]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}finishes}(T_1, T_2) \\ \quad\quad \equiv before(\mathit{start\text{-}of}(T_2), \mathit{start\text{-}of}(T_1)) \\ \quad\quad\quad \wedge \mathit{end\text{-}of}(T_1) = \mathit{end\text{-}of}(T_2)]\end{array}$$

$$\begin{array}{l}\mathit{interval}(T_1) \wedge \mathit{interval}(T_2) \\ \quad \supset [\mathit{int\text{-}finished\text{-}by}(T_1, T_2) \\ \quad\quad \equiv \mathit{int\text{-}finishes}(T_2, T_1)]\end{array}$$

In addition, it will be useful below to have a single predicate for "starts or is during". This is called $\mathit{int\text{-}in}$.

$$\begin{array}{l}\mathit{int\text{-}in}(T_1, T_2) \\ \quad \equiv [\mathit{int\text{-}starts}(T_1, T_2) \vee \mathit{int\text{-}during}(T_1, T_2)]\end{array}$$

It will also be useful to have a single predicate for intervals intersecting in at most an instant.

$$\begin{aligned}&\textit{int-disjoint}(T_1, T_2)\\ &\quad \equiv [\textit{int-before}(T_1, T_2) \vee \textit{int-after}(T_1, T_2)\\ &\qquad \vee \textit{int-meets}(T_1, T_2)\\ &\qquad \vee \textit{int-met-by}(T_1, T_2)]\end{aligned}$$

So far, the concepts and axioms in the ontology of time would be appropriate for scalar phenomena in general.

2.4. Linking Time and Events

The time ontology links to other things in the world through four predicates—*at-time*, *during*, *holds*, and *time-span-of*. We assume that another ontology provides for the description of events—either a general ontology of event structure abstractly conceived, or specific, domain-dependent ontologies for specific domains.

The term "eventuality" will be used to cover events, states, processes, propositions, states of affairs, and anything else that can be located with respect to time. The possible natures of eventualities would be spelled out in the event ontologies.

The predicate *at-time* relates an eventuality to an instant, and is intended to say that the eventuality holds, obtains, or is taking place at that time.

$$\textit{at-time}(e, t) \supset \textit{eventuality}(e) \wedge \textit{instant}(t)$$

The predicate *during* relates an eventuality to an interval, and is intended to say that the eventuality holds, obtains, or is taking place during that interval.

$$\textit{during}(e, T) \supset \textit{eventuality}(e) \wedge \textit{interval}(T)$$

If an eventuality obtains during an interval, it obtains at every instant inside the interval.

$$\textit{during}(e, T) \wedge \textit{inside}(t, T) \supset \textit{at-time}(e, t)$$

Whether a particular process is viewed as instantaneous or as occuring over an interval is a granularity decision that may vary according to the context of use, and is assumed to be provided by the event ontology.

Often the eventualities in the event ontology are best thought of as propositions, and the relation between these and times is most naturally called *holds*. *holds* can be defined in terms of *at-time* and *during*:

$$\begin{aligned}&\textit{holds}(e, t) \wedge \textit{instant}(t) \equiv \textit{at-time}(e, t)\\ &\textit{holds}(e, T) \wedge \textit{interval}(T) \equiv \textit{during}(e, T)\end{aligned}$$

The event ontology may provide other ways of linking events with times, for example, by including a time parameter in predications.

$$p(x, t)$$

This time ontology provides ways of reasoning about the t's; their use as arguments of predicates from another domain would be a feature of the ontology of the other domain.

The predicate *time-span-of* relates eventualities to instants or intervals. For contiguous states and processes, it tells the entire instant or interval for which the state or process obtains or takes place.

$$\begin{aligned}&\textit{time-span-of}(T, e)\\ &\quad \supset \textit{temporal-entity}(T) \wedge \textit{eventuality}(e)\\ &\textit{time-span-of}(T, e) \wedge \textit{interval}(T)\\ &\quad \supset \textit{during}(e, T)\\ &\textit{time-span-of}(t, e) \wedge \textit{instant}(t)\\ &\quad \supset \textit{at-time}(e, t)\end{aligned}$$

$$\begin{aligned}&\textit{time-span-of}(T, e) \wedge \textit{interval}(T)\\ &\qquad \wedge \neg \textit{inside}(t_1, T) \wedge \neg \textit{start-of}(t_1, T)\\ &\qquad \wedge \neg \textit{end-of}(t_1, T)\\ &\quad \supset \neg \textit{at-time}(e, t_1)\\ &\textit{time-span-of}(t, e) \wedge \textit{instant}(t) \wedge t_1 \neq t\\ &\quad \supset \neg \textit{at-time}(e, t_1)\end{aligned}$$

time-span-of is a predicate rather than a function because until the time ontology is extended to aggregates of temporal entities, the function would not be defined for non-contiguous eventualities. Whether the eventuality obtains at the start and end points of its time span is a matter for the event ontology to specify. The silence here on this issue is the reason *time-span-of* is not defined in terms of necessary and sufficient conditions.

The event ontology could extend temporal functions and predicates to apply to events in the obvious way, e.g.,

$$\begin{aligned}&\textit{ev-start-of}(e) = t\\ &\quad \equiv \textit{time-span-of}(T, e) \wedge \textit{start-of}(T) = t\end{aligned}$$

This would not be part of the time ontology, but would be consistent with it.

Different communities have different ways of representing the times and durations of states and events (processes). In one approach, states and events can both have durations, and at least events can be instantaneous. In another approach, events can only be instantaneous and only states can have durations. In the latter approach, events that one might consider as having duration (e.g., heating water) are modeled as a state of the system that is initiated and terminated by instantaneous events. That is, there is the instantaneous event of the start of the heating at the start of an interval, that transitions the system into a state in which the water is heating. The state continues until another instantaneous event occurs—the stopping of the heating at the end of the interval. These two perspectives on events are straightforwardly interdefinable in terms of the ontology we have provided. This is a matter for the event ontology to specify. This time ontology is neutral with respect to the choice.

3. Measuring Durations

3.1. Temporal Units

This development assumes ordinary arithmetic is available.

There are at least two approaches that can be taken toward measuring intervals. The first is to consider units of time as functions from Intervals to Reals, e.g.,

$$\begin{aligned}&\textit{minutes}\text{: Intervals} \rightarrow \text{Reals}\\ &\textit{minutes}([5:14, 5:17]) = 3\end{aligned}$$

The other approach is to consider temporal units to constitute a set of entities—call it TemporalUnits—and have a

single function *duration* mapping Intervals × TemporalUnits into the Reals.

$$duration([5:14, 5:17], *Minute*) = 3$$

The two approaches are interdefinable:

$$seconds(T) = duration(T, *Second*)$$
$$minutes(T) = duration(T, *Minute*)$$
$$hours(T) = duration(T, *Hour*)$$
$$days(T) = duration(T, *Day*)$$
$$weeks(T) = duration(T, *Week*)$$
$$months(T) = duration(T, *Month*)$$
$$years(T) = duration(T, *Year*)$$

Ordinarily, the first is more convenient for stating specific facts about particular units. The second is more convenient for stating general facts about all units.

The aritmetic relations among the various units are as follows:

$$seconds(T) = 60 * minutes(T)$$
$$minutes(T) = 60 * hours(T)$$
$$hours(T) = 24 * days(T)$$
$$days(T) = 7 * weeks(T)$$
$$months(T) = 12 * years(T)$$

The relation between days and months (and, to a lesser extent, years) will be specified as part of the ontology of clock and calendar below. On their own, however, month and year are legitimate temporal units.

In this development durations are treated as functions on intervals and units, and not as first class entities on their own, as in some approaches. In the latter approach, durations are essentially equivalence classes of intervals of the same length, and the length of the duration is the length of the members of the class. The relation between an approach of this sort (indicated by prefix *D-*) and the one presented here is straightforward.

$$(\forall T, u, n)[duration(T, u) = n$$
$$\equiv (\exists d)[D\text{-}duration\text{-}of(T) = d$$
$$\wedge D\text{-}duration(d, u) = n]]$$

At the present level of development of the temporal ontology, this extra layer of representation seems superfluous. It may be more compelling, however, when the ontology is extended to deal with the combined durations of noncontiguous aggregates of intervals.

3.2. $Hath$

The multiplicative relations above don't tell the whole story of the relations among temporal units. Temporal units are *composed of* smaller temporal units. The basic predicate used here for expressing the composition of larger intervals out of smaller clock and calendar intervals is $Hath$, from statements like "30 days hath September" and "60 minutes hath an hour." Its structure is

$$Hath(S, N, u, x)$$

meaning "A set S of N calendar intervals of type u hath the calendar interval x." That is, if $Hath(S, N, u, x)$ holds, then x is composed of the disjoint union of N intervals of type u; S is the set of those intervals. For example, if x is some month of September and S is the set of the successive days of that September, then $Hath(S, 30, *Day*, x)$ would be true.

The principal properties of $Hath$ are as follows:

The type constraints on its arguments: S is a set, N is an integer, u is a temporal unit, and x is an interval:

$$Hath(S, N, u, x)$$
$$\supset set(S) \wedge integer(N)$$
$$\wedge temporal\text{-}unit(u) \wedge interval(x)$$

The elements of S are intervals of duration u:

$$Hath(S, N, u, x)$$
$$\supset (\forall y)[member(y, S)$$
$$\supset interval(y) \wedge duration(y, u) = 1]$$

S has N elements:

$$Hath(S, N, u, x) \supset card(S) = N$$

The elements of S are disjoint:

$$Hath(S, N, u, x)$$
$$\supset (\forall y_1, y_2)[member(y_1, S)$$
$$\wedge member(y_2, S) \wedge y_1 \neq y_2$$
$$\supset int\text{-}disjoint(y_1, y_2)]$$

There are elements in S that start and finish x:

$$Hath(S, N, u, x)$$
$$\supset (\exists y_1)[member(y_1, S)$$
$$\wedge int\text{-}starts(y_1, x)]$$

$$Hath(S, N, u, x)$$
$$\supset (\exists y_2)[member(y_2, S)$$
$$\wedge int\text{-}finishes(y_2, x)]$$

Except for the first and last elements of S, every element of S has an element that precedes and follows it:

$$Hath(S, N, u, x)$$
$$\supset (\forall y_1)[member(y_1, S)$$
$$\supset [int\text{-}finishes(y_1, x)$$
$$\vee (\exists y_2)[member(y_2, x)$$
$$\wedge int\text{-}meets(y_1, y_2)]]]$$

$$Hath(S, N, u, x)$$
$$\supset (\forall y_2)[member(y_2, S)$$
$$\supset [int\text{-}starts(y_2, x)$$
$$\vee (\exists y_1)[member(y_1, x)$$
$$\wedge int\text{-}meets(y_1, y_2)]]]$$

If time is linearly ordered, the existential quantifier $\exists$ in the last four axioms can be replaced by $\exists !$.

Finally, we would like to say that the set S covers x. A simple way to say this is as follows:

$$Hath(S, N, u, x)$$
$$\supset (\forall t)[inside(t, x)$$
$$\supset (\exists y)[member(y, S)$$
$$\wedge in\text{-}interval(t, y)]]$$

That is, if an instant t is inside x, there is a smaller unit y that t is inside or the start of.

However, this is a good place to introduce notions of granularity. In describing the temporal properties of some class of events, it may make sense to specify their time with respect to some temporal unit but not with respect to a smaller temporal unit. For example, one might want to talk about an election as a point-like event being at some instant, and specifying the day that instant is in, but not specifying the hour or minute.

To accomodate this, the above axiom can be loosened by applying it only when the instant t is located in *some interval* of size u. The axiom above would be modified as follows:

$$\begin{array}{l} Hath(S, N, u, x) \\ \quad \supset (\forall t, y_1)[inside(t, x) \wedge inside(t, y_1) \\ \qquad \wedge duration(y_1, u) \\ \qquad \supset (\exists y)[member(y, S) \\ \qquad \wedge in\text{-}interval(t, y)]] \end{array}$$

Essentially, the conjuncts $inside(t, y_1) \wedge duration(y_1, u)$ specify that t can be viewed at a granularity of u.

This treatment of $Hath$ could be extended to measurable quantities in general.

3.3. The Structure of Temporal Units

We now define predicates true of intervals that are one temporal unit long. For example, $week$ is a predicate true of intervals whose duration is one week.

$$\begin{array}{l} second(T) \equiv seconds(T) = 1 \\ minute(T) \equiv minutes(T) = 1 \\ hour(T) \equiv hours(T) = 1 \\ day(T) \equiv days(T) = 1 \\ week(T) \equiv weeks(T) = 1 \\ month(T) \equiv months(T) = 1 \\ year(T) \equiv years(T) = 1 \end{array}$$

We are now in a position to state the relations between successive temporal units.

$$\begin{array}{l} minute(T) \supset (\exists S) Hath(S, 60, *Second*, T) \\ hour(T) \supset (\exists S) Hath(S, 60, *Minute*, T) \\ day(T) \supset (\exists S) Hath(S, 24, *Hour*, T) \\ week(T) \supset (\exists S) Hath(S, 7, *Day*, T) \\ year(T) \supset (\exists S) Hath(S, 12, *Month*, T) \end{array}$$

The relations between months and days are dealt with in Section 4.4.

4. Clock and Calendar

4.1. Time Zones

What hour of the day an instant is in is relative to the time zone. This is also true of minutes, since there are regions in the world, e.g., central Australia, where the hours are not aligned with GMT hours, but are, e.g., offset half an hour. Probably seconds are not relative to the time zone.

Days, weeks, months and years are also relative to the time zone, since, e.g., 2002 began in the Eastern Standard time zone three hours before it began in the Pacific Standard time zone. Thus, predications about all clock and calendar intervals except seconds are relative to a time zone.

This can be carried to what seems like a ridiculous extreme, but turns out to yield a very concise treatment. The Common Era (C.E. or A.D.) is also relative to a time zone, since 2002 years ago, it began three hours earlier in what is now the Eastern Standard time zone than in what is now the Pacific Standard time zone. What we think of as the Common Era is in fact 24 (or more) slightly displaced half-infinite intervals. (We leave B.C.E. to specialized ontologies.)

The principal functions and predicates will specify a clock or calendar unit interval to be the nth such unit in a larger interval. The time zone need not be specified in this predication if it is already built into the nature of the larger interval. That means that the time zone only needs to be specified in the largest interval, that is, the Common Era; that time zone will be inherited by all smaller intervals. Thus, the Common Era can be considered as a function from time zones to intervals.

$$CE(z) = T$$

Fortunately, this counterintuitive conceptualization will usually be invisible and, for example, will not be evident in the most useful expressions for time, in Section 4.5 below. In fact, the CE predication functions as a good place to hide considerations of time zone when they are not relevant.

Time zones should not be thought of as geographical regions. Most places change their time zone twice a year, and a state or county might decide to change its time zone, e.g., from Central Standard to Eastern Standard. Rather it is better to have a separate ontology articulate the relation between geographical regions X times and time zones. For example, it would state that on a certain day and time a particular region changes its time zone from Eastern Standard to Eastern Daylight.

Moreover, time zones that seem equivalent, like Eastern Standard and Central Daylight, should be thought of as separate entities. Whereas they function the same in the time ontology, they do not function the same in the ontology that articulates time and geography. For example, parts of Indiana are always on Eastern Standard Time, and it would be false to say that they shift in April from that to Central Daylight time.

In this treatment it will be assumed there is a set of entities called time zones. Some relations among time zones are discussed in Section 4.5.

4.2. Clock and Calendar Units

The aim of this section is to explicate the various standard clock and calendar intervals. A day as a calender interval begins at and includes midnight and goes until but does not include the next midnight. By contrast, a day as a duration is any interval that is 24 hours in length. The day as a duration was dealt with in Section 3. This section deals with the day as a calendar interval.

It is useful to have three ways of saying the same thing: the clock or calendar interval y is the nth clock or calendar

interval of type u in a larger interval x in time zone z. This can be expressed as follows for minutes:

$$min(y, n, x)$$

Because y is uniquely determined by n and x, it can also be expressed as follows:

$$minFn(n, x) = y$$

For stating general properties about clock intervals, it is useful also to have the following way to express the same thing:

$$clock\text{-}int(y, n, u, x)$$

This expression says that y is the nth clock interval of type u in x. For example, the proposition $clock\text{-}int(10:03, 3, *Minute*, [10:00, 11:00])$ holds.

Here u is a member of the set of clock units, that is, one of *Second*, *Minute*, or *Hour*.

In addition, there is a calendar unit function with similar structure:

$$cal\text{-}int(y, n, u, x)$$

This says that y is the nth calendar interval of type u in x. For example, the proposition $cal\text{-}int(12Mar2002, 12, *Day*, Mar2002)$ holds. Here u is one of the calendar units *Day*, *Week*, *Month*, and *Year*.

The unit *DayOfWeek* will be introduced below in Section 4.3.

The relations among these modes of expression are as follows:

$$\begin{aligned}
&sec(y, n, x) \equiv secFn(n, x) = y \\
&\quad \equiv clock\text{-}int(y, n, *sec*, x) \\
&min(y, n, x) \equiv minFn(n, x) = y \\
&\quad \equiv clock\text{-}int(y, n, *min*, x) \\
&hr(y, n, x) \equiv hrFn(n, x) = y \\
&\quad \equiv clock\text{-}int(y, n, *hr*, x) \\
&da(y, n, x) \equiv daFn(n, x) = y \\
&\quad \equiv cal\text{-}int(y, n, *da*, x) \\
&mon(y, n, x) \equiv monFn(n, x) = y \\
&\quad \equiv cal\text{-}int(y, n, *mon*, x) \\
&yr(y, n, x) \equiv yrFn(n, x) = y \\
&\quad \equiv cal\text{-}int(y, n, *yr*, x)
\end{aligned}$$

Weeks and months are dealt with separately below.

The am/pm designation of hours is represented by the function $hr12$.

$$\begin{aligned}
&hr12(y, n, *am*, x) \equiv hr(y, n, x) \\
&hr12(y, n, *pm*, x) \equiv hr(y, n + 12, x)
\end{aligned}$$

Each of the calendar intervals is that unit long; a calendar year is a year long.

$$\begin{aligned}
&sec(y, n, x) \supset second(y) \\
&min(y, n, x) \supset minute(y) \\
&hr(y, n, x) \supset hour(y) \\
&da(y, n, x) \supset day(y) \\
&mon(y, n, x) \supset month(y) \\
&yr(y, n, x) \supset year(y)
\end{aligned}$$

A distinction is made above between clocks and calendars because they differ in how they number their unit intervals. The first minute of an hour is labelled with 0; for example, the first minute of the hour [10:00,11:00] is 10:00. The first day of a month is labelled with 1; the first day of March is March 1. We number minutes for the number just completed; we number days for the day we are working on. Thus, if the larger unit has N smaller units, the argument n in clock−int runs from 0 to $N - 1$, whereas in $cal\text{-}int$ n runs from 1 to N. To state properties true of both clock and calendar intervals, we can use the predicate $cal\text{-}int$ and relate the two notions with the axiom

$$cal\text{-}int(y, n, u, x) \equiv clock\text{-}int(y, n - 1, u, x)$$

The type constraints on the arguments of $cal\text{-}int$ are as follows:

$$\begin{aligned}
&cal\text{-}int(y, n, u, x) \\
&\quad \supset interval(y) \wedge integer(n) \\
&\quad \wedge temporal\text{-}unit(u) \wedge interval(x)
\end{aligned}$$

There are properties relating to the labelling of clock and calendar intervals. If N u's hath x and y is the nth u in x, then n is between 1 and N.

$$\begin{aligned}
&cal\text{-}int(y, n, u, x) \wedge Hath(S, N, u, x) \\
&\quad \wedge member(y, S) \\
&\quad \supset 0 < n <= N
\end{aligned}$$

There is a 1st small interval, and it starts the large interval.

$$\begin{aligned}
&Hath(S, N, u, x) \\
&\quad \supset (\exists y)[member(y, S) \wedge cal\text{-}int(y, 1, u, x)] \\
&Hath(S, N, u, x) \wedge cal\text{-}int(y, 1, u, x) \\
&\quad \supset int\text{-}starts(y, x)
\end{aligned}$$

There is an nth small interval, and it finishes the large interval.

$$\begin{aligned}
&Hath(S, N, u, x) \\
&\quad \supset (\exists y)[member(y, S) \\
&\quad \wedge cal\text{-}int(y, N, u, x)] \\
&Hath(S, N, u, x) \wedge cal\text{-}int(y, N, u, x) \\
&\quad \supset int\text{-}finishes(y, x)
\end{aligned}$$

All but the last small interval have a small interval that succeeds and is met by it.

$$\begin{aligned}
&cal\text{-}int(y1, n, u, x) \wedge Hath(S, N, u, x) \\
&\quad \wedge member(y_1, S) \wedge n < N \\
&\quad \supset (\exists y_2)[cal\text{-}int(y_2, n + 1, u, x) \\
&\quad \wedge int\text{-}meets(y_1, y_2)]
\end{aligned}$$

All but the first small interval have a small interval that precedes and meets it.

$$\begin{aligned}
&cal\text{-}int(y_2, n, u, x \wedge Hath(S, N, u, x) \\
&\quad \wedge member(y_2, S) \wedge 1 < n \\
&\quad \supset (\exists y_1)[cal\text{-}int(y_1, n - 1, u, x) \\
&\quad \wedge int\text{-}meets(y_1, y_2)]
\end{aligned}$$

If time is linearly ordered, the existential quantifier $\exists$ can be replaced by $\exists !$ in the above axioms.

4.3. Weeks

A calendar week starts at midnight, Saturday night, and goes to the next midnight, Saturday night. It is independent of months and years. However, we can still talk about the nth week in some larger period of time, e.g., the third week of the month or the fifth week of the semester. So the same three modes of representation are appropriate for weeks as well.

$$\begin{aligned} &wk(y,n,x) \equiv wkFn(n,x) = y \\ &\quad \equiv cal\text{-}int(y,n,*Week*,x) \end{aligned}$$

As it happens, the n and x arguments will often be irrelevant.

A calendar week is one week long.

$$wk(y,n,x) \supset week(y)$$

The day of the week is a temporal unit (*DayOfWeek*) in a larger interval, so the three modes of representation are appropriate here as well.

$$\begin{aligned} &dayofweek(y,n,x) \\ &\quad \equiv dayofweekFn(n,x) = y \\ &\quad \equiv cal\text{-}int(y,n,*DayOfWeek*,x) \end{aligned}$$

Whereas it makes sense to talk about the nth day in a year or the nth minute in a day or the nth day in a week, it does not really make sense to talk about the nth day-of-the-week in anything other than a week. Thus we can restrict the x argument to be a calendar week.

$$dayofweek(y,n,x) \supset (\exists n_1, x_1) wk(x,n_1,x_1)$$

The days of the week have special names in English.

$$\begin{aligned} &dayofweek(y,1,x) \equiv Sunday(y,x) \\ &dayofweek(y,2,x) \equiv Monday(y,x) \\ &dayofweek(y,3,x) \equiv Tuesday(y,x) \\ &dayofweek(y,4,x) \equiv Wednesday(y,x) \\ &dayofweek(y,5,x) \equiv Thursday(y,x) \\ &dayofweek(y,6,x) \equiv Friday(y,x) \\ &dayofweek(y,7,x) \equiv Saturday(y,x) \end{aligned}$$

For example, $Sunday(y,x)$ says that y is the Sunday of week x.

A day of the week is also a day of the month (and vice versa), and thus a day long.

$$\begin{aligned} &(\forall y)[[(\exists n,x) dayofweek(y,n,x)] \\ &\quad \equiv [(\exists n_1,x_1) da(y,n_1,x_1)]] \end{aligned}$$

One correspondance will anchor the cycle of weeks to the rest of the calendar, for example, saying that January 1, 2002 was the Tuesday of some week x.

$$\begin{aligned} &(\forall z)(\exists x) Tuesday(dayFn(1, monFn(1, \\ &\qquad yrFn(2002, CE(z)))), x) \end{aligned}$$

We can define weekdays and weekend days as follows:

$$\begin{aligned} &weekday(y,x) \\ &\quad \equiv [Monday(y,x) \vee Tuesday(y,x) \\ &\quad \vee Wednesday(y,x) \vee Thursday(y,x) \\ &\quad \vee Friday(y,x)] \\ &weekendday(y,x) \\ &\quad \equiv [Saturday(y,x) \vee Sunday(y,x)] \end{aligned}$$

4.4. Months and Years

The months have special names in English.

$$\begin{aligned} &mon(y,1,x) \equiv January(y,x) \\ &mon(y,2,x) \equiv February(y,x) \\ &mon(y,3,x) \equiv March(y,x) \\ &mon(y,4,x) \equiv April(y,x) \\ &mon(y,5,x) \equiv May(y,x) \\ &mon(y,6,x) \equiv June(y,x) \\ &mon(y,7,x) \equiv July(y,x) \\ &mon(y,8,x) \equiv August(y,x) \\ &mon(y,9,x) \equiv September(y,x) \\ &mon(y,10,x) \equiv October(y,x) \\ &mon(y,11,x) \equiv November(y,x) \\ &mon(y,12,x) \equiv December(y,x) \end{aligned}$$

The number of days in a month have to be spelled out for individual months.

$$\begin{aligned} &January(m,y) \\ &\quad \supset (\exists S) Hath(S,31,*Day*,m) \\ &March(m,y) \supset (\exists S) Hath(S,31,*Day*,m) \\ &April(m,y) \supset (\exists S) Hath(S,30,*Day*,m) \\ &May(m,y) \supset (\exists S) Hath(S,31,*Day*,m) \\ &June(m,y) \supset (\exists S) Hath(S,30,*Day*,m) \\ &July(m,y) \supset (\exists S) Hath(S,31,*Day*,m) \\ &August(m,y) \\ &\quad \supset (\exists S) Hath(S,31,*Day*,m) \\ &September(m,y) \\ &\quad \supset (\exists S) Hath(S,30,*Day*,m) \\ &October(m,y) \\ &\quad \supset (\exists S) Hath(S,31,*Day*,m) \\ &November(m,y) \\ &\quad \supset (\exists S) Hath(S,30,*Day*,m) \\ &December(m,y) \\ &\quad \supset (\exists S) Hath(S,31,*Day*,m) \end{aligned}$$

The definition of a leap year is as follows:

$$\begin{aligned} &(\forall z)[leap\text{-}year(y) \\ &\quad \equiv (\exists n,x)[year(y,n,(CE(z)) \\ &\qquad \wedge [divides(400,n) \\ &\qquad\quad \vee [divides(4,n) \wedge \neg divides(100,n)]]] \end{aligned}$$

We leave leap seconds to specialized ontologies.

Now the number of days in February can be specified.

$$\begin{aligned} &February(m,y) \wedge leap\text{-}year(y) \\ &\quad \supset (\exists S) Hath(S,29,*Day*,m) \\ &February(m,y) \wedge \neg leap\text{-}year(y) \\ &\quad \supset (\exists S) Hath(S,28,*Day*,m) \end{aligned}$$

A reasonable approach to defining month as a unit of temporal measure would be to specify that the start and end points have to be on the same days of successive months.

$$\begin{aligned} &month(T) \\ &\quad \equiv (\exists d_1,d_2,n,x,m) \\ &\qquad [in\text{-}interval(start\text{-}of(T),d_1) \\ &\qquad \wedge in\text{-}interval(end\text{-}of(T),d_2) \\ &\qquad \wedge da(d_1,n,monFn(m,x)) \\ &\qquad \wedge da(d_2,n,monFn(mod+(m,1,12),x))] \end{aligned}$$

Here $mod+$ is modulo addition to take care of months spaning December and January. So the month as a measure of duration would be related to days as a measure of duration only indirectly, mediated by the calendar.

To say that July 4 is a holiday in the United States one could write

$$(\forall\, d, m, y)[da(d, 4, m) \wedge July(m, y) \supset holiday(d, USA)]$$

4.5. Time Stamps

Standard notation for times list the year, month, day, hour, minute, and second. It is useful to define a predication for this.

$$time\text{-}of(t, y, m, d, h, n, s, z) \equiv in\text{-}interval(t, secFn(s, minFn(n, hrFn(h, daFn(d, monFn(m, yrFn(y, CE(z)))))))))$$

For example, an instant t has the time

5:14:35pm PST, Wednesday, February 6, 2002

if the following properties hold for t:

$$time\text{-}of(t, 2002, 2, 6, 17, 14, 35, *PST*)$$
$$(\exists\, w, x)[in\text{-}interval(t, w) \wedge Wednesday(w, x)]$$

The second line says that t is in the Wednesday w of some week x.

The relations among time zones can be expressed in terms of the time−of predicate. Two examples are as follows:

$$h < 8 \supset [time\text{-}of(t, y, m, d, h, n, s, *GMT*) \equiv time\text{-}of(t, y, m, d-1, h+16, n, s, *PST*)]$$
$$h \geq 8 \supset [time\text{-}of(t, y, m, d, h, n, s, *GMT*) \equiv time\text{-}of(t, y, m, d, h-8, n, s, *PST*)]$$
$$time\text{-}of(t, y, m, d, h, n, s, *EST*) \equiv time\text{-}of(t, y, m, d, h, n, s, *CDT*)$$

5. Deictic Time

Deictic temporal concepts, such as "now", "today", "tomorrow night", and "last year", are more common in natural language texts than they will be in descriptions of Web resources, and for that reason we are postponing a development of this domain until the first three are in place. But since most of the content on the Web is in natural language, ultimately it will be necessary for this ontology to be developed. It should, as well, mesh well with the annotation standards used in automatic tagging of text.

We expect that the key concept in this area will be a relation *now* between an instant and an utterance or document.

$$now(t, d)$$

The concept of "today" would also be relative to a document, and would be defined as follows:

$$today(T, d) \equiv (\exists\, t, n, x)[now(t, d) \wedge in\text{-}interval(t, T) \wedge da(T, n, x)]$$

That is, T is today with respect to document d if and only if there is an instant t in T that is now with respect to the document and T is a calendar day (and thus the nth calendar day in some interval x).

Present, past and future can be defined in the obvious way in terms of now and before.

Another feature of a treatment of deictic time would be an axiomatization of the concepts of "last", "this", and "next" on anchored sequences of temporal entities.

6. Aggregates of Temporal Entities

A number of common expressions and commonly used properties are properties of sequences of temporal entities. These properties may be properties of all the elements in the sequence, as in "every Wednesday", or they may be properties of parts of the sequence, as in "three times a week" or "an average of once a year". We are also postponing development of this domain until the first three domains are well in hand.

This may be the proper locus of a duration arithmetic, since we may want to know the total time an intermittant process is in operation.

7. Vague Temporal Concepts

In natural language a very important class of temporal expressions are inherently vague. Included in this category are such terms as "soon", "recently", "late", and "a little while". These require an underlying theory of vagueness, and in any case are probably not immediately critical for the Semantic Web. This area will be postponed for a little while.

Acknowledgments

I have profited from discussions with James Allen, George Ferguson, Pat Hayes, Adam Pease, and Stephen Reed, among others, none of whom however would necessarily agree entirely with the way I have characterized the effort. The research was funded by the Defense Advanced Research Projects Agency under Air Force Research Laboratory contract F30602-00-C-0168 and by the Advanced Research and Development Agency.

8. References

Allen, James F. and Henry A. Kautz. 1985. "A model of naive temporal reasoning." *Formal Theories of the Commonsense World*, ed. by Jerry R. Hobbs and Robert C. Moore, Ablex Publishing Corp., pp. 251-268.

Temporal Information in Collateral Texts for Indexing Movies

Andrew Salway and Eleftheria Tomadaki

Department of Computing
University of Surrey
Guildford, Surrey
GU2 7XH
United Kingdom
a.salway@surrey.ac.uk

Abstract

This paper suggests that video indexing is an interesting and important natural language application for which it is crucial to identify temporal information in collateral text that articulates the semantic content of moving images. Recently a rich source of information about the content of films and television programmes has become available in the form of audio description scripts. The analysis of the expression of temporal information in a corpus of audio description scripts leads to a discussion of some consequences for schemes to annotate such information in a video indexing application.

1. Introduction

The further development of digital libraries to retrieve, manipulate, browse and generate complex multimedia artefacts depends upon the machine-based representation of those artefacts, and in particular their 'semantic content'. An image can be understood at different levels of meaning: an image sequence, like a movie, can also tell a story by depicting a sequence of events. A crucial part of a film's semantic content is the narrative that it relates. As the story unfolds, the viewer constructs their understanding of the story guided by the director's careful sequencing of scenes and editing of shots. A machine-level representation of a film should maintain its rich structure and detail the entities, events and themes depicted; but how can a representation be instantiated for a given film?

One general approach to video indexing is based on the association of moving images and *collateral text* so that keywords, and potentially richer representations, are extracted from text fragments. Consider, for example, the speech of news and documentary presenters, sports commentaries and even newspaper film reviews. The challenge is to explicate the relationship between the moving image and the text. This involves dealing with temporal information in various ways; for example it is necessary to associate text fragments with the video intervals for which they are true; temporal relationships between the events depicted in the moving image must be extracted from the text; and, the time at which the action takes place must be ascertained.

A newspaper film review gives an incomplete and temporally re-ordered account of the events in a film. The speech of a newsreader is temporally aligned with the moving image but does not always refer to the visual information directly – much of a news broadcast comprises head and shoulders shots of newsreaders or stock video footage used to illustrate a story, thus keywords are more likely to be indicative of general story content than refer directly to what can be seen. By contrast, an audio description is a kind of 'narrative monologue' that gives a detailed account of what can be seen on screen in which the text order tends to follow the order of events in a programme or film. Audio description enhances the enjoyment of most kinds of films and television programmes for visually impaired viewers. In the gaps between existing speech the audio description gives key information about scenes, people's appearances and on-screen actions so that in effect the story conveyed by the moving image is retold in words.

We are interested in applying information extraction technology to generate machine-level representations of video content from audio description. It is hoped that the enhanced representation of video content could facilitate more complex querying ("find all clips showing X happening" – where X is a detailed description of events) and perhaps also contribute further to systems for video generation and maybe even question answering about what happened in a movie and why. As a first step towards information extraction we are considering the annotation of a corpus of audio description scripts to explicate what and how information is conveyed. At the moment priority is being given to temporal information because it seems to be crucial for the proper integration of moving images and collateral text.

Audio description is scripted before it is recorded. An audio description script is thus a text which is 'written to be spoken' and includes time-codes to indicate when each utterance is to be spoken. The task of processing audio description scripts is constrained because audio describers follow guidelines that restrict the language they use, i.e. normally the present tense, simple sentences and few pronominal references. This restricted language, the presence of time-codes and the relatively straightforward chronological order of the texts make audio description scripts a good starting point for extracting information for video indexing.

Though it is straightforward to associate a time-coded text fragment approximately with a video interval, a more precise association requires consideration of tense and aspect. For example, consider how the following fragments relate to intervals in the moving image: they are from audio description for *The English Patient* – time codes are in the format [minute:second][1].

[1] This sample is reproduced from *The English Patient*. Please note that further examples in the paper are fictitious but based closely on actual audio description and maintain grammatical structure (i.e. only names and events have been changed).

[11:43] Hanna passes Jan some banknotes – a near instantaneous event in the present tense, so the fragment relates to a short video interval at the time of speaking;

[11:55] Laughing, Jan falls back into her seat – the present participle indicates that 'laughing' is ongoing and so relates to a longer video interval that includes the instantaneous 'falls back';

[12:01] An explosion on the road ahead – use of nominalisation to refer to an event;

[12:08] The jeep has hit a mine – the present perfect indicates that the event is completed and the video interval that the text relates to must have start and end points before the time-code of the text fragment (general knowledge tells us that this event occurred before the explosion and was its cause).

Once text fragments have been associated with video intervals the events depicted in the steady flow of the moving image must be related to each other according to a different time-line – that of the diegetic world depicted by the movie. For example the 'hit mine' event happens immediately before the 'explosion' described above and it might be appropriate to label the relationship with causality. There are also examples of simultaneous and included events, such as – *he prevents her from leaving, holding her firmly*. Events in a movie are grouped in scenes where each scene has a (normally) unique combination of time and location. In audio description an explicit time reference might be used to introduce a new scene, e.g. *October 1944*; *later* is also used to introduce scenes and indicate story progression.

This paper suggests that video indexing is an interesting and important natural language application for which it is crucial to identify and analyse temporal information in collateral texts that articulate the semantic content of moving images. A review of video retrieval systems shows that the use of collateral text is important, but in order to extend the approach to more kinds of video material and collateral text it will be essential to process temporal information. The conceptualisation of time and events with respect to semantic content in digital video systems is outlined, particularly for films (Section 2).

We attempt to formalise the challenge of integrating video data and collateral text by describing three tasks that would contribute to the use of collateral text for video indexing. These tasks guided the analysis of an audio description script corpus (70,856 words): prominent expressions of temporal information are quantified and exemplified (Section 3). The results begin to give a basis for discussing what would be required of a scheme to annotate temporal information in this scenario: existing annotation schemes are reviewed and some tentative extensions are proposed (Section 4). The paper closes by considering further directions for this work (Section 5).

2. Digital Video Systems

Video data can be indexed with visual features based on the distribution of pixels, e.g. colour, texture, shape and motion: however a 'semantic gap' appears between video databases and users who often conceive their information needs in terms of the relationships between entities, events and themes to be depicted in the video sequence of interest. Indexing could be achieved by attaching keywords and other descriptors manually to either whole video data files or intervals and regions within them. A cheaper alternative is to use language technology to process 'collateral text'; Srihari introduced this term to refer to textual information associated with visual information, specifically photo captions (1995). Video data sometimes includes an *integral* textual component in the form of speech and closed-captions. Other *external* textual information arises in the production and distribution of video material, e.g. scripts and production notes, and now audio description (legislation in a number of countries makes the provision of audio description mandatory for an increasing amount of digital TV and film output).

In news broadcasts and documentary programmes much of the information content is carried by the spoken words of the presenters, and the subjects on which they are speaking will reflect, albeit to varying degrees, the entities, events and themes shown in the accompanying moving images. The *Informedia* system indexes news broadcasts and documentary programs by keywords that are extracted from speech and closed captions: since the speech is time-aligned with the moving images the keywords can be associated with specific video intervals (Wactlar et al., 1999). This approach has been extended into a multi-lingual context in the Pop-Eye and Olive projects, and to deal with sports footage in the current MUMIS project (de Jong et al., 2000). Other researchers have applied text segmentation techniques to the speech stream of video data in order to segment video sequences (Mani et al., 1997; Takeshita, Inoue and Tanaka, 1997). The transcribed speech of news presenters has been exploited in a system for browsing through news broadcasts by following hypertext links between terms and viewing associated video sequences (Shahraray, 1999). Research and systems focused on accessing broadcast news, including further tasks like multi-stream segmentation, combined name/face recognition and multimedia summarisation are collected in Maybury (2000).

There are moving images that do not contain 'integral' text, but that can be indexed with text that was produced specifically to elucidate the video's content. The *WebSEEK* system, which has indexed hundreds of thousands of images and videos on the WWW, selects keywords from the text of hyperlinks to images and videos on WWW-pages (Smith and Chang, 1997); note that this system only indexes whole videos rather than intervals. Another system, developed at the Japan Broadcasting Corporation, parses the notes kept in the production of wildlife documentary programs that describe the entities and events in the recorded footage and are time-coded. Queries for video intervals can be made in terms of the relationships between entities and actions (Kim and Shibata, 1996).

More recently there have been developments to combine visual and textual features for the classification of video sequences. For example, visual features may indicate the location of a scene (indoors/outdoors) and whether there is one or many people in the shot, and textual features may indicate the nature of the spoken words (a news report / a political speech): taken together these features suggest whether a video sequence depicts a political rally, an outside news broadcast, or a political party's conference (Satoh, Nakamura and Kanade, 1999).

Textual information from TV sit-com scripts has been combined with visual features, through a process involving user interaction, so a system can locate scenes containing a particular character (Wachman and Picard, 2001).

Collateral text could potentially be used for extracting information about other kinds of video, including those with rich semantic content like films and dance sequences. In specialist domains, like dance, there is an extensive range of collateral texts available (dance programmes, newspaper reviews, textbooks, choreographer's notes, biographies, etc.) and spoken commentaries can be elicited from experts asked to 'describe' and to 'interpret' sequences. The KAB system was developed to index fixed-length intervals of dance videos with keywords from such commentaries: this work also specified requirements for a general system to process diverse collateral texts (Salway and Ahmad, 1998; Salway, 1999). A key requirement is a video data model and representation scheme that captures semantic video content, including temporal organisation, at an appropriate level of detail to facilitate complex queries, browsing and even video generation; the link with collateral text also needs to be modelled.

In the video database literature semantic content is usually treated as comprising the objects and events depicted by a moving image and the spatio-temporal relationships that hold between them; for a survey see Chen, Kashyap and Ghafoor (2000). Descriptions of objects and events (as keywords and propositions, for example) are associated with intervals of video data which can be modelled either as a hierarchy of discrete intervals (Weiss, Duda and Gifford, 1995) or as multi-layered overlapping intervals (Davenport, Aguierre Smith and Pincever, 1991).

Allen's (1983) temporal logic has been applied widely in video data models to facilitate reasoning about video content and more complex queries: the number of the 13 possible interval relationships that are used varies between applications. A hierarchical model is appropriate for dealing with film in terms of scenes and shots (see Corridoni et al., 1996). However, to capture more detail about the events within a shot it might be necessary to allow for overlapping video intervals and more description of the relationships between events.

Knowledge representation schemes aim to provide unambiguous representations of meaning and to facilitate inferencing: a number of proposals have been made to use such schemes for semantic video content. Regarding the composition of events/sub-events in moving images, particularly in stereotypical situations, a framework was developed based on Schank and Abelson's scripts (1977), see (Parkes, 1989; Nack and Parkes, 1997). Semantic networks have been used in a video browsing system to elaborate the description of events, for example to specify participants and causal relationships between events (Roth, 1999). The use of conceptual dependency graphs and story grammars has also been discussed (Tanaka, Ariki and Uehara, 1999). Independent to this, but sharing some similar goals, researchers in computer vision have proposed levels to deal with complex visual information at stages from raw visual input to final representation, e.g. 'change – event – verb – history' (Nagel, 1988), and specifically for human motion 'movement – activity – action' (Bobick, 1997).

3. Temporal Information in Collateral Text

This section characterises the expression of temporal information in a corpus of audio description scripts with respect to three tasks we consider important for video indexing with collateral text. First though, in order to extend the use of collateral text to index films it is necessary to explicate how a linear text relates to a film with multi-faceted content. The discussion here is limited to film content that is conveyed visually, and hence accessible through audio description - we are not currently considering dialogue and sound effects. The focus is on films and accompanying audio description but much of what is discussed could be relevant to other kinds of video and other collateral text types.

3.1. Integrating Moving Images and Text

In order to integrate audio description text with film at a semantic level it is necessary to deal with film in terms of the shots and scenes by which it is structured. It is also important to recognise two timelines: (i) film time, i.e. the time it takes to watch the film; and, (ii) story time, i.e. the time in which the events depicted take place. Figure 1 shows how a film (stored as a video data file) can be modelled in terms of shots which are defined as continuous pieces of filming, and scenes which are characterised by each having a unique combination of location and time. The story timeline is shown in parallel with layers of events taking place. Of course the relative position of events may differ between the two timelines, e.g. the film may depict events in a different order than they happen in the story, and events that are happening at the same time but in different locations will be depicted in different scenes. For video retrieval purposes it is important to maintain temporal relationships between events; different sub-sets of Allen's relationships will be required for different applications.

The structure of film provides some useful constraints when dealing with temporal information. It is reasonable to assume that all events depicted within a scene take place close together in the story timeline, and are likely to form larger events (information about scene boundaries may be available from sources like film scripts and automatic video analysis). When considering how events are depicted at the shot level it is important to note film-making techniques that are used to convey that an event is taking place, or has taken place, without showing it in its entirety; a director may choose to portray only the end result of an event and allow the viewer to infer that the event took place.

The collateral text is shown as a series of time points that indicate the time at which the speaker starts the utterance (assuming a temporally aligned collateral text, like an audio description). The three tasks outlined next relate to the extraction of temporal information from collateral text to: (i) associate an utterance with the video interval for which it is true, be it a shot, scene or some other interval; (ii) specify event-event relationships – here we only consider relationships holding within a scene (in film time); (iii) establish the time at which scene is set (in story time).

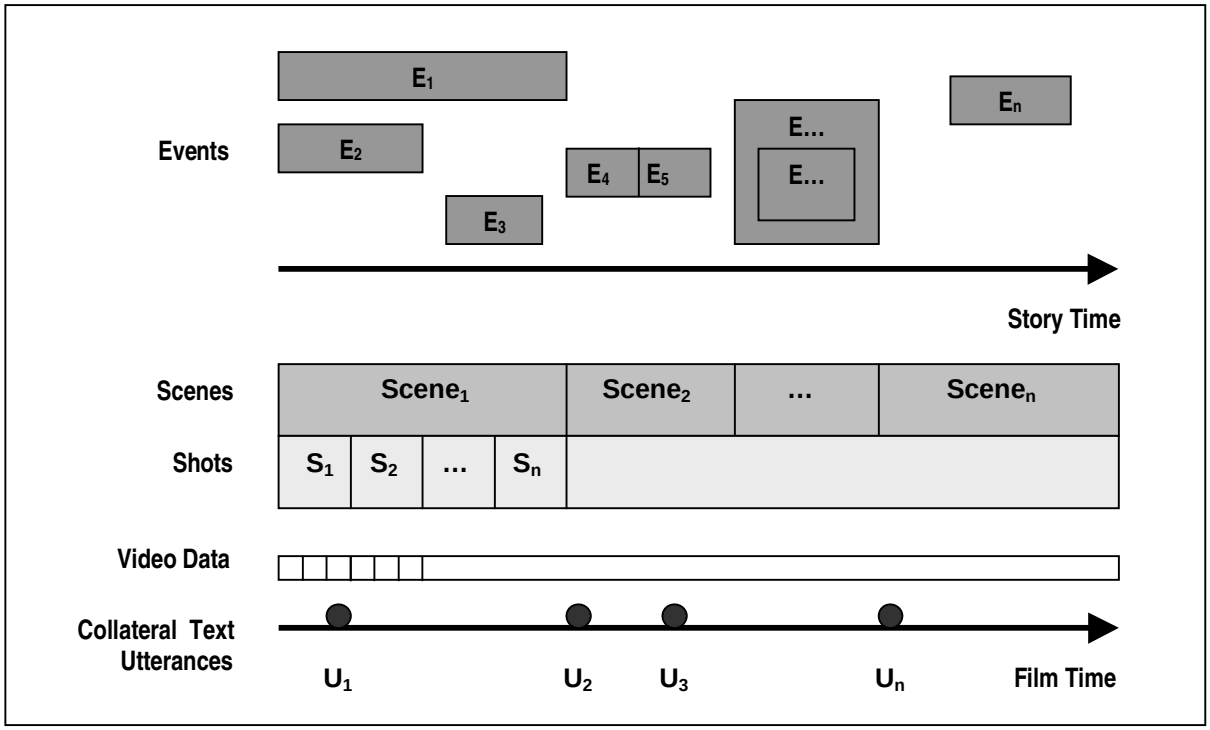


Figure 1. The organisation of a film's content in terms of shots and scenes (which relate to film time) and the events that comprise the semantic video content (which relate to story time). Collateral text such as audio description is temporally aligned with the video data in film time.

3.1.1. Task 1: Associate an audio description fragment with the interval in film time for which it is true.

Given a time coded text fragment it is relatively straightforward to associate it approximately with the video interval for which it is true, i.e. the interval in which the event it refers to is taking place; the time-code plus and minus an arbitrary number of seconds works as a crude approximation of start and end times. However, it is desirable to be more precise about at least one of: start time, end time or duration. (A greater challenge, not addressed here, is to ascertain whether the event is depicted on-screen throughout the duration). As well as events, it is also appropriate to deal with states if they change significantly during the movie, e.g. to indicate scenes in which a character is a child or grown-up.

The problem can be gauged to some extent by considering an earlier feasibility study for indexing moving images with audio description (Turner, 1998). A small sample of video material with accompanying audio description was analysed (including a film and various kinds of television programme). Results showed that overall about 50% of shots were described but only about 40% of the audio description referred directly to the shot on-screen at the time of speaking.

To ascertain appropriate video intervals it may help to consider some of the aspectual features of events classified by Comrie (1976). Whether an event has internal structure (*punctual* / *durative*) gives some information about its duration; this may be an inherent characteristic of a verb but may be modified grammatically, e.g. with the progressive. Knowing about an event's end result, if it has one (*telic* / *atelic*), gives information about its completion (and in audio description may be all that is referred to).

3.1.2. Task 2: Event-event relationships in story time (within the same scene).

Moving images can depict many events at the same time, and in the case of film the temporal organisation of events and relationships such as event / sub-event and causality are crucial to a viewer's understanding. As discussed previously some or all of Allen's 13 temporal relationships might be needed, though whether they can all be extracted from collateral text remains to be seen. In a narrative dialogue by default events are mentioned in the order in which they occur – however, events may occur simultaneously, or there may be stylistic reasons to mention them out of order.

Ascertaining basic temporal relationships, like before / after / overlapping, may be possible just from the collateral text. However, to construct rich representations of composite events within scenes perhaps relies more on prior 'world knowledge' than it does on information immediately available in a narrative monologue (cf. the use of Schank's scripts to deal with semantic video content); the problem becomes harder still when event-event relationships across different scenes are considered. A lexical resource, like WordNet, might help as a first step to relate events, in light of the entailment relations for verbs described by Fellbaum (1998). When considering temporal inclusion some sets of verbs are *co-extensive*, e.g. 'march and walk', 'whisper and talk'; whereas other share a relationship of *proper inclusion*, e.g. 'sleep and snore'. Having access to these relationships may help to

associate descriptions of the same event and to establish sub-event relationships. Other relationships allow events to be associated according to *backward presupposition*, e.g. 'forget and know', and on the basis of *causality*, e.g. 'show and see'.

3.1.3. Task 3: Establish the time a scene takes place (in story time).

A viewer's appreciation of a film requires knowing when it is set, and if it is set over a long time period then the time of each scene must be known – thus information needs to be extracted to give each scene a time. Unless a film is based upon true-life events then it is normally set within a time period without specific dates being implied. Similarly, within the course of a day in story time exact times are usually less important than whether the viewer knows it is morning, afternoon, evening or night (of course exact times will be crucial for some plots). Unless otherwise indicated the assumption is that scenes are ordered sequentially according to the story timeline, but for some movies the use of flashback will have to be dealt with.

3.2. Temporal Information in Collateral Text: a case study with audio description scripts

The intention of this case study is to quantify and exemplify prominent expressions of temporal information in audio description scripts: the analysis is organised around the three tasks for video indexing described in the previous section. The corpus comprises audio description scripts for 12 movies, covering a range of movie genres, and written by six different describers. It currently totals 70,856 words – this will be expanded to around 500,000 words in coming months.

When carrying out the analysis we considered the variety of ways temporal information can be expressed in English as outlined by Quirk et al. (1985), i.e. by using tense, aspect, adverbials, prepositional phrases, subordinate clauses, nouns and proper nouns. Of course in narrative monologues text order is highly informative about the order in which events take place. Our 'conceptualisation of time' is guided by approaches to video data modelling, i.e. the association of event and state descriptions with video intervals, the specification of interval relationships following Allen (1983), and the organisation of complex events using knowledge representation schemes. Theoretical perspectives on events, such as Comrie's classification of aspect (1976) and Fellbaum's entailment relations (1998) were also considered.

Based on the 50 most frequent verbs in the corpus it appears that the majority of events are material processes (84%), with some mental processes (10%), a few relational processes (4%) and a few behavioural processes (2%), following Halliday (1994).

3.2.1. Information to Associate Text Fragments with Video Intervals

The present tense proliferates in the audio description corpus. It is even used to describe events that are about to happen, for example to describe speech acts which cannot be described at the time they occur – *the doctor questions Tom*. The occasional use of the present perfect is important to describe events after they happened (possibly because there was not an opportunity to describe them at the time they occurred, or because only the end result is depicted on screen) – *the cake has been eaten*. Past events are also sometimes referred to in relative clauses used to identify unnamed characters – *the woman who visited Paul is walking down the street*. In order to be more precise about the start, end or durations of events it seems that a variety of aspectual information is important, especially aspectual verbs and the inherent aspect of verbs.

In the audio description corpus the verbs *start, stop, begin* and *finish* occur relatively far more frequently than they do in the general language British National Corpus (BNC), Table 1 shows just 3rd person singular forms. These verbs almost always appeared in the present tense to refer to another event so it would be straightforward to use them to compute their arguments' start and end times.

Verb	Abs. Freq	Ratio with BNC
stops	105	65.79
starts	60	25.13
begins	19	4.67
finishes	3	25.65

Table 1: Showing prominent aspectual verbs in Surrey's audio description corpus (only for 3rd person singular). The third column is calculated by dividing relative frequency in the audio description corpus with relative frequency in the British National Corpus (BNC)

Regarding the duration of events the adverb *still* is frequently used with durative events that have not finished at the time of speaking (85 occurrences of *still* in the corpus; 62 of these are in the time sense). There was little sign of time expressions giving information about exact durations but relatively short periods were frequently indicated with *for a moment* (29 occurrences). Other frequent, adverbs like *slowly* (111 occurrences) and *quickly* (20) might make a small contribution to understanding the duration of an event.

The grammatical marking of progressive aspect does not appear to be significant for the task of associating text fragments with video intervals. In a narrative monologue we learn nothing about the duration of the event from the distinction between 'he runs' and 'he is running'; the simple present and the progressive are used interchangeably in the corpus. In fact it is probably an event's inherent aspect that is most important to determine, at least approximately, its duration in film time. In general language this will be problematic given that the many senses of common verbs often have different aspectual characteristics, however in specialist domains it may be possible to store default durations for events, like dance movements.

3.2.2. Information to Specify Event-Event Relationships in Film Time

The most frequent conjunction to indicate events happening at the same time in the audio description corpus was *as* (350 occurrences in the time sense) – *the children*

play as the crowd moves away; sometimes *as* indicates more of a connection between events – *she continues to hide as the monster approaches*. Used only to indicate simultaneity (without implying further connection between events) *while* occurred 37 times. Both these conjunctions indicate some degree of overlap between events but further information is required to know whether the events are strictly simultaneous, whether one is included within the other, or if they simply overlap. Non-finite verbs with sub-ordinate clauses tend to indicate that the second event is included in the first – *Coughing, Mary gives the medicine to Tom*. When linking events *and* was 'ambiguous' as to whether the events occurred serially or in parallel.

Occurring only in its time sense, *then* (173 occurrences) was still relatively more frequent in the audio description corpus than the BNC. Though it is redundant as far as indicating sequence is concerned (that is already conveyed by text order) it does seem to imply the completion of the first event before the start of the next one – *Sarah chops the tomatoes then fries an egg*. Furthermore in many examples it suggests that the events meet in time, i.e. the endpoint of one equals the start of the other. (This kind of information could be useful in relation to our Task 1). The less frequent *when* (29 occurrences) and *until* (20) were used respectively to indicate the start and end of events in relation to other events or states, often suggesting that the first event led to the second – *she walks through the forest until she finds the house*.

Like *then*, *now* (40 occurrences) adds little or nothing by way of basic temporal information in these narrative monologues, however it does seem to indicate a change or contrast between two events across a passage of audio description – *Jane is dancing with George … Now she is dancing with her cousin*. Perhaps surprisingly, *after* and *before* occur relatively infrequently in the corpus (compared with the BNC) and when they are used they only serve to emphasise the sequence of events already conveyed by text order, i.e. we find 'after E1, E2' but not 'E2 after E1', and 'E1 before E2' but not 'before E2, E1'.

The adverb *again* is prominent in the corpus (141 occurrences – 2.5 times relatively more often than in the BNC). It generally indicates that an event is happening for a second time within a scene – for video retrieval purposes it might be useful to relate the two instances.

3.2.3. Information to Specify When Scenes Take Place in Story Time

The most frequent time expressions used to locate a scene on the story timeline relate to non-specific times of day: *night* (37 occurrences), *morning* (19), *evening* (11), *dusk* (6), *dawn* (3). Less frequent were expressions for non-specific times of year, i.e. months (without years), seasons and festival days (17 occurrences in total). This probably reflects the fact that the progression of time during a film is more often at the granularity of days. The relative paucity of specific times and dates (there were only a few examples) is explained in part by the fact that for many films the viewer need only understand a general time period. This may be conveyed by costumes, props and, for times of day, lighting: these will all be referred to by audio description.

Scenes are sometimes introduced with one of the time expressions mentioned above – indicating a change of time is a shortcut to indicate a new scene. Quite often *later* (32 occurrences) is used for this purpose and as such may be a useful cue for scene segmentation.

4. An Annotation Scheme for Temporal Information in Collateral Text?

Based on the preceding analysis this section discusses some tentative requirements of an annotation scheme for temporal information in collateral text: such a scheme would be a step in applying information extraction technology to the task of video indexing. The extent to which existing schemes would cater for these requirements is reviewed. A number of factors suggest that some extensions to existing schemes will be required: (i) there seems to be a need to maintain two timelines; (ii) if practical in terms of time and inter-annotator agreement, it would be desirable to record aspectual information regarding the internal structure of events and end-states; (iii) also subject to practicality, it is important for film to specify sub-event and causal relationships.

A canonicalized representation of times was proposed as part of a set of guidelines for annotating temporal expressions by Mani et al. (2001), who targeted a variety of text genres such as both print and broadcast news, and meeting scheduling dialogues. The emphasis of their approach was on detailing different classes of time expressions like points in time (when), durations (how long) and frequencies (how often) and handling context-dependent expressions. It also addressed fuzzy temporal boundaries that arise from the use of phrases that refer to times of year and times of the day, e.g. *summer* and *morning*, and addressed non-specific times, such as *a sunny day in April* (not a specific day, nor a specific year).

Of the time expressions dealt with it is points in time that seem to apply most directly in our scenario in order to locate events (at the granularity of scenes) on the story timeline (our Task 3). Though duration and frequency relate to the kinds of aspectual characteristics that we would like to describe for events, they annotate only words and phrases that express this information directly; though there were some frequent phrases in the audio description corpus for which it might be applicable – *for a moment*.

Another scheme that has been proposed is more concerned with associating temporal information with events, and annotating the temporal relationships between events (Setzer and Gaizauskas, 2000; Setzer, 2001); this scheme was developed initially for newswire texts but is extensible. Annotations are attached to the heads of finite verb groups as representatives of events, as well as to temporal expressions. It is possible to specify the type of event (Occurrence / Perception / Reporting / Aspectual) as well as the tense and grammatical aspect of the verb. The annotations have attributes to specify five event-event and event-time relationships: 'before', 'after', 'includes', 'included' and 'simultaneous'. The features of the scheme that have been summarised here are exemplified in Appendix A which shows the annotation of 9 utterances of audio description.

The annotation of event-event relationships within a scene (our Task 2) would be dealt with quite

comprehensively by Setzer's and Gaizauskas' scheme: although as many as 13 temporal relationships (from Allen) are discussed in the video retrieval literature the five used in this scheme would probably serve most purposes. Being able to annotate aspectual events, i.e. to indicate the start and end time of occurrence events, is certainly important given their frequency in the audio description corpus - cf. our Task 1. For other parts of Task 1 it might be necessary to extend the scheme to specify the start and end of events when there is no explicit time expression, or to do it relative to the time-code in the text; a further minor extension would be to allow for the annotation of states as well as events. It certainly would be desirable to be able to specify causal and sub-event relationships between events as these are crucial to the narrative structure of movies, however this would depend upon annotators' ability to apply them consistently.

5. Closing Remarks

Dealing with temporal information is an important first step towards generating machine-level representations of video content from collateral text, especially when dealing with a complex multimedia artefact, like film, and richly informative collateral text, like audio description. This work is in its early stages but the three tasks outlined here begin to give us a handle on some of the challenges involved in integrating moving images and narrative monologues. The corpus analysis showed an extensive range of temporal information that needs to be dealt with in respect to these tasks. Progress will be made by more extensive application of existing annotation schemes leading to decisions about exactly what is required by way of extensions. Such decisions need to be informed by considerations of any new scheme's practicality (is it simple enough to be applied consistently and quickly) and the extent to which it captures important information (the criteria for which will vary between video applications and users). The final test would perhaps be a comparison of video retrieval using: (i) unannotated audio description (i.e. relying on time codes and text order alone); (ii) annotated audio description (with no further processing); and, (iii) machine-based representations generated from annotated audio description.

6. Acknowledgements

This research was carried out as part of the Television in Words project (TIWO) supported by an Engineering and Physical Sciences Research Council (EPSRC) grant, GR/R67194/01. The authors would like to thank the members of the TIWO Round Table for sharing their knowledge of audio description and providing samples for our corpus. Finally, we are very grateful for the comments of two anonymous reviewers and have tried to take heed.

7. References

Allen, J.F., 1983. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM* 26 (11):832-843.

Bobick, Aaron F., 1997. Movement, Activity, and Action: The Role of Knowledge in the Perception of Motion. *Philosophical Transactions of the Royal Society of London Series B – Biological Sciences* 352 (1358):1257-1265.

Chen, S.-C., R.L. Kashyap, and A. Ghafoor, 2000. *Semantic models for Multimedia Database Searching and Browsing*. Kluwer Academic Publishers.

Comrie, B., 1976. *Aspect: an introduction to the study of verbal aspect and related problems*. Cambridge University Press.

Corridoni, J.M., A. Del Bimbo, D. Lucarella, and H. Wenxue, 1996. Multi-perspective Navigation of Movies. *Journal of Visual Languages and Computing*, 7:445-466.

Davenport, G., T. Aguierre Smith, and N. Pincever, 1991. Cinematic Primitives for Multimedia. *IEEE Computer Graphics and Applications* July:67-74.

de Jong, F., J.-L. Gauvain, D. Hiemstra, and K. Netter, 2000. Language-Based Multimedia Information Retrieval. *Proceedings RIAO 2000 Content-Based Multimedia Information Access, Paris, April 2000*, 713-722.

Fellbaum, C., 1998. A Semantic Network of English Verbs. In Fellbaum (eds.), *WordNet: an electronic lexical database*. Cambridge MA: The MIT Press.

Halliday, M. A. K., 1994. *An Introduction to Functional Grammar*. London: Edward Arnold, 2nd edition.

Kim Y.-B., and M. Shibata, 1996. Content-Based Video Indexing and Retrieval – A Natural Language Approach. *IEICE Transactions on Information and Systems* E79-D(6):695-705.

Mani I., D. House, M.T. Maybury, and M. Green, 1997. Towards Content-Based Browsing of Broadcast News Video. In M. Maybury (ed.), *Intelligent Multimedia Information Retrieval*. Menlo Park CA / Cambridge MA: AAAI Press / MIT Press.

Mani, I., G. Wilson, L. Ferro, and B. Sundheim, 2001. Guidelines for Annotating Temporal Information. *Procs. HLT 2001, First International Conference on Human Language Technology Research*, San Francisco: Morgan Kaufmann.

Maybury, Mark, 2000 (ed.). Special Issue – 'News on Demand'. *Communications of the ACM*, 43(2).

Nack, F., and A. Parkes, 1997. Toward the Automated Editing of Theme-Oriented Video Sequences. *Applied Artificial Intelligence*, 11:331-366.

Nagel, H.-H., 1988. From image sequences towards conceptual descriptions. *Image and Vision Computing*, 6(2):59-74.

Parkes, A. P., 1989. The Prototype CLORIS System: Describing, Retrieving and Discussing Videodisc Stills and Sequences. *Info. Proc. and Management*, 25(2):171-186.

Quirk, R., S. Greenbaum, G. Leech, and J. Svartvik, 1985. *A Comprehensive Grammar of the English Language*. London and New York: Longman.

Roth, V., 1999. Content-based retrieval from digital video. *Image and Vision Computing*, 17:531-540.

Salway, A., 1999. *Video Annotation: the role of specialist text*. Ph.D. thesis, University of Surrey.

Salway, A., and K. Ahmad, 1998. Talking Pictures: Indexing and Representing Video with Collateral Texts. *Procs. 14th Workshop on Language Technology* -

Language Technology for Multimedia Information Retrieval, 85-94.
Satoh, S., Y. Nakamura, and T. Kanade, 1999. Name-it: Naming and detecting faces in news videos. *IEEE Multimedia*, 6 (1):22-35.
Schank, R. C. and R. P. Abelson, 1977. *Scripts, Plans, Goals and Understanding: an inquiry into human knowledge structures*. Hillsdale, NJ: Lawrence Erlbaum Associates.
Setzer, A., 2001. *Temporal Information in Newswire Articles: An Annotation Scheme and Corpus Study*. Ph.D. thesis, University of Sheffield.
Setzer, A., and R. Gaizauskas, 2000. Annotating Events and Temporal Information in Newswire Texts. *Procs. LREC 2000, 2nd Int. Conf. On Language Resources and Evaluation*, 1287-1293.
Shahraray, B., 1999. Multimedia Information Retrieval Using Pictorial Transcripts. In B. Furht (ed.), *Handbook of Multimedia Computing*. Florida: CRC Press.
Smith J.R., and S.-F. Chang, 1997. Visually Searching the Web for Content. *IEEE Multimedia* July-Sept:12-20.
Srihari, R.K., 1995. Computational Models for Integrating Linguistic and Visual Information: A Survey. *Artificial Intelligence Review*, 8(5-6):349-369.
Takeshita, A., T. Inoue, and K. Tanaka, 1997. Topic-based Multimedia Structuring. In M. Maybury (ed.), *Intelligent Multimedia Information Retrieval*. Menlo Park CA / Cambridge MA: AAAI Press / MIT Press.
Tanaka, K., Y. Ariki, and K. Uehara, 1999. Organization and Retrieval of Video Data. *IEICE Trans. on Information and Systems*, E82-D(1):34-44.
Turner, J.M., 1998. Some Characteristics of Audio Description and the Corresponding Moving Image. *ASIS Annual Meeting*, 35:108-117.
Wachman, J. S., and R.W. Picard, 2001. Tools for Browsing a TV Situation Comedy Based on Content Specific Attributes. *Multimedia Tools and Applications*, 13(3):255-284.
Wactlar, H.D., M.G. Christel, and Y. Gong, and A.G. Hauptmann, 1999. Lessons Learned from Building a Terabyte Digital Video Library. *Computer*, Feb:66-73.
Weiss, R., A. Duda, and D.K. Gifford, 1995. Composition and Search with a Video Algebra. *IEEE Multimedia*, Spring 1995:12-25.

Appendix A

Annotation of an audio description script following Setzer's scheme

The following passage of audio description (from *The English Patient*) has been annotated following the scheme and guidelines given by Setzer (2001). The sample here exemplifies how: (i) tense and aspect features can be associated with an event's verb; (ii) how the class of an event can be noted; (iii) how relationships between events can be specified. The sequence of events inherent in the text order has not been annotated, though it could have been – only exceptions to the 'default' have been annotated, e.g. simultaneous events, and events that are mentioned in a different order to which they occur.

[11.43] Hanna <event eid=1 tense=present class=occurrence> passes </event> Jan some banknotes.
[11.55] <event eid=2 tense=present class=occurrence aspect=progressive relatedToEvent=3 eventRelType=includes> Laughing </event>, Jan <event eid=3 tense=present class=occurrence relatedToEvent=4 eventRelType=simultaneous signalID=1> falls </event> back into her seat <signal sid=1> as </signal> the jeep <event eid=4 tense=present class=occurrence> overtakes </event> the line of the lorries.
[12.01] An <event eid=5 tense=present class=occurrence relatedToEvent=6 eventRelType=after> explosion </event> on the road ahead.
[12.08] The jeep has <event eid=6 tense=present class=occurrence aspect=perfective > hit </event> a mine.
[12.09] Hanna <event eid=7 tense=present class=occurrence> jumps </event> from the lorry.
[12.20] Desperately she <event eid=8 tense=present class=occurrence> runs </event> towards the mangled jeep.
[12.27] Soldiers <event eid=9 tense=present class=occurrence> try </event> to stop her.
[12.31] She <event eid=10 tense=present class=occurrence> struggles </event> with the soldier who <event eid=11 tense=present class=occurrence> grabs </event> hold of her firmly.
[12.35] He <event eid=12 tense=present class=occurrence> lifts </event> her bodily from the ground <event eid=13 tense=present class=occurrence aspect=progressive relatedToEvent=12 eventRelType=simultaneous> holding </event> her tightly in his arms.

Recognizing and Tagging Temporal Expressions in Spanish

Estela Saquete, Patricio Martínez-Barco and Rafael Muñoz
Departamento de Lenguajes y Sistemas Informáticos. Universidad de Alicante
Estela.Saquete@ua.es
{patricio, rafael}@dlsi.ua.es

Abstract

This paper shows a system about the recognition of temporal expressions in Spanish and the resolution of their temporal reference. For the identification and recognition of temporal expressions we have based on a temporal expression grammar and for the resolution on an inference engine, where we have the information necessary to do the date operation based on the recognized expressions. For further information treatment, the output is proposed by means of XML tags in order to add standard information of the resolution obtained. Different kinds of annotation of temporal expressions are explained in another articles [WILSON2001][KATZ2001]. In the evaluation of our proposal we have obtained successful results.

1. Introduction

The study of anaphora phenomena has been carried out for a lot of researches. Most of these researches have focused on pronominal anaphora and a few of them on definite descriptions. Most of temporal expressions could be considered as a type of definite description, but a few of them are temporal adverbs like "mañana" (*tomorrow*). The research work developed in definite description is focused on establishing a relationship between anaphoric expressions and their antecedents. In these work, if the definite description is a temporal expression it has been only solved establishing the relationship but not inferring the new date. The resolution of temporal expressions involves the recognition of them and the inference of the new date.

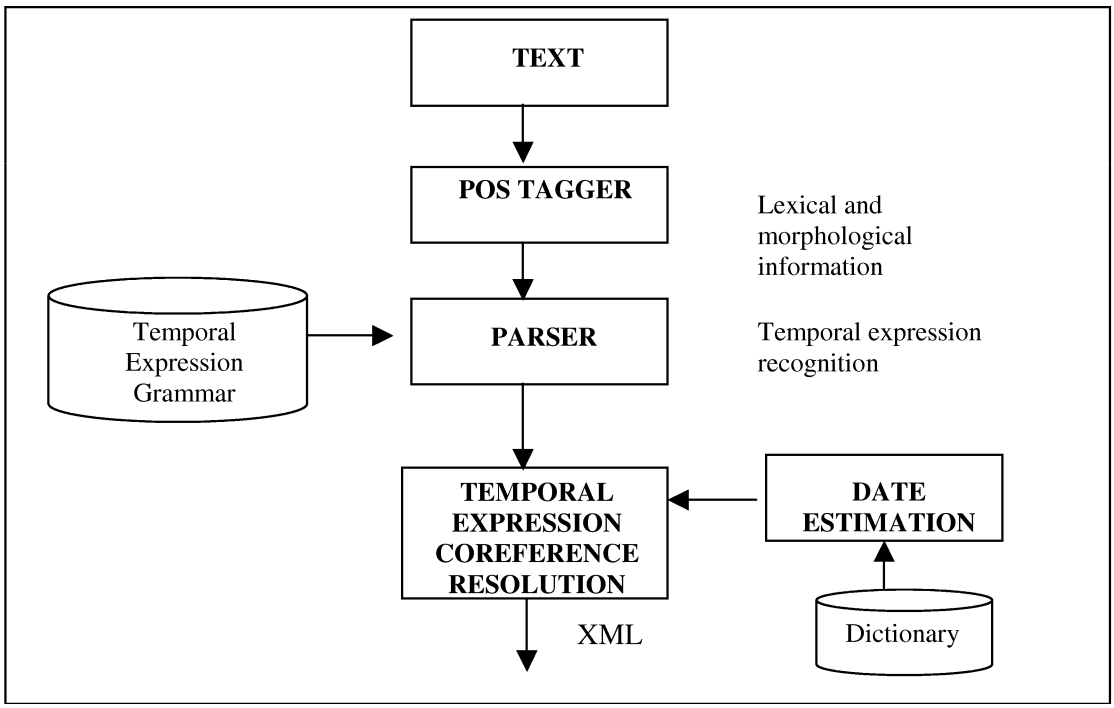


Fig. 1. Graphic representation of the system proposed

Work focused on temporal expression should to solve both tasks. In the literature we can found several studies focused on temporal expressions (Guillen et *al.* 1995) (Wiebe et *al.* 1998). These studies are based on the use of a temporal model that is able to interpret different formats for date and time expressions. Some of them are based on the application of empirical methods using the focus theory proposed in (Grosz et al. 1986).

In this paper a proposal of a grammar for the recognition of temporal expressions in Spanish is presented, as well as an approximation to the resolution of the coreference introduced by them, as is explained in (Saquete and Martínez-Barco 2000). Moreover, in this paper a set of tags is used to annotate the temporal expressions in the corpus.

In a text, there are dates with typical representations like, for example: "23/01/2000" o "23 de enero del 2000" (*23rd of January of 2000*), but we can find references to dates named previously too, for example: "dos días antes" (*two days before*), "la semana anterior" (*the previous week*), etc. This kind of coreference should be solved and mapped to dates with a standard format for a more efficient analysis of the text. For that, we use a grammar for the recognition of temporal expressions with their correspondent temporal parser, and an inference engine to solve and to map these expressions in a standard format: mm/dd/yyyy (hh:mm, for time expressions). Once the kind of reference and the interpretation of the expression are solved, the text is tagged with XML tags.

2. System structure

The system proposed is shown in Figure 1. The system has the plain text as input. These texts are tagged with lexical and morphological information using the POS-tagger developed by Pla (Pla 2000) and this information is the input of the temporal parser. The temporal parser is implemented using an ascending technique and it is based in a temporal grammar shown below. Once the temporal expressions are recognized, these are introduced into the resolution unit called Temporal Expression Coreference Resolution, which will update the value of the reference according to the date that it is referring to, and then it will generate the XML tags for each expression.

There are two different kinds of rules that are used for the grammar because there are two different kinds of temporal expressions too:

1. There are anaphoric and not anaphoric expressions. That is why there are rules for the date and time recognition (non-anaphoric expressions like "12/06/1975").

2. There are rules for the temporal reference recognition (anaphoric temporal expressions that need another complete temporal expression to be understood "*two days before*").

Temporal references could be divided in two groups: time adverbs (i.e. *yesterday, today*) and nominal phrases that refer to temporal relationships (i.e. *two years before*).

Tables 1 and 2 show some rules used for the recognition of dates and the detection of anaphoric temporal expressions, respectively.

3. Coreference resolution based on a temporal model

Previous to coreference resolution, the parser identifies temporal expressions. Temporal expressions can be anaphoric or non-anaphoric. For this reason, we have split the rules for the identification or recognition of temporal expression in two different sets. The first set is made up by the rules for the identification of non-anaphoric temporal expressions (table 1) and the second one is made up by the rule to recognize the anaphoric temporal expressions (table 2). The coreference module should be applied to the temporal expressions recognized by the rules of the second set.

For the coreference resolution we use an inference engine that contains the interpretation for every reference named before. If we compare this system with traditional anaphoric systems, the algorithm for the treatment of temporal expressions needs to carry out an additional step. In our algorithm to solve the coreference of anaphoric temporal expressions, two different tasks can be distinguished:

1. Looking for the antecedent. This task is similar to the traditional approach to anaphora resolution. The algorithm chooses the antecedent from a list of candidates. Two main candidates are usually chosen. The first candidate is related to the date of the text, i.e. the date when the newspaper was written. This date is considered as *default date,* called in this paper *FechaP*. The second candidate is the previous non-anaphoric temporal expressions, called in this paper *FechaAnt* (*two days before* has as antecedent the previous date in the text). If none is found the default date is considered as antecedent. Sometimes, the temporal expression includes prepositional phrase with information about and event or process (*the day of the final match*), in this case the algorithm should look for the date associated to the event or process from the list of candidates. The following process is carried out:

    3. By default, the newspaper's date is used as a base referent (temporal expression) if it exits. If not, the system date is used. ("ayer" *(yesterday)* Day(FechaP) –1 / Month(FechaP) / Year(FechaP).

    4. In case of finding a non-anaphoric temporal expression, it is stored as FechaAnt storing the old FechaAnt in a list of candidates. This value is updated every time that a non-anaphoric temporal expression appears in the text.

2. Providing the new date. Once the antecedent is selected, the new date should be inferred. This new step is related to provide a new date or time. The references are estimated using the antecedent selected in the previous step. This model is based on the two rules below and it is only applicable to these dates that are not *FechaP*, since for *FechaP* there is nothing to resolve

In Table 3 some of the entries of the dictionary used in the inference engine are shown. Moreover, the inference engine has the correspondence between numeric and string expressions of days and months, that is, *one* have the value *1* and *July* is *07*.

The module that makes the estimation of the dates will accede to the right entry in the inference engine in each case and it will apply the function specified obtaining a date in the format *mm/dd/yyyy or* a range of dates. So, at that point the coreference will have been resolved.

date dd + ” /” + mm + “/” + (yy)yy (12/06/1975) *(06/12/1975)*
date dd + “-“ + month + “ -“+ (yy)yy (12-junio-1975) *(12th-June-1975)*
date dd + “de” + mm + “de” + (yy)yy (12 de junio de 1975) *(12th of June of 1975)*
date (“El”) + day_of_week+ dd + ”de”+ month + “de” + (yy)yy (El domingo 12 de junio de 1975) *(Sunday, 12th of June of 1975)*
date month+ ”de”+ yy(yy) (Febrero de 1975) *(February of 1975)*
date dd + ”de”+ month + “de” + (yy)yy + “a las”+ time (12 de junio de 1975 a las 6 y media) *(12th of June of 1975 at half past six)*
dd [“01”|”02”|”03”|….|”31”]
day [“uno”| ”dos” |… |”treinta y uno”] *[one|two|…|thirty one]*
mm [“01”|”02”|”03”|…|”12”]
month [“enero”| “febrero”| “marzo”	“abril”	“mayo”	“junio”	“julio”	“agosto”	“septiembre”	“octubre”	“noviembre”	“diciembre”] *(January|February|March|April|May|June|July|August|September|October|November|December)*
a [“1”	“2”	“3”	…”9”	“0”]
day_of_week [“lunes”	“martes”	“miércoles”	“jueves”	“viernes”	“sábado”	“domingo”] *(Monday|Tuesday|Wednesday|Thursday|Friday|Saturday|Sunday)*
time [hh:mm(:ss)	hh (y	menos) cuarto	hh y media	...]

Table 1. Sample of rules for the date recognition

Time Adverbs	reference “ayer” *(yesterday)* reference “mañana” *(tomorrow)* reference “anteayer” *(the day before yesterday)* reference “anoche” *(last night)*
Temporal Nominal Phrases	reference ”el”+ “próximo” + [“día”	“mes”	“año”] *(the next day|month|year)* reference “un” + [“día”	“mes”	“año”] + “después” *(a day|month|year later)* reference num + [“días”	“meses”	“años”] + “después” *(num days|months|years later)* reference “un” + [“día”	“mes”	“año”] + “antes” *(a day|month|year before)* reference num + [“días”	“meses”	“años”] + “antes” *(num days|months|years before)* reference “dentro” + “de” + “un” + [“día”	“mes”	año”] *(within a day|month|year)* reference “dentro” + “de” + num +[“días”|“meses”| “años”] *(within num days|months|years)* reference “el” + “pasado” + [“día”	“mes”	“año”] *(the last day|month|year)* reference “el” + [“día”	“mes”	“año”] + “siguiente” *(the next day|month|year)* reference “los” + num + [“días”	“meses”	“años”] + “siguientes” *(the num next days|months|years)* reference “el” + [“día”	“mes”	“año”] + “pasado” *(the last day|month|year)* reference “los” + num + [“días”	“meses”	“años”] +“pasados” *(the last num days|months|years)* num [“dos”	“tres”	“cuatro”	“cinco”	….] *(two|three|four|five|…)*

Table 2. Sample of rules for the reference recognition

REFERENCE	DICCIONARY ENTRY
"ayer" *(yesterday)*	Day(FechaP) -1 / Month(FechaP) / Year(FechaP)
"mañana" *(tomorrow)*	Day(FechaP) +1 / Month(FechaP) / Year(FechaP)
"anteayer" *(the day before yesterdary)*	Day(FechaP) -2 / Month(FechaP) / Year(FechaP)
"anoche" *(last night)*	Day(FechaP) -1 / Month(FechaP) / Year(FechaP) [09:00-05:00]
"el"+ "próximo"+"día" *(the next day)*	Day(FechaP)+1 / Month(FechaP) / Year(FechaP)
"un"+"mes"+"después" *(a month later)*	[DayI/Month(fechaAnterior)+1/Year(fechaAnterior)-- DayF/Month(fechaAnterior) +1/ Year(fechaAnterior)]
num+"años"+ "después" *(num years later)*	[01/01/ Year(fechaAnterior) + num -- 31/12/ Year(fechaAnterior) + num]
"un" + "día" + "antes" *(a day before)*	Day(fechaAnterior)- 1/Month(fechaAnterior)/Year(fechaAnterior)
num+"meses"+ "antes" *(num months before)*	[DayI/Month(fechaAnterior) -num / Year(fechaAnterior) - DayF/ Month(fechaAnterior) - num / Year(fechaAnterior)]
"dentro"+"de"+"un"+ "año" *(within a year)*	[01/01/ Year(fechaAnterior) +1 - 31/12/ Year(fechaAnterior) +1]
"dentro"+"de"+num+ "días" *(within num days)*	Day(fechaAnterior)+num / Month(fechaAnterior) / Year(fechaAnterior)
"el" + "pasado" + "día" *(the last day)*	Day(fechaAnterior)-1/Month(fechaAnterior) / Year(fechaAnterior)
"el"+"mes"+"siguiente" *(the next month)*	[DayI / Month(fechaAnterior) +1 / Year(fechaAnterior) -- DayF / Month(fechaAnterior) +1 / Year(fechaAnterior)]
"los"+num+"años"+ "siguientes" *(the num years later)*	[01/01/Year(fechaAnterior) -- 31/12 / Year(fechaAnterior) +num]
"el" + "día" + "pasado" *(the last day)*	Day(fechaAnterior)-1/Month(fechaAnterior) / Year(fechaAnterior)
"los"+num+"meses"+ "pasados" *(the num last months)*	[DayI/Month(fechaAnterior) - num / Year(fechaAnterior) - DayF/Month(fechaAnterior) - 1 / Year(fechaAnterior)]

Table 3. Sample of some of the entries of the dictionary

4. Tagging of temporal expressions

Several proposals for the annotation of temporal expressions have been arisen in the last few years (Wilson et *al.* 2001) (Katz and Arosio 2001) since this kind of research has started. In this section, we proposed doing this annotation using XML tags, in order to standardize anaphoric and non-anaphoric temporal expressions.

4.1. XML

In our proposal we have chosen XML to define the set of tags we are going to use. XML stands for *eXtensible Markup Language* and it provides a subset of the SGML (*Standard Generalized Markup Language*). XML offers a non-ambiguous text-based method to develop data structures. XML documents represent data by means of tags.

XML was developed by a Generic SGML Editorial Review Board formed under the auspices of the W3 Consortium in 1996 and chaired by Jon Bosak of Sun Microsystems, with the participation of a Generic SGML Working Group also organized by the W3C (W3C 2002). Since then, the use of XML has been generalizing until becoming the universal standard for data electronic exchange.

XML has specific rules that must be strictly followed in order to make a new document. The XML document must fulfill the set of constraints being established in the *Document Type Declaration* (DTD). This DTD contains the structure of the document, and the validity of a XML document could be tested through this DTD: a well-formed document is valid only if it contains a proper document type declaration and if the document obeys the constraints of that declaration (element sequence and nesting is valid, required attributes are provided, attribute values are of the correct type, etc.).

As a result, XML provides us several advantages to our proposal:

- Both persons and machines easily interpret it. As a consequence that makes easy both the manual and the automatic extraction of dates from a text.
- XML documents are easily tagged from an automatic process, but also a manual annotator could make use of commercial XML editors to develop this task.
- XML is standard.
- A DTD has been built to check the validity of each XML document. So manual and automatic annotations can be automatically tested looking for possible mistakes.

4.2. Annotation schema

An appropriate annotation schema has been defined to mark every temporal expression found. This schema is base on the following ideas:

First, every temporal expression is going to be marked. That includes the markup of dates and times that could be expressed in whatever format, including anaphoric and not anaphoric expression. In this way, the following rules are going to be applied:

a) Full date expressions, that is, non-anaphoric temporal expressions are going to be marked using the standard format for dates: mm/dd/yyyy.

10 de abril de 2002 *04/01/2002*
(*10th of April of 2002*)

b) Full date and time expressions (again, non-anaphoric expressions) are going to be marked in the same way, including in this case the time parameter, in the standard 24-hour format: hh:mm

10 de abril de 2002, a las nueve
04/10/2002, 09:00
(*10th of April of 2002, at nine o'clock*)

c) Time expressions (without explicit date, but referring to an omitted date) are anaphoric expressions. Then, the coreference resolution module is applied before tagging. Once the date of the time is calculated, date and time are tagged.

A las nueve *04/10/2002, 09:00*
(*At nine o'clock*)

d) Some kind of time expressions without explicit date are not anaphoric expressions because they do not refer to an omitted date. In this case only the time parameter is need, so the coreference resolution module is not used.

Todos los días a las nueve *09:00*
(*Everyday at nine o'clock*)

e) Anaphoric date expressions need the coreference resolution module to define the absolute expressions to which they refer to. After that, the appropriate tag will be marked.

El próximo miércoles *04/17/2002*
(*Next Wednesday*)

f) Anaphoric date and time expressions follow the previous rule, calling the coreference resolution previous to be marked.

El próximo miércoles, a las nueve
04/17/2002, 9:00
(*Next Wednesday, at nine o'clock*)

g) Non-anaphoric ranges of dates and/or time are directly tagged by means of the initial and the final date.

Del 10 de abril al 20 de abril de 2002
04/10/2002 — 04/20/2002
(*From 10 of April to 20 of April of 2002*)[1]

El 10 de abril, de 9 a 11 y media
04/10/2002, 9:00 — 11:30
(*10th of April from 9 to half past 11*)

Del 10 de abril de 2002 a las nueve al 20 de abril de 2002 a las doce
04/10/2002, 9:00 — 04/20/2002, 12:00
(*From 10 of April of 2002 at nine o'clock to 20 of April of 2002 at twelve o'clock)*

h) Anaphoric ranges of dates and/or time are previously solved using the coreference resolution module. Then the full date is marked.

[1] Direct translation from Spanish

Del miércoles al jueves 04/17/2002 — 04/18/2002
(From Wednesday to Thursday)

i) What we have called *fuzzy temporal expressions* have not a concrete date or time related to. For this reason the coreference resolution module is useless. However, in order to be identified as temporal expressions we decided to mark them using a special parameter with the "FUZZY" value.

The grammar used to identify anaphoric and not anaphoric expressions acts as a trigger launching the appropriate rule in each case.

4.3. Tag definitions

The structure of tags used to define temporal expression data is the following:

```
< DATE_TIME TYPE="value"
       VALDATE1="value"
       VALTIME1="value"
       VALDATE2="value"
       VALTIME2="value" >
       Expression
</DATE_TIME>
```

In this structure the next elements are used:

- DATE_TIME is the name of the tag for non-anaphoric temporal expressions.
- VALDATE# store the range of dates obtained from the inference engine.
- VALTIME# store the range of times obtained from the inference engine.
- TYPE attribute could have the following values: CONCRETE, PERIOD and FUZZY:
  - § CONCRETE is referring to only a date.
  - § PERIOD is referring to a period of time.
  - § FUZZY attribute is used when we really do not know the date or period of time when a temporal expression is referring to.

Moreover, VALDATE1, VALDATE2, VALTIME1 and VALTIME2 are optional attributes:

- VALDATE2 and VALTIME2 are used to establish ranges. So, if we try to tag a concrete date (TYPE adopt the value CONCRETE) then these attributes are omitted.
- VALTIME1 could be omitted if only a date is specified.
- VALDATE1 could be omitted if only a time must be specified. This is the case in which the date does not mind. For example, *todos los días a las nueve* (*everyday at nine o'clock*). However, when only a time expression is specified, such as *a las nueve* (*at nine o'clock*), the VALDATE1 of this time must be computed.

The use of XML allows us to take advantage of the XML schema in which the tag language is defined. This schema let an application know if the XML file is valid and well-formed. The schema defines the different kind of elements, attributes and entities that are allowed, and can express some limitations to combine them. Moreover, use the same syntax as XML and the schemas are extensible. Once the XML file has been generated, a parser of our XML needs to be defined to make the information useful.

Tables 4a and 4b show several examples for tagging temporal expressions (non-anaphoric and anaphoric). Table 5 shows an example of an annotated text in which the features of the used tags are shown. In this example we assume that the newspaper's date is 04/25/2000. The system, for the reference "el próximo año"*(the next year),* will return "01/01/2001-12/31/2001". For the reference "mañana" *(tomorrow)* it will return 04/26/2000.

ABSOLUTE TAGS (non-anaphoric temporal expressions)
<DATE_TIME VALDATE1="06/12/2001">12 de junio de 2001</DATE_TIME>
<DATE_TIME VALDATE1="06/12/2001" VALTIME1="20:30">12 de junio de 2001 a las ocho y media de la tarde</DATE_TIME>
<DATE_TIME VALDATE1="06/12/2001" VALTIME1="20:00" VALDATE2="06/12/2001" VALTIME2="21:00" >12 de junio de 2001 entre las ocho y las nueve de la tarde</DATE_TIME>

Table 4a. Sample of the tags generated by this system

REFERENCE TAGS (anaphoric temporal expressions)
<DATE_TIME_REF VALDATE1="06/11/2002">ayer</DATE_TIME_REF> <DATE_TIME_REF VALDATE1="01/01/2002" VALDATE2="12/31/2007">los 5 años siguientes</DATE_TIME_REF>

Table 4b. Sample of the tags generated by this system

"La oficina de Congresos de la Universidad ha propuesto 5 congresos para **<DATE_TIME_REF TYPE= "PERIOD" VALDATE1="01/01/2000" VALDATE2= "12/31/2000>** este año**</DATE_TIME_REF>**, sin embargo, el crecimiento para **<DATE_TIME_REF TYPE="PERIOD" VALDATE1= "01/01/2001" VALDATE2="12/31/2001>** el próximo año**</DATE_TIME_REF>** será superior a los 15. Por otro lado, el Director de la oficina ofrece **<DATE_TIME_REF TYPE= "CONCRETE" VALDATE1= "04/26/2000" >** mañana**</DATE_TIME_REF>** una conferencia."

(The University Conference Office has proposed 5 conferences for **<DATE_TIME_REF TYPE="PERIOD" VALDATE1="01/01/2000" VALDATE2="12/31/2000>** *this year* **</DATE_TIME_REF>**, *however, the increase for* **<DATE_TIME_REF TYPE="PERIOD" VALDATE1= "01/01/2001" VALDATE2="12/31/2001>** *the next year* **</DATE_TIME_REF>** *will be over 15. On the other hand, the Office Manager offers* **<DATE_TIME_REF TYPE= "CONCRETE" VALDATE1= " 04/26/2000" >** *tomorrow* **</DATE_TIME_REF>** *a lecture).*

Table 5. Example of annotated text

5. System evaluation

For implementing the system we need two different units, it is necessary to implement a parser and for that we have used LPA Prolog, because this language is based on rules as the parser is. The other unit is implemented in Visual Basic because this language has several time functions. This unit generated the XML tags too. The implementation of a XML parser is an optional possibility. The evaluation of the system has been done with a sample extracted from 16 articles that belong to the digital edition on the Internet of two Spanish newspapers describing different topics. The results obtained for the articles showed a precision and a recall of 95.59 % and 82.28% respectively.

The total has been calculated according to the number of successes being 195, the number of treated references is 204 and the number of total references is 237.

5.1. Error analysis

However, some fails in the system have been detected and we show their possible improvements below:

- The unit that resolves temporal references is not able to resolve undetermined temporal references like "hace unos cuantos días" (*some days before*) accurately. Here, one possible solution is the use of the semantic information. For example, if the sentence is "*some days before*", the system will suppose that is less time than a week, because we usually use the word "*week*" referring to seven days.

- It is possible that we have non-anaphoric expressions that make reference to an event or a fact and, despite they are not temporal expressions themselves, they mean a date or period of time too. For example: "ganó el mundial y al día siguiente se lesionó" (*he won the World Champion and the next day he hurt himself*).

- In newspaper articles, sometimes we find expressions like "el sábado hubo un accidente" (*Saturday there was an accident*). To resolve this expression we should know some extra information of the context where the reference is. This extra information could be the sentence verb. If it is a past verb that means that the sentence is

referring to *the last Saturday*. However, if the verb is future, it is referring to *the next Saturday*. In our system, we are not using this kind of information, so we assume that this kind of reference if referring to the last day, not the next, because the news usually tells us facts occurred previously.

6. Conclusions

In this paper a system for temporal expressions recognition in Spanish and their reference resolution has been presented, based on a temporal model proposed. The system has two different units: the parser based on a temporal expression grammar, which allows to identify these kind of expressions and a coreference resolution unit which is based in a inference engine and make a transformation of the expressions to dates, resolving their reference in this way. The evaluation of the system shows successful results of precision and recall for our proposal.

For future works, it is pretended to extend the system with the temporal references that are not treated in this paper. Moreover, the study of the verbal forms in the sentences where the references are found will improve the efficiency of the system solving some kind of expressions.

7. Acknowledgements

This paper has been supported by the Spanish Government (MCYT) under grant TIC2000-0664-C02-01/02.

8. References

Katz, G. and Arosio, F. (2001). The Annotation of Temporal Information in Natural Language Sentences. In *Proceedings of the Workshop On Temporal And Spatial Information Processing (ACL'2001).*

Pla, F. (2000). Etiquetado Léxico y Análisis Sintáctico Superficial basado en Modelos Estadísticos. Ph D. Thesis. Departamento de Sistemas Informáticos y Computación. Universidad de Politécnica de Valencia.

Saquete, E. & Martinez-Barco, P (2000). Grammar specification for the recognition of temporal expressions. In *Proceedings of the*

Wilson, G., Sundheim, B., & Ferro, L. (2001). A Multilingual Approach to Annotating and Extracting Temporal Information. In *Proceedings of the Workshop On Temporal And Spatial Information Processing (ACL'2001).*

W3C (2002). Extensive Markup Language web page. W3C World Wide Web Consortium. http:// www.w3.org/XML.

On the Importance of Annotating Event-Event Temporal Relations in Text

Andrea Setzer and Robert Gaizauskas

Department of Computer Science
University of Sheffield
Regent Court
211 Portobello Street
Sheffield S1 4DP, UK
{A.Setzer, R.Gaizauskas}@dcs.shef.ac.uk

Abstract

Many natural language processing applications, such as information extraction, question answering, topic detection and tracking, and multi-document summarisation, would benefit significantly from the ability to accurately position reported events in time, either relatively with respect to other events or absolutely with respect to calendrical time. However, only recently has concerted work started on the automatic extraction of temporal information from text. The overall aim of our work is to automatically establish the temporal relations holding between events as well as between events and calendrical times in newspaper articles. This information makes it possible to create a 'time-event graph' to represent the temporal information contained in a text, and would in turn support the applications mentioned above. In this paper we first argue for the superiority of the time-event graph over a time-stamped event sequence as a target representation for extracted temporal information and discuss the importance of annotating temporal relations. We then give a brief account of the annotation scheme we have devised which allows us to annotate relational information as well as temporal referring expressions. We also discuss a pilot study in which we assessed the utility and feasibility of the scheme and the annotation tool we have developed to aid the annotation process. Finally, we discuss potential improvements in the annotation tool which are aimed at making the annotation of larger scale corpora possible.

1. Introduction

Many natural language processing applications, such as information extraction, question answering, topic detection and tracking, and multi-document summarisation, would benefit significantly from the ability to accurately position reported events in time, either relatively with respect to other events or absolutely with respect to calendrical time. However, only recently has concerted work been started on the automatic extraction of temporal information from text.

In addressing the goal of extracting temporal information from text, it is necessary to:

1. specify the target temporal representation which we wish to obtain for a text;
2. identify ancillary information which we **may** want to extract because of its utility in arriving at the target temporal representation (by analogy with, e.g. part-of-speech tagging or parsing as intermediate goals towards semantic interpretation).

For example, one candidate for target representation is an association of a calendrical time point or interval with each event in a text, i.e. a list of pairs of calendrical times and events. Arriving at this representation might require extracting additional information, such as temporal relational information, about events. For example, assigning "before 1984" to an event A might only be possible by recognising that event B occurs in 1984 and that A occurs before B. Thus, the capability to determine temporal relations between events might be a useful component capability in a temporal information extraction system, even if the information identified by such a component is not directly included in the target representation.

Our view is that target representation should be a **time-event graph** where the nodes in the graph are either times or events and the arcs are temporal relations. This is somewhat different from the "time-stamping" representation introduced in the preceding paragraph and one of our major goals in this paper is to argue that it is a superior representation.

With respect to ancillary temporal information to be extracted, our view is that time-referring expressions, event representatives, and temporal relations as signalled by, e.g. prepositions and temporal adverbials, all convey important temporal information and should be extracted. This information is necessary to derive a time-event graph for a text; but of course it is useful for creating a time-stamp representation as well – arguably both necessary and sufficient.

In this paper, we first give an overview over existing approaches to temporal annotation and information extraction in Section 2. Then in Section 3. we discuss the importance of a target representation that captures temporal relations and describe the annotation scheme we have developed to do so. Section 4. presents some results of a pilot study we have conducted based on the scheme. Further improvements to the process of annotation, to support the creation of larger annotated resources, are discussed in section 5.

2. Overview of Existing Approaches

Existing approaches to capturing temporal information in text can be divided broadly into the following three groups: (1) approaches that concentrate on an accurate and detailed annotation of temporal referring expressions, (2) time-stamping approaches that aim to associate a calendrical time with some or all events in the text, and (3) approaches that focus on the temporal relations between events and times, between events and events or both. We give a brief overview of existing work on each approach in this section.

2.1. Annotating Temporal Referring Expressions

The most extensive work on annotating temporal referring expressions so far has been done as part of the MUC language technology evaluations or the subsequent TIDES[1] and ACE [2] programmes.

2.1.1. MUC Named Entity Task

Between 1987 and 1998 the DARPA-sponsored Message Understanding Conferences (MUCs) developed a quantitative evaluation regime for message understanding (MU) systems, now generally called information extraction (IE) systems. The last MUC, MUC-7, was held in 1998, but related work continues within the ACE workshops. For more information about the message understanding conferences see MUC (1998).

While MUC evaluations typically defined several evaluation tasks, the relevant task here is the *named entity (NE) recognition* task, introduced in MUC-5. The NE task required the recognition and classification of specified named entities such as persons, locations, organisations, monetary amounts and, most importantly in the current context, time expressions (timex). The aim of the timex task was to mark up time expressions in text using SGML tags and to classify these expressions using a TYPE attribute. Type DATE referred to complete or partial date expressions and type TIME referred to complete or partial expressions of time of day. Both absolute and relative time expressions had to be marked up, although these two types were not distinguished in the annotation.

In the MUC-7 evaluation, the best systems were able to obtain F-measure scores approaching 94% on this task.

2.1.2. An Annotation Scheme for Temporal Expressions

Wilson et al. (2001) describe a set of guidelines[3] being developed within the TIDES programme for annotating time expressions and associating with them a canonical representation of the times to which they refer. A method for extracting such time expressions from multiple languages is also introduced. The main novel features as compared to the MUC temporal annotation task are:

1. In MUC the task called merely for surface time expressions to be annotated and crudely classified, whereas the Wilson et al. (2001) guidelines also call for each expression to be *evaluated*, i.e., to have associated with it a normalised representation of the time referred to.

2. The range of expressions flagged is much wider.

3. Context-dependent time expressions like *today* are handled in addition to fully specified time expressions like *September 3rd, 1997*. Context can be local (within the same sentence) or global (outside the sentence). Indexical time expressions, that require knowledge about the time of speech, like *now* are also included. A corpus study (Wilson and Mani, 2000) showed that two-thirds of time expressions in print and broadcast news are context dependent, so this feature is significant.

Wilson et al. (2001) have developed a tagger to do time expression tagging as described in the TIDES guidelines, and report F-measure scores of 96.2% on expression identification and 83.2% on evaluating these expressions.

2.2. Time-Stamping of Events

Annotating temporal referring expressions is only a first step towards extracting rich temporal information from text. The approaches introduced in this section aim at 'stamping' some or all events in a text with a calendar time – possibly the time value of an associated temporal referring expression.

2.2.1. MUC-5 and MUC-7 Time Slots

In addition to the Named Entity time expression tagging task, MUC-5 and MUC-7 also required relations between times and events to be established as part of the scenario template task. Participants were required to assign a calendrical time to certain specified event types (joint venture announcements and rocket launchings, respectively).

Scenario template filling requires the identification of specific relations holding between template elements. For example, the MUC-7 scenario template filling task concerned rocket launch events. The scenario template contains information about vehicles, pay load, launch site, mission function etc. It also contained a slot called LAUNCH_DATE, which was to be filled with a link to a time entity which in turn contained slots for a normalised representation of the start and end times of the temporal interval containing the launch event, if the interval could be determined from the text.

Temporal relations between events and other events were not explicitly addressed, though insofar as they were necessary to infer correct slot fills, systems needed to take them into account. Scores were quite low on this slot reflecting the difficulty of correctly assigning to it.

2.2.2. Assigning Time-Stamps to Event Clauses

In the MUC task, times were only to be determined for the events of interest, the scenario events. A more ambitious goal is to attempt to associate calendrical times or time intervals with *every* event in a text.

Filatova and Hovy (2001) describe a method for breaking news stories into their constituent events and assigning time-stamps to them. The time-stamps assigned are either full specified calendrical dates, sets of dates, closed date ranges (both end points specified), or date ranges open at one end or the other, indicating some time before or after the specified date.

The syntactic units conveying events are assumed to be simple clauses and they are identified using a parser which produces semantically labelled syntactic parse trees. Some problems are ignored in this approach, for example multiple verbs with different tenses in one sentence.

The time-stamper uses two time-points for anchoring. One time-point is the time of the article (at the moment only the date is used and the time of day is not taken into

[1] See http://www.darpa.mil/ipto/research/tides/.

[2] See http://www.itl.nist.gov/iaui/894.01/tests/ace/.

[3] The full set of guidelines are available as Ferro et al. (2000)

account) and the other time-point is the last time-point assigned within the same sentence. The procedure of time-stamping is as follows:

1. The text is divided into event clauses
2. All date phrases in the text are extracted
3. A date is assigned to each event clause based on either
   (a) the most recent date phrase in the same sentence, or
   (b) if this is not defined, then the date of the article.

In assigning dates various time assignment rules are used. When a date phrase is present in the sentence these rules both take into account nearby prepositions, such *on, after, before*, and carry out fuller specification. For example if the date phrase is simply a day of the week, then the article date is also used to derive a date-stamp that is fully specified with respect to year and position within the year. If no date phrase is present in the sentence then tense information is used to assign a time interval relative to the date of the article.

After all events have been stamped with a time, the event clauses are arranged in chronological order. The authors report scores of 77.85% correct time-stamp assignment to event clauses which have been manually (i.e. correctly) extracted from sample texts of a small trial corpus.

2.2.3. Temporal Semantic Tagging of Newswire Texts

The ultimate goal for Schilder and Habel (2001), as for ourselves, is to establish the temporal relations between all events in news articles.

In Schilder and Habel's approach temporal expressions are classified into *time-denoting expressions* that refer to a calendar or clock time and *event-denoting expressions* which refer to events. They view their goal as anchoring these temporal expression on the absolute time-line, so as to produce a linearly ordered anchored set of temporal entities; hence a time-stamp representation appears to be their target representation. For time-denoting expressions this may mean resolving indexicals (*now*, *yesterday*) or fleshing out expressions like *Thursday* to fully specified calendar dates. For event-denoting expressions a calendar time which is the time of the event must be associated with the event, possibly by extracting temporal relations which are signalled by prepositional phrases like *on Friday*. The set of temporal relations proposed is *before, after, incl, at, starts, finishes* and *excl* (equivalent to Allen (1983)'s relations).

They have developed a semantic tagging system for temporal expressions in newswire articles. The main part of their system is a Finite State Transducer (FST) based on handwritten rules. Their target language is German. The FST tags all time-denoting expressions, all verbs and an experimental version tags event-signalling nominal expressions. A semantic representation is then proposed, based on which inferences are drawn, especially about temporal relations. In its current state, the FST establishes temporal relations between times and events. The tagger was evaluated with respect to a small corpus (10 news articles) and an overall precision rate of 84.49% was achieved.

2.3. Annotating Temporal Relations

The work described in the preceding section aims at associating a calendar time with some or all events reported in a text, but none of these approaches view the identification of temporal relations as a explicit goal in its own right. These temporal relations are clearly of importance, even for time-stamping approaches. The work described in this section, as well as the approach we develop in the next section, address temporal relations directly.

2.3.1. Annotation of Intrasentential Temporal Information

Katz and Arosio (2001) aim to create a large multilingual corpus, in which intrasentential temporal relations are tagged in addition to standard morphological and syntactic features. To aid this, they have developed a language-neutral and theory-neutral method for annotating sentence internal temporal relations. With this corpus, Katz and Arosio (2001) hope to be able to automatically acquire the lexical knowledge required for determining temporal interpretation in narrative discourse.

A temporal interval is associated with each verb in the sentence; it is the temporal relations between these intervals that are of concern. The temporal interpretation should be closely linked to the syntactic context (which is of importance since it is not known beforehand to what degree the cues used by the speaker are lexical and to what degree they are grammatical). This linking is needed to keep track of both the semantic relations among times as well as the syntactic relations among the words in the sentences that refer to these times.

The authors add a layer of semantic annotation to already syntactically annotated text. The verbs in the sentence are linked via secondary edges labelled with a temporal relation. Precedence and inclusion and their duals are the possible relations. Indexical information is included by introducing the symbol $\circ$ for the speech time, which is automatically prefaced to all sentences prior to annotation.

A searchable multi-language annotated treebank has been created where each sentence is stored in a relational database with both syntactic and temporal annotations. This makes is possible to query the corpus ("Find the sentences containing a relative clause which is interpreted as temporally overlapping the main clause" (Katz and Arosio, 2001)).

This work is valuable, especially for linguists interested in the studying, cross-lingually, the complex interrelationship of lexical and syntactic mechanisms used to convey temporal relations between events in the same sentence. However, if one's goal is extraction of the full temporal content of a text, it is limited in only considering intrasentential temporal relations.

3. Annotating Temporal Information in Text

From the preceding overview of existing work on temporal information extraction it is clear that the bulk of work so far has gone into the identification of temporal referring expressions and the assignment of time-stamps to events. Only Katz and Arosio (2001) focus directly on the prob-

lem of identifying temporal relations between events, and in their case only between events in the same sentence.

In this section we start by arguing that a time-event graph, in which not all events are necessarily directly anchored on a time-line, is a superior target representation for a text to a time stamped representation. We then present the conceptual underpinning for the approach we advocate for annotating temporal information in text, followed by the details of the annotation scheme itself.

3.1. Why Annotate Temporal Relations?

Recall that a time-event graph is a graph in which the nodes are either times or events and the arcs are temporal relations. There are two principal arguments for preferring a time-graph representation to a time-stamp representation.

First, in many cases texts position events in time only by relation to other events and any attempt to coerce these events onto a time-line must either lose information, invent information, or rely on a notion of an underspecified time point constrained by temporal relations (i.e. introduce a representation of temporal relations by the back door).

Consider this example:

> After the plane crashed, a search was started. Afterwards the coast guard reported finding debris.

and assume that an earlier sentence specifies the calendrical time of the plane crash.

An approach attempting to map the information presented in this example onto a time-line is faced with the situation depicted in Figure 1.

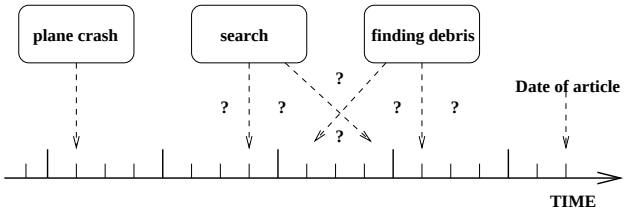


Figure 1: A Time-line Representation

While the crash event can be placed on the time-line the other two events cannot. Either time points must be guessed, or an interval be assigned. The first option is clearly not satisfactory. But if an interval is assigned the only possible interval, for both the searching and finding events is the interval from the crash till the date of the article. But if this is assigned to both events then the information about their ordering with respect to each other is lost.

A simpler representation which while not attempting to be as specific actually carries more information is shown in Figure 2.

This representation preserves the information that the searching event precedes the finding event, without forcing any early commitment to points on a time-line.

The second argument for preferring a time-event graph representation that captures event-event temporal relations as well as time-event relations is that to position events on a time-line accurately requires the extraction of event-event relational information. In the example, the placing of the searching and finding events in the interval between the plane crash and the date of the article requires the recognition that these events occurred after the crash as signalled by the words “after” and “afterwards”. Without identifying the relations conveyed by these words the searching and finding events could only be positioned before the time of the article, and not following the plane crash. Thus, even if a time-stamp representation is viewed as the best target representation, achieving it requires the extraction of temporal relational information. In this case adopting a time-event graph as an intermediate representation is still a good idea, which begs the question of why it should not simply be taken as the final target representation.

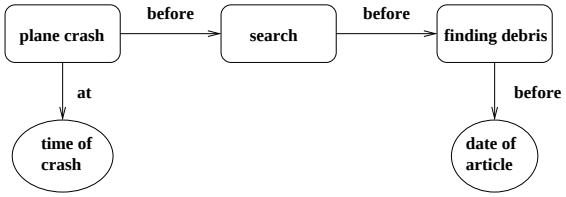


Figure 2: A Time-Event Graph Representation

3.2. Conceptualising Time

Before we describe the annotation scheme we have developed, we will very briefly explain what kind of temporal entities and relations we suppose exist. We presume the world contains the following primitives: events, states, times, temporal relations and event identity. Each primitive is described briefly below.

Events Intuitively an event is something that happens, something that one can imagine putting on a time map. Events can be ongoing or conceptually instantaneous, we do not distinguish between these. What defines an event is very much dependent on the application and domain, but generally events have to be anchorable on a time-line and they are usually conveyed in language by finite verbs or by nominalisations. Examples of events are:

> *A small single-engine plane **crashed** into the Atlantic Ocean.*
>
> *The 1996 **crash** of the TWA 747 remains unexplained.*

Times Like events, times can be viewed as having extent (intervals) or as being punctual (points). Rather than trying to reduce one perspective to the other, the focus of much of the philosophical debate on time, we shall simply treat both as *time objects*. A time object must, however, be capable of being placed on a time-line (fictional or real).

Following general convention, and the approach taken in MUC, we distinguish between two classes of time objects, DATES and TIMES, time objects which are larger or smaller than a day, respectively.

States A state is a relation between entities or the holding of an attribute of an entity which, while capable of change, is ongoing over a time span, usually longer than the time span covered by the text of interest. Examples are:

> *The plane, **which can carry four people**, ...*

*The water **is about 125 feet deep** in that area.*

Typically, a change of state constitutes an event. At this point we are less interested in states, and we have not taken them into account in our annotation scheme.

Temporal Relations Events stand in certain temporal relations to other events and to times. Times are temporally related to other times as well, but this phenomenon is not only very rarely explicitly expressed in text, it is also of lesser importance and is not taken into account here.

*The plane crashed **after** the pilot and his crew ejected.*

*A small single engine plane crashed into the Atlantic Ocean **on** Wednesday.*

The full set of temporal relations we suppose at present is { *included, includes, after, before, simultaneous* } . This is a minimal set, which was defined after analysing a number of newspaper articles, and can easily be expanded.

3.3. The Annotation Scheme

Given this conceptual framework, we can describe the annotation scheme we have defined. For more details see Setzer and Gaizauskas (2000).

Annotating Events Events are marked by annotating a representative in the clause conveying the event. The first choice for a representative is the head of the finite verb group. If a nominalisation conveys the event, then the head of the nominalisation serves as the representative. In the rare case of an event being conveyed by a non-finite clause, the non-finite verb is marked as the representative.

An events carries attributes for some or all of the following properties: unique event ID, event class, verb tense, verb aspect, other event to which it is related and temporal relation by which it is related, time object to which it is related and temporal relation by which it is related, the word(s) by which the temporal relation is signalled, and the ID of events it might have as an argument. For example, ignoring temporal relations for the moment:

A small single-engine plane
<event eid=16 class=OCCURRENCE tense=past>
crashed
</event>
into the Atlantic Ocean about eight miles off New Jersey

Annotating Times We distinguish between *simple* and *complex* time referring expressions. Simple time referring expressions refer to times directly, as in example (1). Complex time referring expressions, as in (2), refer to a point in time by relating (*after*) an interval (*17 seconds*) to an event (*hearing the sound*). The point in time referred to is the point at the end of the interval.

(1) *last Thursday*

(2) *17 seconds after hearing the sound ...*

For simple time referring expressions we annotate the whole text span conveying the time-object:

<timex tid=5 type=DATE calDate=12041997>
last Thursday </timex>

Each time referring expression has a unique ID, an attribute flagging whether it is a time or a date, and an attribute carrying the calendar date the expression refers to.

Complex time referring expressions, like the one in example (2), include a time interval (*17 seconds*), a preposition (*after*) and an event (*hearing the sound*) or time. The way these are annotated is similar to the way events are annotated. The interval is chosen as the representative for the time referring expression and related to the event expression via the temporal relation, usually signalled by the preposition.

<timex tid=5 type=complex eid=3 signalID=7
relType=after> 17 seconds </timex>
<signal sid=7> after <signal>
<event eid=3> hearing</event> the sound...

Annotating Temporal Relations Events and times can be related to other events or times. If two events are related then one of the events carries the ID of the other as well as the temporal relation in which they stand to each other. If an event is related to a time then the event carries the ID of the time object and the temporal relation. In either case, if the relation is signalled explicitly in the text, then the ID of this signal is an attribute as well, as the following two examples illustrate.

All 75 people on board the Aeroflot Airbus
<event eid=4 class=OCCURRENCE tense=past
relatedToEvent=5 eventRelType=simultaneous
signal=7>
died </event>
<tr_signal sid=7> when </tr_signal>
it
<event eid=5 class=OCCURRENCE tense=past >
ploughed </event>
into a Siberian mountain.

A small single-engine plane
<event eid=9 class=OCCURRENCE tense=past
relatedToTime=5 timeRelType=included
signal=9>
crashed </event>
into the Atlantic Ocean about eight miles off
New Jersey
<tr_signal sid=9> on </tr_signal>
<timex tid=5> Wednesday </timex>.

If the temporal relation is implicitly expressed, then the only difference is that the attribute for the signal is simply left out.

One problem with this annotation scheme is that it is not possible to relate one event to two or more other events or times, though by and large we have not found this to be a problem in annotating real text. This problem has been addressed by the TERQAS[4] workshop, which is working towards defining a general time markup language and has adopted many aspects of the current annotation scheme.

[4]See http://www.cs.brandeis.edu/ jamesp/arda/time/.

The solution proposed there is to introduce independent SGML `LINK` entities, which consume no text, to serve as relational objects tying events and times together. One event can then participate in as many links as is necessary.

4. The Pilot Study

To study the feasibility of the annotation scheme and to gain insight into the linguistic mechanisms conveying temporal information in text, we have applied the annotation scheme to a small trial corpus.

4.1. The Corpus

The trial corpus consists of 6 newswire articles taken from the New York Times, 1996, which were part of the MUC7 (MUC, 1998) training data. Basic statistics about the corpus are presented in table 1.

	sentences	words	number of annotators
text1	26	448	3
text2	18	333	2
text3	13	269	3
text4	13	213	2
text5	10	211	3
text6	13	399	3
total	93	1873	3

Table 1: The corpus

Each text was annotated by either two or three annotators, in addition to one of the authors, who produced what in the following is taken to be the 'gold standard' or 'key' annotation.

4.2. The Process of Annotation

The annotation takes place in two stages, both of which are described briefly in this section. To aid the annotator with her or his task, we have developed an annotation tool which not only allows the annotation of the information required by the scheme but which also interactively supports the annotator during the second phase, where additional temporal relations are established.

Stage I During Stage I, all event and time expressions are annotated as well as all signal expressions. Afterwards, those temporal relations that are explicitly expressed, e.g. by temporal prepositional phrases or subordinate clauses, and hold between events or events and times are established and stored as event attributes. Some implicitly expressed temporal relations are also established during this stage, for example, when events are clearly positioned in time but the signal expression has been omitted, as in *The army said Friday [...]*. In addition, *ing*-clauses without a subject can also be used to implicitly express a temporal relation between two events and are annotated during this stage.

Stage II The annotation scheme we have developed is aimed at establishing as many temporal relations in the text as possible. To relieve the burden on the annotator, and to increase the number of temporal relations annotated, we introduced stage II, which is cyclical in nature. Based on the information available, which in the beginning consists of the events, times and the temporal relations annotated in stage I, all inferences possible are drawn, according to an axiomatisation of the temporal relations *included, includes, after, before, simultaneous*. This is conducted automatically by the annotation tool which computes the deductive closure over these temporal relations. If the temporal relation between any pair of events or events and times is still unknown, the annotator is prompted for one of these [5] and, again, all possible inferences are automatically drawn. The process continues until every event-event and event-time pair in the text has been related.

4.3. The Results

In this section, we briefly describe the distribution of temporal phenomena over the trial corpus, as far as this is relevant to the issues discussed in this paper. Note that although this is a trial corpus, the results are indicative. We will not talk here about recall and precision values of the individual annotators with respect to the gold standard here – see section 5. For more information about the pilot study and its outcome see Setzer (2001).

Table 2 shows the number of event expressions, time expressions, and the number of event-event relations annotated in each text of the corpus in Stage I of the annotation process – i.e. these are the temporal relations that are explicitly expressed in the texts.

	# sentences	# words	# event expr.	# event-event relations	# event-time relations
text1	26	448	40	10	12
text2	18	333	30	10	5
text3	13	269	19	7	3
text4	13	213	27	5	0
text5	10	211	16	1	4
text6	13	399	26	13	5
total	93	1873	158	46	30

Table 2: Number of event expressions and explicit temporal relations per text

Table 3 shows for each text the number of event and time expressions in the text, the number of explicit temporal relations annotated in Stage I, the number of relations inferred from these without any further input from the annotator, the number of relations solicited from the annotator (i.e. the implicit temporal relations), and the number of inferred temporal relations overall.

4.4. Discussion

In Section 3.1. we criticised the time-stamped event sequence as a target representation on two grounds:

1. Forcing events to be placed on a time-line may result in the loss of event-event ordering information,

[5] Note that *unknown* is a possible value for a temporal relation here.

	event + time expr.	annotated ev.-ev. and ev.-time relations	inferred relations based on annotated relations	solicited rel.	total inferred relations
text1	32 + 11	10 + 12	222	124	1005
text2	26 + 5	10 + 5	122	93	380
text3	17 + 3	7 + 3	21	49	141
text4	18 + 0	5 + 0	8	45	120
text5	10 + 4	1 + 4	13	18	110
text6	24 + 5	13 + 5	107	52	514
total	127 + 28	46 + 29	493	381	2270

Table 3: Annotated, solicited, and inferred temporal relations

since the time-stamps assigned to distinct events may be identical even though we know the events occurred at separate times and know their order.

2. Event-event relational information must be extracted in order to position events on a time-line. Given this, why not choose a target representation that includes this richer information.

While both of these observations are true in general, ideally we would like to substantiate them empirically and quantitatively with respect to the trial corpus. Unfortunately we have not as yet been able to carry out the analysis for the whole corpus. However, we have chosen one text from the corpus (text6) and investigated it in detail.

To corroborate the first point above, we read text6 and, assuming perfect knowledge of the temporal information contained, then represented this information on a time-line, associating an interval with each event. In other words, without worrying about *how* the temporal information is extracted we time-stamped each event, where each time-stamp contains a start and end time expressed as calendar dates or, for at most one of the times, a symbol indicating the time is unknown.

For example, the sentences *A senior investigator looked at the wreckage Tuesday* and *Flight 800 exploded midair 20 days before Tuesday and then plunged into the ocean*[6] can be represented on a time-line as shown in Figure 3.

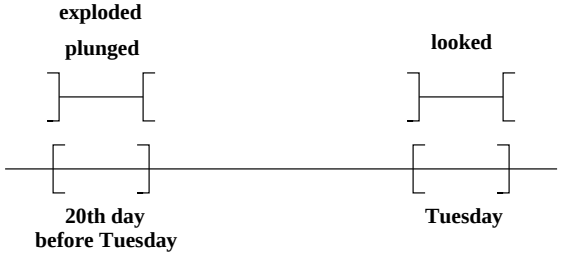


Figure 3: Example of a time-line Representation

Note that the events *exploded* and *plunged* have to be associated with the interval which encompasses the 20th day before Tuesday. We have lost the information that the plane plunged into the ocean **after** the explosion. This information can be easily represented in a time-event graph, as shown in figure 4.

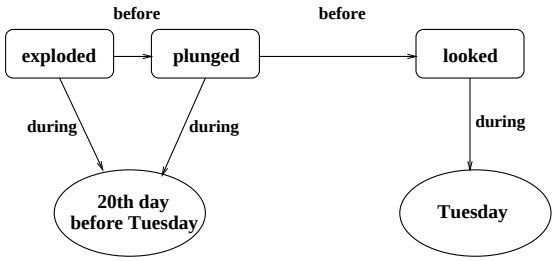


Figure 4: Example of a time-event graph Representation

Overall, 7 event-event relations that were explicitly mentioned in the text were lost in the time-line representation. While we have not performed the detailed analysis to let us say how many of the 514 inferred temporal relations in text6 are dependent on these 7 relations, it seems fair to assume that a significant number are.

To corroborate the second point we investigated how many of the 107 relations inferred for text6 from the explicitly annotated event-event and event-time relations resulted in new event-time relations involving events for which no event-time relation existed already. This corresponds to the intuitive notion of how many events are placeable on the time-line solely due to event-event relational information. For text6 we discovered that 20 of the 107 new relations were time-event relations for events for which no previous time-event relation existed. These 20 relations mentioned 4 distinct events (i.e. these 20 relations involved relating 4 events to different times, perhaps redundantly, but also potentially defining separate start and end points for intervals associated with them). Thus, 4 of the 24 events in text6 can be placed on a time-line using event-event relational information which is explicitly present in the text – positional information that otherwise would either be lost or require knowledge of implicit relations to extract.

Finally, we can make the general observation of the trial corpus that from 127 event-event relations plus 28 event-time relations, a total of 2270 additional temporal relations has been inferred. Even though we do not have the exact figure of how many of these inferred temporal relations are based on annotated event-event relations, it seems likely that the event-event relations contribute significantly to the number of relations inferred. We base this observation on the fact that there are nearly twice as many event-event relations as event-time relations annotated, and that subsequent inferences in the deductive closure calculation build on these initial relations. This observation adds weight to our claim that annotating event-event relations is important for temporal information extraction.

5. Improvements to the Annotation Process

The pilot study has shown that the interannotator agreement and the recall and precision figures need to be improved and that the burden on the annotator needs to be lessened, before the annotation scheme can be used to cre-

[6]The sentences have been slightly altered to make them more comprehensible out of context, but the temporal information they convey is the same as in the original text.

ate a larger corpus. Larger corpora will be necessary to train and evaluate temporal information extraction systems.

In Setzer and Gaizauskas (2001) we identified five main causes of low annotator precision and recall scores (with respect to the gold standard): imprecision/incompleteness of the guidelines; imperfect annotator understanding of the task; intrinsic difficulty of identifying the appropriate temporal relation in some cases; annotator fatigue; and annotator carelessness. In this section we do not address all of these problems, but focus on a number of proposals to enhance the annotation process, thereby lightening the load on annotators and increasing the accuracy of the annotations.

Pre-tagging An automatic first annotation pass could be used to reduce the amount of manual annotation and to raise recall. A part-of-speech tagger or word group parser, could be used to mark up finite verbs and signals and a time expression tagger such as Wilson et al. (2001)'s could be used to tag time referring expressions. Using the corpus as an indication, we know that a large percentage of the finite verbs will indicate events and the annotator can easily add attribute information to those or delete the mark up of mistakenly flagged verbs which do not indicate events. The high accuracy of time expression taggers mean that most of the work of tagging time expressions would be done automatically with the annotator left only to confirm details and scan for missed expressions.

Signals are a slightly different case. These are mostly prepositions and subordinating conjunctions, but a smaller number will have to be marked up. Here we have two options. We can mark up all prepositions and conjunctions and leave it to the annotator to delete inappropriate annotations, which is an easy process. Alternatively we could only automatically annotate those prepositions which are followed by a time referring expression. This approach carries the danger of not pre-annotating all signals, and the annotator, concentrating on the pre-annotated sections, might not catch all signals.

Intelligent Interaction with the Annotation Tool: Question Ordering The second phase of stage II of the annotation process, during which the deductive closure over the temporal relations is calculated and the annotator is prompted for unknown temporal relations, is problematic for the following reasons.

1. It is a long process, during which the annotator was prompted for 62 temporal relations per text on average, even for the short texts in the pilot corpus.

2. There is, for now, only marginal consistency checking and it is not possible to correct errors. Once the annotator notices that she or he made a mistake earlier in the process, then the whole stage II annotation process has to be restarted.

One possible solution for the first problem would be to optimise the order in which unknown temporal relations are prompted for. As we explained in section 4.2., after each temporal relation solicited from the user, all possible inferences are drawn. The larger the number of the inferences, the smaller the number of remaining unknown temporal relations will be. The following simple example illustrates the effect non-optimal soliciting can have. Imagine four events, forming a 'precedence chain':

$$e_1 < e_2 < e_3 < e_4$$

Imagine also that the link between e_2 and e_3 is missing in the response:

$$e_1 < e_2 \qquad e_3 < e_4$$

If the first question establishes the temporal relation holding between e_2 and e_3, then all other temporal relations can be inferred, based on the transitivity of before. The temporal model can be completed with one question. However, the order of questioning could be very different, establishing the temporal relations between e_1 and e_4, then between e_1 and e_3, e_2 and e_4 and then between e_2 and e_3. In this case four questions are asked to establish the relations holding between them.

Thus, question order can be important in determining how many questions the annotator ultimately gets asked. Clearly, one wants to minimise the number of questions asked, but it is not clear (to us) whether there is a question order that is guaranteed to minimise this number, and if so how to determine it. We propose to investigate initially a naive approach in which given two temporally-ordered event chains we first ask questions which attempt to link their end points, simply on the grounds that such questions could lead to maximal gains. However, considerably more empirical and theoretical investigation needs to be carried out here.

Intelligent Interaction with the Annotation Tool: Correcting Mistakes The second point requires a more elaborate solution. Once an incorrect temporal relation has been added an indeterminate number of further incorrect inferences may have been drawn on the basis of it. Two solutions suggest themselves:

1. Provide the possibility of check-pointing, i.e. saving intermediate stages to which the annotator can return when an error has been detected. This could be done automatically after each new user-solicited relation is added. This has the advantage of being easy to implement but the disadvantage of erasing possibly correct temporal relations added after the error, but independently of it, with the consequence that work that will have to be redone unnecessarily.

2. Implement a sort of truth maintenance system (Doyle, 1987; de Kleer, 1987), whereby only the incorrect temporal relation and those temporal relations which were inferred from it are deleted. This has the advantage of minimising the amount of work the annotator needs to redo unnecessarily, but the disadvantage of being more complex to implement.

Clearly the second solution is the better in the long run, as annotator effort is the chief quantity to conserve. We are working on solution whereby all temporal relations added to the temporal fact database record with them a justification which includes a reference to any facts from which they have been derived. Removing a temporal fact f then becomes a recursive procedure which begins with a search

for all facts f' whose justification mentions f followed by a recursive call to delete f'. This will ensure that all dependents of f will be removed, while not touching any facts, solicited or derived, that may have been added after f in the annotation process, but which are logically independent of it.

6. Conclusion

We have argued that when extracting temporal information from texts a target representation, such as a time-event graph, which explicitly admits event-event temporal relations as well as time-event relations, is superior to one which does not, such as a time-stamped event sequence representation. In essence the arguments are that a time-stamp representation forces overspecification leading to information loss, and that event-event relations must be extracted even if a time-stamp representation is the target, and hence might as well be retained.

We also described the annotation scheme we have developed, which enables us to annotate temporal relations as well as events and time referring expressions, thus providing the necessary information to build time-event graphs for texts. A trial corpus which we constructed based on this scheme was described and used to corroborate the argument in support of the time-event graph approach.

One potential practical argument against the time-event graph approach is that building annotated resources capturing the required information is costly and error-prone. In the final section of the paper we introduced ideas for improving quality and reducing effort in the annotation process, improvements which we hope will make future larger scale application of the annotation scheme feasible.

7. References

J.F. Allen. 1983. Maintaining Knowledge About Temporal Intervals. *Communications of the ACM*, 26:832–843.

J. de Kleer. 1987. An Assumption-based TMS. In M.L. Ginsberg, editor, *Readings in Nonmonotonic Reasoning*, pages 280–298. Morgan Kaufman Publishers, Los Altos, California.

J. Doyle. 1987. A Truth Maintenance System. In M.L. Ginsberg, editor, *Readings in Nonmonotonis Reasoning*, pages 259–279. Morgan Kaufman Publishers, Los Altos, California.

L. Ferro, I. Mani, B. Sundheim, and G. Wilson. 2000. TIDES Temporal Annotation Guidelines. Technical Report MTR 00W0000094, The MITRE Corporation, October. Draft-Version 1.0.

E. Filatova and E. Hovy. 2001. Assigning Time-Stamps to Event-Clauses. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 88–95, Toulouse.

G. Katz and F. Arosio. 2001. The Annotation of Temporal Information in Natural Language Sentences. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 104–111, Toulouse. Association for Computational Linguistics.

1998. *Proceedings of the Seventh Message Understanding Conference (MUC-7)*. Morgan Kaufman. http://www.itl.nist.gov/iaui/894.02/related_projects/muc/.

F. Schilder and C. Habel. 2001. From Temporal Expressions to Temporal Information: Semantic Tagging of News Messages. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 65–72, Toulouse.

A. Setzer and R. Gaizauskas. 2000. Annotating Events and Temporal Information in Newswire Texts. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC2000)*, pages 1287–1293, Athens, Greece.

A. Setzer and R. Gaizauskas. 2001. A Pilot Study On Annotating Temporal Relations In Text. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 73–80, Toulouse, France.

A. Setzer. 2001. *Temporal Information in Newswire Articles: An Annotation Scheme and Corpus Study*. Ph.D. thesis, University of Sheffield.

G. Wilson and I. Mani. 2000. Robust Temporal Processing of News. In *Proceedings of the 38th Meeting of the Association of Computational Linguistics (ACL 2000)*, pages 69–76, Hong Kong, October.

G. Wilson, I. Mani, B. Sundheim, and L. Ferro. 2001. A Multilingual Approach to Annotating and Extracting Temporal Information. In *Proceedings of ACL-EACL 2001, Workshop for Temporal and Spatial Information Processing*, pages 81–87, Toulouse.

Tense and Implict Role Reference

Joel Tetreault

University of Rochester
Department of Computer Science
Rochester, NY, 14627
tetreaul@cs.rochester.edu

Abstract

This paper presents a probe study into the use of temporal information in the resolution of implicit roles. It has been shown that temporal information can influence rhetorical relations between utterances which in turn can influence the resolution of referential entities. We describe a focusing-based algorithm for resolving implicit roles using such information and test the algorithm on an annotated corpus. Our results show that temporal information can be useful in resolving implicit roles in some cases, but a much larger and varied corpus is needed to strengthen this claim.

1. Introduction

This paper describes preliminary work relating tense to implicit role reference. Past work has shown that tense can influence the resolution of other reference types such as pronouns as well as discourse structure ((Webber, 1988a) and (Hwang and Schubert, 1992)). We extend this claim to the reference of implicit roles. Verbs have certain required roles, which refer to discourse entities, that are necessary for comprehending the verb phrase, and thus aid in natural language processing. Roles that do not have explicit antecedents in the verb phrase are deemed implicit. We annotated a small corpus for NPs and VPs and tense information and show, in some cases, that one can improve resolution rates of implicit roles by using simple heuristics incorporating tense with focusing. To our knowledge, this is the first time that an automated corpus study has been done analyzing the effects of temporal information in reference.

First, we describe implicit role reference in more detail and how temporal information can be used in resolving implicit roles. In section 3, we describe our annotation scheme, in section 4, our focusing-based algorithm for implicit roles and finally the results of our algorithm on an annotated corpus.

2. Implicit Role Reference

We claim that in addition to canonical reference types such as pronominal reference, VP ellipsis, and discourse deixis, verb phrases have certain required roles that can be viewed as anaphoric. These required roles refer to discourse entities and are necessary for the interpreter to understand the verb phrase, and thus the complete utterance. For example, in order to use the verb "take" one needs to understand that an entity is being moved, that it is being moved to one place from some other place, and that there is some entity that is responsible for moving it.

Implicit role reference has been briefly studied as a side effect of bridging and discourse relations ((Poesio, 1994) and (Asher and Lascarides, 1999)) but no major empirical work has been done in the area.

Resolution of implicit roles occurs frequently in naturally occurring dialog. Consider the following, modified from Asher and Lascarides (p. 90):

```
(1)  Take engine E1 from Avon to Dansville.
(2a) Pick up the box-
car and take it to Broxburn.
(2b) And then take the box-
car from Corning.
(2c) Also take the boxcar.
(3)  Leave E1 there but move the box-
car some more to Evansville.
```

For the sake of simplicity, assume that the verb "take" has these roles: "Theme": the entity being moved; "To-Loc": the location we are taking the "theme"; and "From-Loc" the location we are leaving.

In utterance (2a) one needs to know the At-Loc of the boxcar, in order to send it to Broxburn. This role is implicit and is resolved to Dansville. In order to resolve "there" in utterance (3) after utterance (2b) one must resolve the implicit "From-Loc" in "take" in the previous sentence. But the main point is that in a natural language understanding/planning system, one must keep track of entities and their locations in order for it to plan and carry out the task.

Asher and Lascarides point out that use of rhetorical roles, specifically whether utterances are in a narrative or parallel relationship, can aid in reference resolution. For example, the relationship between (1) and (2a) is a narrative while (1) and (2c) is parallel. While it is hard to annotate rhetorical relations we believe that one can approximate them by calculating the temporal relation between the two utterances. For instance, if we know that there is a narrative relation then we know that the entity that serves as the To-Loc role will probably serve as the From-Loc role in the next utterance, since entities move from the place they were just taken.

In our corpus we found the following distribution (see Figures 1 & 2) for the roles we focus on in this study (From-Loc and To-Loc) and their antecedents. These figures show that for a given role, how many sentences back (depth) its antecedent is found and in what role focus list it is located in. The trend is that antecedents for a From-Loc or To-Loc are predominantly found in the current utterance or the previous two utterances.

We describe our work in implicit roles in more detail in (Tetreault, 2002).

Depth	From-Loc	To-Loc
1	11	9
2	4	1
3	0	0
4	1	0
5+	0	0
%	61.5%	38.5%

Figure 1: From-Loc

Depth	Theme	From-Loc	To-Loc
1	0	1	1
2	0	0	2
3	0	0	0
4	1	0	1
5+	0	1	0
%	14.3%	28.6%	57.1%

Figure 2: To-Loc

3. Annotation

We use a subset of the TRAINS-93 Corpus (Heeman and Allen, 1994) annotated with coreference information for pronouns (Byron and Allen, 1998). The dialogs typically consist of short sentences, usually 10 words or less and are annotated using a sgml-style encoding. Our corpus consists of a 86-utterance dialog in which two human participants are given a task involving moving commodities and trains around a fictional world. We manually annotated each NP with an unique ID and its class (engine, tanker, location, food). Each VP was annotated with an ID, a time ID, and what NP ID(s) each role refers to. If a role is not mentioned explicitly in the text such as the "at-loc" role in (2a), then it is marked as implicit. The roles from each verb are taken from the TRIPS natural language system lexicon (Allen et al., 2000). For all the roles that are marked in this study (instrument, theme, from-loc, to-loc) roughly 30% are implicit.

A time point is associated with each verb event and constraints with previously mentioned time points are included in the time tag. The first element of each time tag is the time point associated with that event and is a string of a character followed by a number such as "t0." There are two types of constraint relations: either time x precedes a time y: "$x < y$" or x follows y: "$x > y$". Multiple constraints for a time point are encoded by linking the individual constraints with an ampersand: "$t1 > t2 \& t1 < t0$" which says that t1 comes after t2 and t1 precedes t0. It should be noted that this is a very naive encoding scheme and that complex verb tenses are reduced to their root forms. A sample annotation (modified for readability) is shown in Figure 3.

Annotation of time points was difficult because the goal of each dialogs was to create a plan not to execute a plan in real-time. This means that the two speakers will often talk abstractly about parts of the plan and create hypothetical plans that may be abandoned if the speakers feel that they would not meet the constraints outlined by the experiment. Often utterances such as "We will need to move the boxcar to Avon by midnight" would appear and be followed by statements related to the introduced task. For our purposes, these multiple stand-alone plans complicate annotation because all time points in the discourse are not necessarily related. To deal with this, we give each sub-plan or hypothetical plan its own code, so one sub-plan may have its events labeled with "u": "$u0, u1, u2...$" while another distinct plan would have "v."

4. Algorithm

We have developed a preliminary model for resolving implicit roles that uses a combination of focusing and temporal reasoning. Our algorithm for resolving implicit roles in a discourse is as follows: first, as one progresses through the discourse, each utterance maintains a focus list for each role, such that when a NP is encountered, its discourse entity representation is placed at the top of the appropriate focus stack(s). When a verb is encountered, we check all of its roles and place explicit ones (those found in surface form of the sentence) on the top of the appropriate focus stack. If a role is implicit then it is resolved as determined by its type:

- Instrument: search through current utterance first for an entity that meets the verb's constraints. If one is not found, then search through each past utterance's focus stacks: looking at the instrument and theme stacks in that order.
- Theme: same as above except that the search order of instrument and theme focus stacks is reversed
- From/To-Loc: use temporal reasoning to determine what order to search past To-Loc and From-Loc lists for each utterance.

Take Engine E1 from Avon to Dansville. Pick up the boxcar.

<ve id=ve122 time=t0 theme=ne12 from-loc=ne5 to-loc=ne6> **Take** <ne id=ne12>**engine E1**</ne> **from** <ne id=ne5>**Avon**</ne> **to** <ne id=ne6>**Dansville**</ne></ve>. <ve id=ve123 time=$t1 > t0$ from-loc=ne6 theme=ne13 implicit=from-loc> **Pick up** <ne id=ne13>**the boxcar**</ne></ve>.

Figure 3: Example Annotation

Algorithm	Instrument	Theme	From-Loc	To-Loc	Overall
R-L	78.9%	55.6%	65.4%	22.2%	61.9%
L-R	78.9%	44.4%	88.5%	44.5%	73.0%
Time, L-R	78.9%	55.6%	61.5%	55.6%	65.1%
Time, R-L	78.9%	44.5%	69.3%	55.6%	66.7%
Total	19	9	26	9	

Figure 4: Implicit Role Reference Results

Our temporal reasoning scheme amounts to determining whether the current sentence u_j is in a narrative or parallel relation with a preceding utterance u_i being searched through for an antecedent. Since we annotated event times we can use the following simple algorithm to assign a narrative or parallel relation: If u_j's event time occurs after u_i's event time then we assume that a narrative relation holds between the two and that a From-Loc role in u_j should search through the To-Loc list in u_i. This is because in a narrative, there is a linear movement from place to place. If no such temporal relation is found, then we assume that a parallel relation holds between u_j and u_i and we search the From-Loc of u_i for antecedents first. The same method is used for To-Loc roles.

5. Results

We implemented the implicit role algorithm in a LISP system and and tested it on our dialog. Figure 4 shows the percentage correct for each version of the algorithm on each implicit role. The first two versions of the algorithm do not use temporal reasoning, while the last two do. R-L indicates that each focus list is searched from right to left, or from most recent to least recent. L-R indicates that the focus list is searched in reverse order, meaning that the subject of that utterance would be prominent. The last line is the number of times that role appears implicitly in the corpus.

6. Discussion

The conclusion of this study is that simple temporal reasoning has a mixed effect on the resolution rate of a verb's implicit roles. While there is a moderate improvement over the resolution of To-Loc's (55.6% to 44.5%), the naive method for resolving From-Loc's clearly outperforms its temporal reasoning counterpart (88.5% to 69.3%). Since our corpus is so small it is hard to draw concrete conclusions on whether not temporal reasoning works, especially since a most-recent strategy performs very well. This is not too surprising however since our statistics show that implicit roles typically have antecedents found locally.

It should be noted that this is a work in progress. Our annotation scheme is very basic and our error analysis shows that many of the From-Loc errors using temporal reasoning are due to deficiencies in the annotation (such as reducing complex verb phrases to their one root verb). We believe that a more detailed annotation of tense would make result in a finer temporal ordering which would improve performance. Another area of concern is our very small corpus. Many empirical studies such as (Strube, 1998) and (Tetreault, 2001) have corpora of hundreds or even thousands of annotated sentences. The larger and more varied the corpus, the more reliable the results. We also acknowledge the fact that automating the annotation of temporal relations is complicated task all to itself and that it is an area of future research.

Recent work on this corpus has looked into the effects of breaking up conjoined utterances on reference resolution as suggested by (Kameyama, 1998). and implemented by (Strube, 1998). We found that this simple metric improved scores for all implicit roles (without using temporal reasoning) as well as for pronouns in another corpus (Tetreault, 2001). We tested temporal reasoning with the utterances broken apart and found it did not improve the score any higher.

Currently, we are annotating a much larger corpus of a similar domain (emergency rescue planning for a city). We hope that using this new data will address the problems discussed above.

In short, preliminary results indicate that temporal reasoning could be useful in reference resolution, but a better annotation scheme and a larger corpus are needed to strengthen this claim.

7. Acknowledgments

I would like to thank James Allen for many discussions on implicit role reference and tense. I am also grateful to Mark Core and Donna Byron for their comments. Partial

support for the research reported in this paper was provided by the DARPA research grant no. F30602-98-2-0133 to and the ONR grant no. N00014-01-1-1015, both to the University of Rochester.

8. References

Allen, Byron, Dzikovska, Ferguson, Galescu, and Stent. 2000. An architecture for a generic dialogue shell. *NLENG: Natural Language Engineering, Cambridge University Press*, 6.

Asher, Nicholas and Alex Lascarides. 1999. Bridging. *Journal of Semantics*, 15:83–113.

Byron, D. and J. Allen. 1998. Resolving demonstrative pronouns in the TRAINS93 corpus. pages 68 – 81.

Heeman, P. and J. Allen. 1994. The TRAINS93 dialogues. Technical Report TRAINS TN 94-2, University of Rochester.

Hwang, Chung Hee and Lenhart K. Schubert. 1992. Tense trees as the "fine structure" of discourse. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics*.

Kameyama, Megumi. 1998. Intrasentential centering: A case study. In *Centering Theory in Discourse*.

Poesio, M. 1994. Definite descriptions, focus shift and a theory of discourse interpretation. In *In Proceedings of the Conference on Focus in Natural Language Processing*.

Strube, Michael. 1998. Never look back: An alternative to centering. In *Association for Computational Lingusitics*, pages 1251–1257.

Tetreault, Joel R. 2001. A corpus-based evaluation of centering and pronoun resolution. *Computational Lingusitics*, 27(4).

Tetreault, Joel R. 2002. Implicit role reference. Unpublished.

Webber, B. L. 1988a. Discourse deixis: Reference to discourse segments. In *Proc. of the 26th ACL*, pages 113–122.

Webber, B. L. 1988b. Tense as discourse anaphora. *Computational Linguistics*.

The Workshop Programme

8:00 – 8:15 Welcome

8:15 – 8:45 I. Alegria, M. Aranzabe, A. Ezeiza, N. Ezeiza, R. Urizar
Robustness and customization in an analyzer/lemmatizer for Basque

8:45 – 9:15 M. O. Dzikovska, James F. Allen, Mary D. Swift
Finding the balance between generic and domain-specific knowledge: a parser customization strategy

9 :15 – 9 :45 David M. de Matos, Nuno J. Marnede
Data-driven application configuration

9 :45 – 10 :00 Break

10 :00 – 10:30 Svetlana Sheremetyva, Alexsei Pervuchin, Vladislav Trotsenko, Alexej Tkachev
Towards Saving on Software Customization

10:30 – 11:00 Maria Nava
Resource integration and customization for automatic hypertext information retrieval in a corporate setting

11:00 – 11:30 Patrice Lopez, Christine Fay-Vanier and Azim Roussanaly
Lexicalized Grammar Specialization for Restricted Applicative Languages

11:30 – 11:45 Break

11:45 – 12:15 Hurskainen Arvi
A Versatile Knowledge Management Package

12:15 – 12:45 Remi Zajac
Challenges in MT customization on closed and open text styles

12:45 – 13:15 Anju Saxena and Lars Borin
Locating and reusing sundry NLP flotsam in an e-learning application

13:15 – 13:30 Closing discussion

Workshop Organisers

Federica Busa	The Net Planet, S.p.A.
Robert Knippen	LingoMotors, Inc.
Evelyne Viegas	Microsoft Corporation
Antonio Sanfilippo	Sra International

Workshop Programme Committee

Saliha Azzam	Microsoft Corporation
Federica Busa	The Net Planet, S.p.A.
Robert Knippen	LingoMotors Inc.
Connie Parkes	Dictaphone
Antonio Sanfilippo	Sra International
Evelyne Viegas	Microsoft Corporation
Piek Vossen	Irion Technologies
Remi Zajac	Systran Corporation

Table of Contents

Author Index

Robustness and customisation in an analyser/lemmatiser for Basque

Alegria I., Aranzabe M., Ezeiza A., Ezeiza N., Urizar R.

Informatika Fakultatea
649 P.K. E-20080 Donostia. Basque Country.
{i.alegria, jipecran}@si.ehu.es

Abstract

This paper describes the work carried out to improve the robustness of the morphological analyser/generator for Basque which can be adapted to several domains and variants of the language. This analyser is used as a lemmatiser in several IR applications such as an Intranet search engine.

We present an enhanced analyser that deals not only with standard words but also with linguistic variants (including dialectal variants and competence errors) and words, whose lemmas are not included in the lexicon, by relaxing the constraints of the standard analyser. In addition to this, a user's lexicon can be added to the system in order to customise the tool. This user's lexicon can be obtained by means of a semiautomatic process.

1. Introduction

The starting point of this research is a general morphological analyser/generator described in (Alegria et al., 1996), which reported 95% of coverage. This poor result was due (at least partially) to the recent standardisation and the widespread dialectal use of Basque.

Although in some systems lemmas corresponding to unknown words are included in the main lexicon in a previous step, this solution is not satisfactory if we want to build a flexible system. We decided that it was necessary to manage a user's lexicon, for linguistic variants and forms whose lemmas were not in the lexicon, if we wanted to develop a comprehensive or adapted analyser.

However, the enhancement of coverage leads, in some cases, to produce overgeneration, and, consequently, to increase ambiguity. Although this ambiguity is not real, it causes poor results (lower precision) in applications based on morphology or lemmatisation. Another important issue was the improvement of precision. We studied the results of the analyser and saw that most errors (50%-75%) were made when dealing with proper names. Therefore, we propose some solutions to avoid about 50% of the errors.

2. Architecture of the morphological analyser

Morfeus is a robust morphological analyser for Basque. It is a basic tool for current and future work on NLP of Basque. Some of the tools based on it are a tagger (Ezeiza et al., 1998), an Intranet search engine (Aizpurua et al., 2000) and an assistant for verse making (et al., 2001).

The analyser is based on the two-level formalism. The two-level model of computational morphology was proposed by Koskenniemi (Koskenniemi, 1983) and has had widespread acceptance due mostly to its general applicability, declarativeness of rules and clear separation of linguistic knowledge and program.

This tool is implemented using lexical transducers. A lexical transducer (Karttunen, 1994) is a finite-state automaton that maps inflected surface forms to lexical forms, and can be considered an evolution of the two-level morphology. The tool used for the implementation is the *fst* library of *Inxight*[1] (Karttunen and Bessley, 1992; Karttunen, 1993; Karttunen et al., 1996). A detailed description of the transducers can be found in (Alegria et al., 2001).

We have defined the architecture of the analyser using three main modules (Schiller (Schiller, 1996) and others propose only two levels):

1. The standard analyser that uses a general lexicon and a user's lexicons. This module is able to analyse/ generate standard language word-forms. In our applications for Basque we defined about 75,000 entries in the general lexicon, more than 130 patterns of morphotactics and two rule systems in cascade, the first one for long-distance dependencies among morphemes and the second for morphophonological changes. The three elements are compiled together in the standard transducer. To deal with the user's lexicon the general transducer described below is used.
2. The analysis and normalization of linguistic variants (dialectal uses and competence errors). Due to non-standard or dialectal uses of the language and competence errors, the standard morphology is not enough to offer good results when analysing real text corpora. This problem becomes critical in languages like Basque in which standardisation is in process and dialectal forms are still of widespread use. For this process the standard transducer is extended with new lexical entries and phonological rules producing the *enhanced transducer*.
3. The guesser or analyser of words without lemmas in the lexicons. In this case the standard transducer is simplified removing the lexical entries in open categories (nouns, adjectives, verbs, …), which constitute the vast majority of the entries, and is substituted by a general automata to describe any combination of characters. So, the *general transducer* is produced combining this general set of lemmas with affixes related to open categories and general rules.

[1] Inxight Software, Inc., a Xerox Enterprise Company (www.inxight.com)

The analyser of non-standard words (steps 2 and 3) may sometimes produce overgeneration, and it is important to reduce this ambiguity as soon as possible.

3. Customizing the analyser

In order to deal with unknown words, a general transducer has been designed to relax the need of lemmas in the lexicon. This transducer was initially (Alegria et al., 1997) based on an idea used in a speech synthesis system (Black et al., 1991) but it has been now simplified. Daciuk (Daciuk, 2000) proposes a similar way when he describes the *guessing automaton*, but the construction of our automaton is simpler.

The new transducer is the standard one modified in this way: the lexicon is reduced to affixes corresponding to open categories and generic lemmas for each open class, while standard rules remain. There are seven open classes and the most important ones are: common nouns, personal names, place nouns, adjectives and lexical verbs. Grammatical categories and semantic ones (personal names or place names) are separated because they have different declension.

So, the standard rule-system is composed of a mini-lexicon where the generic lemmas are obtained as a result of combining alphabetical characters and can be expressed in the lexicon as a cyclic sublexicon with the set of letters (some constraints are used with capital/non-capital letters according to the part of speech). In fig. 1 the graph corresponding to the mini-lexicon is shown.

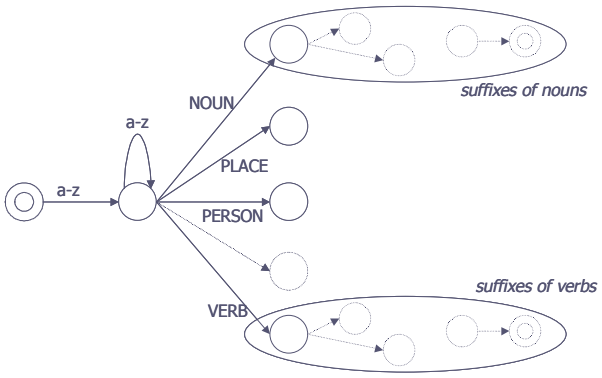


Figure 1. Simplified graph of the mini-lexicon

This transducer is used in two steps of the analysis:

1. in the standard analysis, in order to analyse declension and derivation of lemmas in the user's lexicon.
2. in the analysis without lexicon (called *guesser* in taggers).

The user's lexicon is composed of a list of lemmas along with their parts of speech defined by the users. The general transducer suggests possible interpretations of the word, and these lemmas are searched in the user's lexicon. When any lemma and class given by the general transducer matches the information on the user's lexicon, the analyser selects the corresponding interpretation and gives it as a result.

So, the user's lexicon is an editable resource which can be inferred from corpora or be managed on-line by the user. The use of this lexicon combined with the general transducer allows to customise the applications and it has been included successfully in three tools:

1. A spelling corrector for Basque (Aldezabal et al., 1999) in which for each lemma included in the user's lexicon any inflected form or derivative is accepted.
2. An Intranet search engine (Aizpurua et al., 2000) in which lemmatisation plays an important role and which can be customised when adapted to a special domain. In this case a semiautomatic process is carried out. First, the whole analyser (in the three steps above mentioned) is used to analyse a big corpus and the possible lemmas obtained by the guesser. After being sorted by frequency, they are presented to the user in order to include them in the user's lexicon[2]. The site www.zientzia.net, devoted to scientific documents, was built in this way.
3. A general part-of-speech tagger including customisation similar to the search engine.

4. Increasing coverage

The analyser was designed with the main objective of being robust, that is, capable of treating both standard and non-standard forms in real texts. For this reason, the morphological analyser has been extended in two ways:

1. The treatment of linguistic variants (dialectal variants and competence errors) (Aduriz *et al.*, 1994)
2. A two-level mechanism for lemmatisation without lexicon to deal with unknown words, which has been explained above

Important features of this design are homogeneity, modularity and reusability because the different steps are based on lexical transducers, far from *ad hoc* solutions, and these elements can be used in different tools. This could be considered a variant of constraint relaxation techniques used in syntax (Stede, 1992), where the first constraint demands standard language, the second one combines standard and linguistic variants, and the third step allows free lemmas in open categories. Only if the previous steps fail, the results of the next step are included in the output. Oflazer also uses relaxation techniques in morphology (Oflazer, 1996).

With this design the obtained coverage is 100% and precision over 99.5%. The ambiguity measures of the morphological analyser, taken from a balanced corpus of about 27,000 tokens and from a news collection of about 9,000, are shown in table 1. These measures have been obtained using all the morphological features.

Ambiguity Rate	Interpretations per ambiguous token	Interpretations per token
66.95%	4.38	3.26

Table 1: Ambiguity measures[3]

However, sometimes overgeneration is produced in order to improve robustness. Overgeneration increases ambiguity but often this ambiguity is not real and causes poor results (low precision) in applications based on morphology such as spelling correction, morphological generation or tagging.

[2] At this moment it is a not friendly off-line process

[3] Ambiguity Rate: #ambiguous_token / #token; Interpretations per token: #analyses / #token; Interpretations per ambiguous token: #analyses_ambiguous_token / # ambiguous_token

	Distribution	Ambiguity Rate	Interpretations per ambiguous token	Interpretations per token	Precision
standard	77.90%	80.73%	3.81	3.27	99.73%
variant	1.75%	80.53%	4.23	3.60	92.31%
unknown	2.65%	99.79%	18.05	18.01	98.12%
average	100.00%	66.95%	4.38	3.26	99.61%

Table 2: Ambiguity measures in the output of the analyser

	tokens	standard	variant	unknown	other[4]
corpus1	116,720	76.66%	1.02%	3.28%	19.04%
corpus2	1,288,257	78.44%	0.94%	3.80%	16.82%
corpus3	587,515	74.98%	2.03%	2.92%	20.07%
corpus4	33,232	77.32%	1.42%	4.92%	16.34%
corpus5	148,333	77.91%	1.01%	6.23%	14.85%
corpus6	29,939	60.54%	11.50%	7.90%	20.06%

Table 3: Distribution of tokens in different types of corpora

5. Decreasing ambiguity

The ambiguity for linguistic variants and unknown words is higher and the precision measures are poorer, but they form a small group of the input words (5%-10%) and the influence on average results is not significant.

The morphological analyser may sometimes overgenerate analyses of linguistic variants and unknown lemmas (table 2). Even if most words in texts are analysed in the first phase (see table 3), the small proportion of non-standard words constitutes a great amount of the superfluous interpretations. Yet, the rate of non-standard words varies depending on the type of corpus.

For instance, corpus3 is a balanced corpus with a high rate of standard Basque texts. On the contrary, corpus6 is a subset of texts from corpus3 written mainly in two dialects. Obviously, this corpus has a higher rate of non-standard uses. Corpus1 is a compilation of texts from the Web, and, generally, there is a trend to write these documents following standard rules of the language. Finally, corpus2, corpus4 and corpus5 are texts from the Basque newspaper *Euskaldunon Egunkaria*, and, even if the language variant used on them is standard, there is a relatively high amount of unknown words.

The treatment of non-standard words has been added to the previously developed analyser for two main reasons:

1. The average number of interpretations in non-standard words is significantly higher than in standard words (see table 2).
2. There could be multiple lemmas for the same or similar morphological analysis. This is a problem when we want to build a lemmatiser. For example, if *bitaminiko* (vitaminic) is not in the lexicon the results of the analysis of *bitaminikoaren* (from the vitaminic) as adjective can be multiple: *bitamini+ko+aren*, *bitaminiko+aren* and *bitaminikoaren*, but the only right analysis is the second one.

We think that it is important to reduce the ambiguity at this stage, so that the input of subsequent processes is more precise. But, we do not use information about surrounding words because a tagger will be used later. The process is limited to the word we want to treat, and we only need to know, in some cases, if the previous token was a full stop.

This module consists of different methods for linguistic variants and unknown words, because overgeneration is produced by different facts in each case, as will be described below.

5.1. Disambiguation of linguistic variants

In the case of linguistic variants a heuristic tries to select the lemma that is "nearest" to the standard one according to the number of non-standard morphemes and rules applied. It chooses the interpretation that has less non-standard uses for each POS tag.

For example, analysing the word-form *kaletikan* (dialectal form) two possible analyses are obtained: *kale+tik* (from the street) and *kala+tik* (from the cove). Both analyses have a non-standard morpheme (*-tikan*) but the first analysis is more probable because it applies no other transformation rule and to obtain the second one it has been necessary to apply another rule at the end of the lemma to transform *kale* into *kala*.

Thus, we must decide which of the analyses need to be selected or discarded based on the amount of transformation rules applied to obtain each analysis, but the enhanced transducer does not detail this information. The output of the enhanced transducer displays the normalised lemma/morphemes along with their corresponding morphological features. In the case of non-standard morphemes linked in the lexical database to their normalised form, the analysis details both normalised and variant morphemes.

Thus, the procedure uses these results to select the most probable lemmas for each POS tag. The results of applying this procedure are shown in table 4. The error rate of the procedure is 1.7%, so the error rate added to the whole process is 0.03%. It does not mean a significant drop in overall ambiguity, but it discards 40% of superfluous analyses.

[4] This group represents punctuation marks and other symbols.

	Ambiguity Rate	Interpretations per ambiguous token	Interpretations per token	Precision
before	80.53%	4.23	3.60	92.31%
after	75.35%	2.98	2.49	90.42%

Table 4: Ambiguity measures on linguistic variants before and after the procedure

	Ambiguity Rate	Interpretations per ambiguous token	Interpretations per token	Precision
initial	99.79%	18.06	18.01	98.12%
typographical	99.58%	8.18	8.15	96.46%
derivational	99.58%	7.94	7.91	96.46%
proper names	85.21%	6.93	6.05	95.94%
statistical 3+2+1	83.33%	3.99	3.49	91.98%

Table 5. Ambiguity measures on unknown words using all the procedures

	Distribution	Ambiguity Rate	Interpretations per ambiguous token	Interpretations per token	Precision
standard	77.90%	80.73%	3.81	3.27	99.73%
variant	1.75%	75.35%	2.98	2.49	90.42%
unknown	2.65%	85.21%	4.06	3.61	93.02%
average	100.00%	66.46%	3.80	2.86	99.43%

Table 6. Ambiguity measures in the output of the improved analyser

However, this heuristic treats every rule equally, but not all of them have the same probability of being applied. We think that it could be interesting to use a probabilistic transducer (Mohri, 1997) to improve the precision measures of both the analyser and the disambiguation procedure of variants.

5.2. Disambiguation of unknown words

We have tested several procedures to detect and treat unknown words using different criteria:

1. Typographical disambiguation. Some analyses are discarded based on capital letters.
2. Disambiguation of derivational words to counterbalance overgeneration of the analyser. The goal of this procedure is to discard one of several interpretations when the morphological analyser assigns analyses as derivational and non-derivational word.
3. Identification and disambiguation of proper names not included in the lexicon. Some analyses can be disambiguated when identical lemmas are found in the same document.
4. Disambiguation based on both statistical and linguistic information. These statistics relates final trigrams of characters and POS tags. is used. The main features of the heuristic are: a) for each POS tag, leave at least one interpretation; b) assign a weight to each lemma according to the final trigram and the POS tag; c) select the lemma according to its length and weight –best combination of high weight and short lemma.

These procedures were designed to be applied consecutively. To decide the order in which they must be applied, we tried different combinations.

Finally, table 5 shows the best result of applying all the procedures in cascade.

This treatment has been designed to discard some of the interpretations of unknown words. Even if unknown words are only 2%-3% of the words, they constitute 15%-20% of the analyses. After applying the procedures, they only represent 3%-4.5% of the analyses, depending on the combination of procedures we use, and the average number of interpretations decreases from 18-19 down to 3,5-4,5. The overall results of treating the reference text are shown in table 8. This has been measured using the second level tagset both for disambiguation of linguistic variants and for statistical disambiguation of unknown words, thus leaving (at least) one lemma per class and subclass.

Precision decreases in average around 0.2%, even if the results for unknown words fall from 98% to 93%. Finally, we want to point out that each combination of the procedures may be used for different applications.

6. Improving precision

The main reason for these errors is the incremental architecture of the analyser. The first step in the process, the standard analyser, causes wrong interpretations, primarily when very short or very rare lemmas are involved in the analysis. However, the process stops when the analyser finds (at least) one interpretation of the word.

A clear example of these misinterpretations is *Barak*. This name, when it appears in its base form, is interpreted as *bara*, a common noun of very low frequency. When it appears inflected, i.e. *Barak-ek* (*Barak* in ergative case), the standard analyser assigns no interpretation and the analyser without lexicon interprets it correctly as a proper noun.

	Distribution	Ambiguity Rate	Interpretations per ambiguous token	Interpretations per token	Precision
standard	77.88%	81.02%	3.86	3.32	99.88%
variant	1.66%	81.36%	4.40	3.76	96.51%
unknown	2.76%	99.90%	18.20	18.18	98.34%
average	100.00%	67.21%	4.46	3.32	99.80%

Table 7: Ambiguity measures in the output of the analyser

Most of the errors are avoidable enriching the user's lexicon, but it is necessary to improve the results when this is not done.

So we must avoid rare and improbable analyses when a word has an initial capital letter. In order to avoid odd analyses we have marked short or conflicting lemmas with low probability as rare in the lexical database. Using this information, when all the possible interpretations for a word are marked as rare, the process follows using the next module. If at the next step the analyser does not find a non-rare analysis for the word, the word will be tagged just as the standard analyser did.

In the case of low frequency lemmas, words written with initial capital letter are also analysed by the guesser and only proper name interpretations are added to the ones suggested by the standard analyser.

In order to increase the precision in the analyser of linguistic variants, we limit the number of rules applied to obtain the interpretations. If all the interpretations have been obtained applying a higher value of rules than the threshold, the word will be treated using the guesser, thus, discarding the other interpretations.

We have implemented these proposals and the results are encouraging (see table 7). As a result, we have avoided 50% of the errors relaxing the constraints of the morphological analyser.

7. Conclusions

We have presented the work carried out to improve the robustness of a morphological analyser and to adapt it to new domains. We have made a proposal for the architecture of a morphological analyser combining different transducers to increase flexibility, coverage and precision. The design we propose is quite new as far as we know and we think that our design could be interesting for the robust treatment of other languages.

On the other hand, we have also defined some local disambiguation procedures, which don't take into account the context of the word, so as to discard many of the overgenerated analysis for non-standard words. The results of the research are very encouraging.

8. Acknowledgements

This work has been partial supported by the Education Department of the Government of the Basque Country **(**UE1999-2**)** and the Spanish Science and Technology Ministry (*Hermes* research project; 8/DG00141.226-14247/200).

We would like to thank to Xerox for letting us using their tools, and also to Ken Beesley and Lauri Karttunen for their help.

9. References

Aduriz I., I.Alegria, J. M. Arriola, X. Artola, A. Díaz de Ilarraza, N. Ezeiza, K. Gojenola and M. Maritxalar 1995. Different issues in the design of a lemmatizer/tagger for Basque. *From text to tag SIGDAT, EACL Workshop*.

Aizpurua I., I. Alegria, N. Ezeiza, 2000. GaIn: un buscador Internet/Intranet avanzado para textos en euskera. *Actas del XVI Congreso de la SEPLN*.

Aldezabal I., I. Alegria, O. Ansa, J. Arriola, N. Ezeiza, 1999. Designing spelling correctors for inflected languages using lexical transducers. *Proceedings of EACL'99*, 265-266. Bergen, Norway. 8-12.

Alegria I., M. Aranzabe, A. Ezeiza, N. Ezeiza, R. Urizar, 2001. Using Finite State Technology in Natural Language Processing of Basque. *6th Conf. on Implementation and Applications of Automata. CIAA'2001.*

Alegria I., X. Artola, K. Sarasola, M. Urkia, 1996. Automatic morphological analysis of Basque. *Literary & Linguistic Computing Vol. 11, No. 4*: 193-203. Oxford University Press.

Antworth E.L. 1990. *PC-KIMMO: A two-level processor for morphological analysis*. Occasional Publications in Academic Computing, No. 16, Dallas, Texas.

Arrieta B., X. Arregi, I. Alegria, 2001. An Assistant Tool For Verse-Making In Basque Based On Two-Level Morphology. *Literary and Linguistic Computing, Vol. 16, No. 1, 2001.* Oxford University press.

Black A., J. van de Plassche, B. Williams, 1991. Analysis of Unknown Words through Morphological Descomposition. *Proceedings of 5th Conference of the EACL*, pp. 101-106.

Ezeiza N., I. Aduriz, I. Alegria, J. M Arriola, R. Urizar, 1998. Combining Stochastic and Rule-Based Methods for Disambiguation in Agglutinative Languages. *Proceedings of COLING-ACL'98*.

Karttunen L., 1993. Finite-State Lexicon Compiler. Xerox ISTL-NLTT-1993-04-02.

Karttunen L., 1994. Constructing Lexical Transducers, *Proceedings of COLING´94*, pp. 406-411.

Karttunen L., 2000. Applications of Finite-State Transducers in Natural Language Processing. *Proceedings of CIAA-2000.* Lecture Notes in Computer Science. Springer Verlag.

Karttunen L. and K. R. Beesley, 1992. *Two-Level Rule Compiler*. Technical Report Xerox ISTL-NLTT-1992-2.

Karttunen L., J.P. Chanod, G. Grenfenstette, A. Schiller, 1996. Regular Expressions for Language Engineering. *Natural Language Engineering, 2(4)*: 305:328.

Koskenniemi, K., 1983. *Two-level Morphology: A general Computational Model for Word-Form Recognition and Production*, University of Helsinki, Department of General Linguistics. Publications 11.

Mohri, M., 1997. Finite-state transducers in language and speech processing. *Computational Linguistics 23(2)*:269-322.

Oflazer K, C. Guzey, 1994. Spelling Correction in Agglutinative Languages. *Proceedings of ANLP-94*.

Oflazer K. 1996. Error-tolerant Finite State Recognition with Applications to Morphological Analysis and Spelling Correction. *Computational Linguistics 22(1)*: 73-89.

Schiller A., 1996. Multilingual finite-state noun phrase extraction. In Wo*rkshop on Extended finite state models of language*, ECAI'96, Budapest, Hungary.

Sproat R., 1992. *Morphology and Computation*. The MIT Press.

Stede M., 1992. The Search of Robustness in Natural Language Understanding. *Artificial Intelligence Review 6*, 383-414.

Finding the balance between generic and domain-specific knowledge: a parser customization strategy

Myroslava O. Dzikovska, James F. Allen, Mary D. Swift

Computer Science Department
University of Rochester
Rochester, NY, USA, 14627
{myros, james, swift}@cs.rochester.edu

Abstract

Adapting spoken dialogue systems across domains presents a challenge of finding the balance between wide-coverage parsers which can be easily ported but are slow and inaccurate, and domain-specific parsers which are fast and accurate but lack portability. We propose a method for customizing a wide-coverage, domain-independent parser to specific domains. We maintain a domain-independent ontology and define a set of mappings from it into a domain-specific knowledge representation. With this method, we customize the semantic representations output by the parser for reasoning, and we specialize the lexicon for the domain, resulting in substantial improvement in parsing speed and accuracy.

1. Introduction

Developers of spoken dialogue systems for multiple domains are faced with the challenge of finding the optimal balance between domain-independent and domain-specific parsers. There are wide-coverage parsers (e.g. XTag (Doran et al., 1994), LINGO (Copestake and Flickinger, 2000)) that are domain-independent and therefore easy to port to new domains, but they are often not efficient or accurate enough. The typical approach is to hand-craft parsers specifically for each domain (see for example (Goddeau et al., 1994)), but the performance gains in accuracy and efficiency are offset by their lack of portability, requiring additional effort to adapt them to new domains. We propose an alternative approach to address this challenge with a method for customizing a wide-coverage, domain-independent parser developed for spoken dialogue applications to specific domains. We maintain two ontologies: domain-independent for the parser, and domain-specific for the knowledge representation, and define a set of mappings between domain-specific knowledge sources and the semantic representations output by the parser. This method improves upon the generic parser output by specifically tailoring the semantic representations output by the parser for use by the reasoning components in the system. We also use the mappings to specialize the lexicon to the domain, resulting in substantial improvement in parsing speed and accuracy.

The customization method described here was developed in the process of adapting the TRIPS dialogue system (Allen et al., 2001) to several different domains, such as a transportation routing system (Allen et al., 1996) and a medication scheduling adviser. We assume a generic dialogue system architecture (Allen et al., 2000) that includes a speech module, a parser, an interpretation manager (responsible for contextual processing and dialogue management), and a back-end application responsible for the general problem-solving behavior of the system.

Adapting the spoken dialogue system across domains results in tension between the representation of generic vs. specific information in the ontology. To facilitate development when porting the parser to new domains, we want to retain the syntactic and semantic information that is consistent across domains. However, each domain comes with its own semantic information relevant to the application. For example, the representation of physical objects for the transportation domain requires specifying whether an object is suitable cargo for a transportation action, such as different types of food or supplies. In this respect, the distinctions between, say, oranges and potatoes are irrelevant, since they are equally good as cargo. These distinctions become highly relevant in the medical domain, where food-medicine interactions are important. Ideally, we want to customize the ontology to the domain for the most efficient reasoning. This becomes ever more important when using specialized reasoners with pre-defined input representations, for example, a database query system that must have specific template slots filled. Thus our goal is to preserve the language information that is similar across domains, while addressing specialization issues unique to each domain as much as possible, and keeping the development time spent on custom domain adaptation to a minimum.

To reuse the syntactic information, the AUTOSEM system(Rosé, 2000) uses a syntactic lexicon COMLEX(Macleod et al., 1994) as a source of syntactic information, and manually links subcategorization frames in the lexicon to the domain-specific knowledge representation. The linking is performed directly from syntactic arguments (e.g. subject, object ...) to the slots in a frame-like domain representation output by the parser and used by the reasoners. Rosé shows that her approach speeds up the development process for developing tutoring systems in multiple domains.

Our approach introduces an intermediate layer of abstraction, a generic ontology for the parser (the **LF Ontology**) that is linked to the lexicon and preserved across domains. The parser uses this ontology to supply meaning representations of the input speech to the interpretation manager, which handles contextual processing and dialogue management and interfaces with the back-end application. The domain-specific ontology used for reason-

ing (the **KR ontology**) is localized in the back-end application. We then customize the communication between the parser/interpretation manager and the back-end application via a set of mappings between the LF and KR ontologies. At the same time, the domain-independent ontology preserves semantic information consistent across domains that can be used by the Interpretation Manager for reasoning or reference resolution.

This separation allows us to write mappings in semantic terms without addressing the details of the grammar and subcategorization frames, using a higher level of abstraction. The developers writing the mappings does not need to understand details pertaining to syntax such as those included in COMLEX subcategorization frames, and can instead use descriptive labels assigned to generic semantic arguments (e.g. AGENT, THEME etc.). They can also take advantage of the hierarchical structure in the domain-independent ontology and write mappings that cover large classes of words. Finally, the mappings are used to convert the generic representation into the particular form utilized by the back-end application, either a frame-like structure or a predicate logic representation.

2. The Generic Lexicon

The LF ontology is close in structure to linguistic form, so it can be easily mapped to natural language and used in multiple domains. It classifies entities (i.e., objects, events or properties) primarily in terms of argument structure. Every LF type declares a set of linguistically motivated thematic arguments, a structure inspired by FRAMENET (Johnson and Fillmore, 2000), but which covers a number of areas where FRAMENET is incomplete, such as planning. We use the LF ontology in conjunction with a generic grammar covering a wide range of syntactic structures and requiring minimal changes between domains. For example, adapting the parser from the transportation to the medical domain required adding LF types for medical terms (our generic hierarchy was incomplete in this area) and corresponding vocabulary entries, but we did not need to change the grammar or existing lexical entries, and we continue to use the same lexicon in both domains.

The LF types in the LF ontology are organized in a single-inheritance hierarchy. Obviously, some sort of multiple inheritance is required, because, for example, a person is a living being, but also a solid physical object (as opposed to a formless substance such as water). We implement multiple inheritance via semantic feature vectors associated with each LF type. The features correspond to basic meaning components and are based on the EuroWordNet (Vossen, 1997) feature system with some additional features we have found useful across domains. While the same distinctions can be represented in a multiple inheritance hierarchy, a feature-based representation makes it easy to implement an efficient type-matching algorithm based on (Miller and Schubert, 1988). More importantly, using feature vectors allows us to easily change semantic information associated with a lexical entry, a property utilized during the customization process described below.

Word senses are treated as leaves of the semantic hierarchy. For every word sense in the lexicon, we specify the following information:

- Syntactic features such as agreement, morphology, etc.;
- LF type;
- The subcategorization frame and syntax-semantics mappings.

To illustrate, consider the verb *load* in the sense *to fill the container*. The LF type definition for *LF_LOAD* is shown in Figure 1. It specifies generic type restrictions on the arguments which are then propagated in the lexical entries. Intuitively, it defines a loading event in which an intentional being (AGENT) loads a movable object (THEME) into another physical object that can serve as a container (TO-LOC). The lexicon entry for *load* is linked to LF_Load and contains two possible mappings from the syntax to the LF: one in which the THEME is realized as direct object, corresponding to *load the oranges into the truck*, and another in which the THEME is realized as prepositional complement, corresponding to *load the truck with oranges*. The restrictions from the THEME argument are propagated into the lexicon, and the parser makes use of them as follows: only objects marked as *(mobility movable)* are accepted as a direct object or prepositional *with* complement of *load*.

```
(define-type LF_LOAD
 :sem (situation (aspect dynamic)
                 (cause agentive))
 :arguments
  (AGENT (phys-obj (intentional +)))
  (THEME (phys-obj (mobility movable)))
  (TO-LOC (phys-obj (container +)))
)
```

Figure 1: The LF type definition for LF_LOAD. In the lexicon, feature vectors from LF arguments are used to generate selectional restrictions based on mappings between subcategorization frames and LF arguments

The parser produces a flattened and unscoped logical form using reified events (Davidson, 1967). A simplified representation showing the semantic content of *Load the oranges into the truck* is shown in Figure 2. [1] For every entity, the full type is written as LF-parent*LF-form, where the LF-parent is the type defined in the LF ontology, and the LF-form is the canonical form associated with the word, for example, LF_VEHICLE*truck.

3. The KR customization

To produce domain-specific KR representations from the generic LF representations, we developed a method to customize parser output. The current system supports two knowledge representation formalisms often used by reasoners: a frame-like formalism where types have named

[1] For simplicity, we ignore speech act information in our representations

(TYPE e LF_LOAD*load)
(AGENT e *YOU*) (THEME e v1) (TO-LOC e v2)
(TYPE v1 LF_FOOD*orange)
(TYPE v2 LF_VEHICLE*truck)

Figure 2: The LF representation of the sentence *load the oranges into the truck.*

```
(a)
(LF-to-frame-transform load-transform
       :pattern (LF_LOAD LOAD)
       :arguments (AGENT :ACTOR)
                  (THEME :CARGO)
                  (TO-LOC :VEHICLE))

(b) (define-class LOAD
     :isa ACTION
     :slots
       (:ACTOR AGENT)
       (:CARGO COMMODITY)
       (:VEHICLE (OR TRUCK HELICOPTER)))

(c) [LOAD
      :ACTOR [PERSON +YOU+]
      :CARGO [ORANGE V1]
      :VEHICLE [TRUCK V2]]
```

Figure 3: LF-to-frame-transform. (a) The transform for LF_LOAD type; (b) the definition of LOAD class that the transform maps into; (c) The KR frame that results from applying this transform to the load event representation in Figure 2.

slots, and a representation that has predicates with positional arguments. The KR ontology must have subtype support, and for the lexicon specialization process described in the next section, type restrictions on the arguments of frames/predicates, though it need not be so in the most general case.

We use two basic transform types to map generic representations produced by the parser into the KR representation: LF-to-frame-transforms, shown in Figure 3, and LF-to-predicate-transforms, shown in Figure 4.

The LF-to-frame transforms convert LF types into KR frame structures by specifying the KR frame that the LF type maps into, and how the arguments are transformed into the frame slots. These transforms can be simple and name the slot into which the value is placed, or more elaborate and specify the operator expression that is applied to the value. The LF-to-predicate transforms are used to convert the frame-like LF structures into predicates with positional arguments. They specify a KR predicate that an LF type maps into and the expression that is formed.

After the parser produces the logical form, the Interpretation Manager decides which transform to apply to a given LF with the following algorithm:

- Find all transforms that are consistent with the LF or its ancestors;

```
(a)
(LF-to-pred-transform load-transform
 :pattern (LF_LOAD
           (LOAD *AGENT *THEME *TO-LOC)
          ))

(b) (define-predicate LOAD
       :isa ACTION
       :arguments
       (1:AGENT 2:COMMODITY
        3:(OR TRUCK HELICOPTER)))

(c) (AND (LOAD +YOU+ V1 V2)
         (COMMODITY V1) (TRUCK V2))
```

Figure 4: LF-to-predicate-transform. (a) The transform for LF_LOAD type; (b) the definition of LOAD predicate that the transform maps into; (c) The KR formula that results from applying this transform to the load event representation in Figure 2.

- Select the most specific transform that applies, that is, the transform that uses only the roles realized in this particular LF representation, that has all obligatory mappings filled, and for which the types of the LF arguments are consistent with the type restrictions on the class arguments;
- If there are several candidates, choose the transform that uses the most specific LF, and, if there are several for the same LF, the transform that maps into the most specific KR class;
- Apply the transform to the LF type and all its arguments to produce the new representation.

For example, the parser produces the logical form in Figure 2 for *load the oranges into the truck*. The Interpretation Manager determines that the most specific transform consistent with the arguments is the `load-transform`. If the back-end reasoners use the frame representation, then we use an LF-to-frame transform and obtain the frame shown in Figure 3. Alternatively, a system using predicates with positional arguments as its representation uses an LF-to-predicate transform and obtains the (simplified) representation shown in Figure 4.

Our examples show the simplest versions of the transforms for exposition purposes. The actual implementation permits a variety of constructs that we cannot illustrate due to space limitations, including the application of operators to arguments, default transforms that apply to LF arguments if no mapping is specified in LF-to-frame transform, and the use of the lexical forms in transforms when the KR uses similar terms. For example, from the point of view of the language ontology, medication names have similar distributions across syntactic contexts, and therefore are represented as leaves under the LF_DRUG type, e.g. LF_DRUG*prozac, LF_DRUG*aspirin. The KR ontology makes pragmatic distinctions between them (e.g. prescription vs. over-the-counter medicines), but uses the names as

leaf types in the hierarchy. We can write a single template mapping for all LF_DRUG children that does the conversion based on the lexical form specified in the entry. This allows us to convert the generic representation produced by the parser to a representation that uses the concepts and formalism suited to the domain.

4. Specializing the lexicon

The KR customization described above can be implemented as a two-stage process with a generic grammar and lexicon and a post-processing stage. We also use the mappings to speed up the parsing and improve semantic disambiguation accuracy by integrating the domain-specific semantic information into the lexicon and grammar.

We pre-process every entry in the lexicon by determining all possible transforms that apply to its LF. For each transform, we create a new sense definition identical to the old generic definition plus a new feature *KR-TYPE* in the semantic vector. The value of *KR-type* is the KR ontology class that results from applying this transform to the entry. Thus, we obtain a (possibly larger) set of entries which specify the KR class to which they belong. We then propagate type information into the syntactic arguments, making tighter selectional restrictions in the lexicon. We also increase the preference values for the senses for which mappings were found. This allows us to control the parser search space better and obtain greater parsing speed and accuracy.

Consider the following example. Given the definition of the verb *load* and LF_Load in Figure 1, and the definitions in Figure 3, the algorithm proceeds as follows:

- As part of generating the lexical entry for the verb *load*, the system fetches the definition of *LF_load* and the semantic vectors for it and its arguments;
- Next, the system determines the applicable LF-to-frame-transform, `load-transform`;
- Based on the transform, *KR-type load* is added to the feature vector of *load*;
- Since the mapping specifies that the LF argument THEME maps to KR slot *CARGO*, and the class definition contains the restriction that cargo should be of class *COMMODITY*, *KR-type commodity* is added to the feature vector of the THEME argument. Similar transforms are applied to the rest of the arguments.

As a result, in the lexicon we obtain a new definition of *load* with 2 entries corresponding to the same two usages described in section 2., but with stricter selectional restrictions. Now suitable objects or prepositional complements of *load* must be not only movable, but also identified as belonging to class COMMODITY in our domain. Since similar transforms were applied to nouns, oranges, people and other cargoes will have a *KR-type* value that is a subtype of COMMODITY inserted in their semantic feature vectors.

As a result of the specialization process, the number of distinct lexical entries will increase because there is not a one-to-one correspondence between the LF and KR ontologies, and several transforms may apply to the same LF depending on the syntactic arguments that are filled. A new entry is created for every possible transform, but during parsing the selectional restrictions propagated into the entries will effectively select the correct definitions. The Interpretation Manager thus knows the correct KR types assigned to all entities in the logical form output by the parser and the corresponding transforms, and only needs to apply them to convert the LF expression into the form used by the back-end reasoners.

	Generic	Transportation	Medical
# of senses	1947	2028	1954
# of KR classes	-	228	182
# of mappings	-	113	95

Table 1: Some lexicon statistics in our system

	Transportation	Medical
# of sentences	200	34
Time specialized (sec)	4.35 (870)	2.5 (84)
Time generic (sec)	9.7(1944)	4.3 (146)
Errors specialized	24%(47)	24% (8)
Errors generic	32% (65)	47% (16)

Table 2: Average time per lattice and the sentence error rate for the grammar specialized by our method compared to our generic grammar. Numbers in parentheses denote the total time and error count for the test set.

Lexicon specialization considerably speeds up the parsing process. We conducted an evaluation comparing parsing speed and accuracy on two sets of 50-best speech lattices produced by our speech recognizer: 34 sentences in the medical domain and 200 sentences in the transportation domain. Table 1 describes the ontologies used in these domains. The results presented in Table 2 show that lexicon specialization considerably increases parsing speed and improves disambiguation accuracy. The times represent the average parsing time per lattice, and the errors are the number of cases in which the parser selected the incorrect word sequence out of the alternatives in the lattice [2].

At the same time, the amount of work involved in domain customization is relatively small. The generic lexicon and grammar stay essentially the same across domains, and a KR ontology must be defined for the use of back-end reasoners anyway. We need to write the transforms to connect the LF and KR ontologies, but as their number is small compared to the total number of sense entries in the lexicon and the number of words needed in every domain,

[2] We considered correct the choices where a different pronoun, an article or a tense form were substituted. For example *can I tell my doctor* and *could I tell my doctor* were considered equivalent for purposes of this evaluation. However, we counted as errors the equally grammatical substitutions that selected a different word sense, e.g. *drive the people* vs. *get the people*

this represents an improvement over hand-crafting custom lexicons for every domain.

5. Conclusion

The customization method presented here allows the use of a lexicon and grammar with generic syntactic and semantic representations for improved domain coverage and portability, while facilitating the specialization of the lexicon and the representation produced by the parser to the needs of a particular domain. With this method we can produce specialized grammars for more effi cient and accurate parsing, and allow the parser, in cooperation with Interpretation Manager, to produce semantic representations optimally suited for specifi c reasoners within the domain.

6. Acknowledgments

We would like to thank Carolyn Rosé for her feedback on this article.

This material is based upon work supported by the Offi ce of Naval Research under grant number N00014-01-1-1015 and the Defense Advanced Research Projects Agency under grant number F30602-98-2-0133. Any opinions, fi ndings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of ONR or DARPA.

7. References

J. F. Allen, B. W. Miller, E. K. Ringger, and T. Sikorski. 1996. A robust system for natural spoken dialogue. In *Proceedings of the 1996 Annual Meeting of the Association for Computational Linguistics (ACL'96)*.

Allen, Byron, Dzikovska, Ferguson, Galescu, and Stent. 2000. An architecture for a generic dialogue shell. *NLENG: Natural Language Engineering, Cambridge University Press*, 6.

J. Allen, D. Byron, M. Dzikovska, G. Ferguson, L. Galescu, and A. Stent. 2001. Towards conversational human-computer interaction. *AI Magazine*.

Ann Copestake and Dan Flickinger. 2000. An open source grammar development environment and broad-coverage English grammar using HPSG. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, Athens, Greece.

Donald Davidson. 1967. The logical form of action sentences. In Nicholas Rescher, editor, *The Logic of Decision and Action*, pages 81–95. University of Pittsburgh Press, Pittsburgh. Republished in Donald Davidson, *Essays on Actions and Events*, Oxford University Press, Oxford, 1980.

Christy Doran, Dania Egedi, Beth Ann Hockey, B. Srinivas, and Martin Zaidel. 1994. XTAG system – a wide coverage grammar for English. In *Proceedings of the 15th. International Conference on Computational Linguistics (COLING 94)*, volume II, pages 922–928, Kyoto, Japan.

D. Goddeau, E. Brill, J. Glass, C. Pao, M. Phillips, J. Polifroni, S. Seneff, and V. Zue. 1994. Galaxy: A human-language interface to on-line travel information. In *Proc. ICSLP '94*, pages 707–710, Yokohama, Japan, September. URL http://www.sls.lcs.mit.edu/ps/SLSps/icslp94/galaxy.ps.

Christopher Johnson and Charles J Fillmore. 2000. The framenet tagset for frame-semantic and syntactic coding of predicate-argument structure. In *Proceedings ANLP-NAACL 2000*, Seattle, WA.

C. Macleod, R. Grishman, and A. Meyers. 1994. Creating a common syntactic dictionary of English. In *SNLR: International Workshop on Sharable Natural Language Resources*, Nara, August.

Stephanie A. Miller and Lenhart K. Schubert. 1988. Using specialists to accelerate general reasoning. In Tom M. Smith, Reid G.; Mitchell, editor, *Proceedings of the 7th National Conference on Artificial Intelligence*, pages 161–165, St. Paul, MN, August. Morgan Kaufmann.

Carolyn Rosé. 2000. A framework for robust semantic interpretation. In *Proceedings 1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

P. Vossen. 1997. Eurowordnet: a multilingual database for information retrieval. In *Proceedings of the Delos workshop on Cross-language Information Retrieval*, March.

Data-driven application configuration

David M. de Matos and Nuno J. Mamede

L^{2}F/INESC-ID/IST - Spoken Language Systems Laboratory
Rua Alves Redol 9, 1000-029 Lisboa, Portugal
david.matos@inesc-id.pt, nuno.mamede@inesc-id.pt

Abstract

Constructing modular applications from existing parts is difficult if there are mismatches due to input or output semantic differences during module interconnection. In order to minimize the effort of building such applications, and also as a guideline for designing modular applications from scratch, we propose an architecture in which modules are able to interface with each other without having to be reprogrammed. The architecture can be completely described using a small number of concepts. These factors allow rapid application building and reconfiguration with minimal manual intervention, potentiating module reuse and reducing the effort invested in building new applications.

1. Introduction

When building modular applications, it is possible to use parts that have been constructed by third parties, that solve part of the global problem. While this way of work is desirable because it promotes reuse, reducing the global development effort, it is all but straightforward: in fact, integrating foreign modules is almost never a simple task. The integration effort may become so expensive that it may seem better to build everything from scratch.

Managing these architectures is, thus, a challenging task and their complexity can be a serious hurdle when trying to bring together different components. Although not limited to the group, this problem also occurs when building natural language processing (NLP) applications and on various levels: from file-format handling or network-level communication to interaction between modules in a large application.

Here, we are concerned primarily with the latter aspect, even though the discussion could be applied to other levels, e.g. communication issues in a distributed application. We consider such lower-level aspects transport issues, though, that may be dealt with separately. Thus, CORBA (OMG, nda) and similar architectures are not an issue, since what we are concerned with is the way modules within an application exchange data and how to describe the way they do it.

We have two goals: to define a uniform way for modules to produce/consume data; and to define a uniform module interoperability model. We intend for these aspects to be realized complementarily: the latter will be a consequence of the former. In aiming at reaching these goals we are also promoting reuse and easy construction/configuration, since we provide a way for describing module interfaces for use with existing resources.

This document is organized as follows: the data model is defined in section 2.; a working example is presented in section 3.; and, finally, some concluding remarks and directions for evolution are presented.

2. Model

This section presents the architectural model. The first part presents structural aspects; the second part details the data model; the third part deals with semantics; and the fourth part details the implied application specification.

2.1. Structural aspects

We consider modular applications in which the modules exchange data through connections between their ports. These objects as well as their properties and relationships are presented here.

Definition 1 (portsets) *Let $\mathbb{M}$ be the set of all modules in an application. For a module m, the following portsets are defined: $\mathbb{O}^m$ (all output ports); $\mathbb{I}^m$ (all input ports); and $\mathbb{P}^m = \mathbb{I}^m \cup \mathbb{O}^m$ (all ports). In addition, $\mathbb{I}^m \cap \mathbb{O}^m = \varnothing$. We use p_i^m to denote the i-th port of m (i ranges over the corresponding portset).*

The definition for connection, while still a structural aspect, is better presented below (see def. 6).

2.2. Data model

Definition 2 (unrestricted grammar) *Unrestricted grammars (Lewis and Papadimitriou, 1981, def. 5.2.1.) are quadruples $G = (V, \Sigma, R, S)$, where V is an alphabet; Σ is the set of terminal symbols ($\Sigma \subseteq V$); $(V - \Sigma)$ is the set of nonterminal symbols; S is the start symbol; and R is the set of rules (finite subset of $(V^*(V - \Sigma)V^*) \times V^*$). Direct derivation (eq. 1), derivation (eq. 2), and generated language (eq. 3) are defined as follows:*

$$u \underset{G}{\Rightarrow} v \text{ iff } w_1, w_2 \in V^*, (u', v') \in R, \quad u = w_1 u' w_2 \wedge v = w_1 v' w_2 \tag{1}$$

$$w_0 \underset{G}{\Rightarrow} w_1 \underset{G}{\Rightarrow} \cdots \underset{G}{\Rightarrow} w_n \Leftrightarrow w_0 \overset{*}{\underset{G}{\Rightarrow}} w_n \tag{2}$$

$$L(G) = \{w \mid w \in \Sigma^* \wedge S \overset{*}{\underset{G}{\Rightarrow}} w\} \tag{3}$$

Σ is the union of three disjoint sets (eq. 4): Σ_k, the keyword set – the vocabulary for data description; Σ_d, used for writing data items; and Σ_i, used for writing intrinsic syntactic elements.

$$\Sigma = \Sigma_k \cup \Sigma_d \cup \Sigma_i \tag{4}$$

Definition 3 (data grammar; type grammar) *Consider port p and two grammars (as in def. 2): $\check{G}(p)$ – for*

writing data (the down-turned mark refers to data grammar entities); and $\hat{G}(p)$ – for writing datatypes (the upturned mark refers to datatype grammar entities).

These grammars must share the keyword set (denoted by $\Xi(p)$ (eq. 5)) and must be such that entities belonging to $L(\check{G}(p))$ describe the datatypes of the entities belonging to $L(\hat{G}(p))$. Each of the former entities works as a third grammar restricting $\hat{G}(p)$: it is used to validate data written according to the lowermost-level grammar.

$$\check{\Sigma}_k(p) = \hat{\Sigma}_k(p) = \Xi(p) \tag{5}$$

Definition 4 (data; datatype; correctness; validity) *Consider port p: $dat(p) \in L(\check{G}(p))$ denotes the data at p. We define port datatype, $typ(p) \in L(\hat{G}(p))$, as a data type specification according to $L(\check{G}(p))$ and $L(\hat{G}(p))$ (the third-level entities mentioned before). The following relation exists between a datastream and its associated datatype:*

$$typ(p) \rightsquigarrow dat(p) \tag{6}$$

Data is correct if it belongs to the language generated by the associated grammar: $dat(p) \in L(\check{G}(p))$, by definition; but it may happen that $dat(q) \notin L(\check{G}(p))$ (for some other port q) – in this case, $dat(q)$ would be incorrect according to $\check{G}(p)$.

Data is valid if $typ(p) \rightsquigarrow dat(p)$, i.e., the data stream follows the datatype definition (besides respecting the underlying grammar's rules).

The complete discussion of $typ(p)$ would only be complete taking into account the semantics of $L(\hat{G})$, but that is out of the scope of this document.

Taking into account the definitions in this section, we now give an example. Consider port p and a data representation containing the following XML (W3C, 2001a) fragment:

$dat(p) =$

```
[...]
<class name="nc">dog</class>
[...]
```

Then the terminal symbol sets would be (at least):

$$\check{\Sigma}_i(p) = \{\texttt{<}, \texttt{>}, \texttt{=}, \texttt{/}, \texttt{"}\}$$
$$\check{\Sigma}_k(p) = \Xi(p) = \{\texttt{class}, \texttt{name}\}$$
$$\check{\Sigma}_d(p) = \{w \mid w \notin \check{\Sigma}_i(p) \cup \Xi(p)\}$$

Consider a datatype description, for the data representation above, of which the following XML Document Type Definition (DTD) fragment is a part:

$typ(p) =$

```
[...]
<!ELEMENT class (#PCDATA)>
<!ATTLIST class name CDATA #REQUIRED>
[...]
```

Then the terminal symbol sets would be (at least):

$$\hat{\Sigma}_i(p) = \{\texttt{<}, \texttt{>}, \texttt{!}, \#, \texttt{ELEMENT}, \texttt{PCDATA}, \texttt{ATTLIST}, \texttt{CDATA}, \texttt{REQUIRED}\}$$
$$\hat{\Sigma}_k(p) = \Xi(p) = \{\texttt{class}, \texttt{name}\}$$
$$\hat{\Sigma}_d(p) = \{w \mid w \notin \hat{\Sigma}_i(p) \cup \Xi(p)\}$$

Thus verifying the grammar pair selection conditions (def. 3 and eq. 5).

2.3. Semantic aspects

This section deals with semantic aspects and restrictions that have to be observed when handling connections.

Each module has sole control over its internal semantics, in particular, in what concerns data semantics (defined by the receiver).

Definition 5 (semantics) *Consider port p and some interpretation function I (defined by the module's inner semantics): $sem(p)$ denotes the semantics required at p for normal processing behavior; $sem(dat(p))$ represents the data stream's semantics at p: computed by I (eq. 7). The data stream's semantics must subsume the port's semantics (eq. 8).*

$$sem(dat(p)) = I(dat(p)) \tag{7}$$
$$sem(p) \sqsubseteq sem(dat(p)) \tag{8}$$

Although we have no way of knowing how a module will interpret a piece of data, we can still write the following relations if we consider $\mathbb{D}$, the function denoting its argument's domain, and $\equiv$ the usual identity operator:

$$[\, typ(p) \rightsquigarrow dat(p) \,] \Leftrightarrow [\, I(typ(p)) \equiv \mathbb{D}(I(dat(p))) \,] \tag{9}$$

and thus (from 7, 9, and $\mathbb{D}$'s definition):

$$sem(dat(p)) \in I(typ(p)) \tag{10}$$

Definition 6 (connection) *Consider modules m and n and ports $p_i^m \in \mathbb{O}^m$ and $q_j^n \in \mathbb{I}^n$. Let predicate $con(p_i^m, q_j^n)$ be true if a connection exists between the pair. In the semantics domain, the output port's semantics must subsume the input's, i.e., condition 11 must be met.*

$$sem(q_j^n) \sqsubseteq sem(p_i^m) \tag{11}$$

Definition 7 (semantics mapping function) *When establishing a connection between two ports, p_i^m and q_j^n, if $typ(p_i^m) \neq typ(q_j^n)$, we need a semantics mapping function, $\theta_{i,j}^{m,n}$, for translating semantics across the connection (eq. 12, but also eq. 13). Furthermore, for the ports to be connectable, the receiving port's semantics must be subsumed by a transformation of the semantics of the previous module's output (cond. 14).*

$$\theta_{i,j}^{m,n} : typ(p_i^m) \rightarrow typ(q_j^n) \tag{12}$$
$$\theta_{i,j}^{m,n} : L(\check{G}(p_i^m)) \rightarrow L(\check{G}(q_j^n)) \tag{13}$$
$$sem(dat(q_j^n)) \sqsubseteq sem(\theta_{i,j}^{m,n}(dat(p_i^m))) \tag{14}$$

It is impossible, however, to guarantee a correct translation in the semantics domain, since, ultimately, input semantics is defined by the data consumer: we approach semantics conversion through datatype-directed data conversion. Since this conversion uses outside information about the ontologies of both sender and receiver, $\theta_{i,j}^{m,n}$ cannot be automatically generated solely from the information available at each end. Nevertheless, $\theta_{i,j}^{m,n}$ can be defined extensionally for each $typ(p_i^m)$.

We assume that it is always the receiver's responsibility to convert the data, since the data producer may be unable to determine how its results will be used. In the current discussion, we will also assume that condition 14 always holds, either because $\theta_{i,j}^{m,n}$ can satisfy it or, if that is not the case, because missing data parts can be supplemented by defaults when computing $sem(q_j^n)$.

2.4. Specifying the application

The model above gives rise to a data-oriented module interconnection architecture in which modules send/receive information to/from each other through typed channels that are uniquely defined by the datatypes at each end-point and by the corresponding translation function.

Since the architecture is not concerned with the modules' inner semantics, all that is needed to describe it completely are the collections of port datatypes and translation functions associated with connected ports. These collections are represented, respectively, by T, the datatype matrix, and by Θ, the translation matrix.

The datatype matrix is defined for all modules and their ports. Entries that do not correspond to actual ports are empty.

$$\underset{m_i \in \mathbb{M}}{T} = \begin{bmatrix} typ(p_1^{m_1}) & \cdots & typ(p_1^{m_\mathcal{M}}) \\ \vdots & & \vdots \\ typ(p_\mathcal{P}^{m_1}) & \cdots & typ(p_\mathcal{P}^{m_\mathcal{M}}) \end{bmatrix} \tag{15}$$

$$\begin{gathered} \mathcal{M} \equiv \#\mathbb{M} \qquad \mathcal{P} \equiv \max_{m \in \mathbb{M}}(\#\mathbb{P}^m) \\ \forall_{m \in \mathbb{M}} \forall_{1 \le v \le \mathcal{P}},\ p_v^m \notin \mathbb{P}^m \Rightarrow typ(p_v^m) = \varnothing \end{gathered} \tag{16}$$

The translation matrix is defined for all connected ports: one function for each connection. In all other cases, Θ is undefined.

$$\Theta = \begin{cases} \theta_{i,j}^{m,n} & con(p_i^m, q_j^n) \quad \text{(see def. 6)} \\ \textit{undefined} & \text{otherwise} \end{cases} \tag{17}$$

3. A small example

This example simplifies the model in important ways: all data flowing between ports is represented in XML and all datatypes can be specified either using DTDs or XML Schemas (XSD) (W3C, 2001d). Thus, in principle, all mismatches are due to variations in the XML data type definitions.

$$\forall_{m \in \mathbb{M}} \forall_{p,q \in \mathbb{P}^m}, \check{G}(p) \equiv \check{G}(q), \hat{G}(p) \equiv \hat{G}(q) \tag{18}$$

unless (only keywords are different)

$$\forall_{m \in \mathbb{M}} \forall_{p,q \in \mathbb{P}^m}, typ(p) \neq typ(q) \Rightarrow \Xi(p) \neq \Xi(q) \tag{19}$$

In our example, all datatypes have been described using DTDs and all necessary $\theta_{i,j}^{m,n}$ functions have been specified by Extensible Style Sheet (XSL) (W3C, 2001b) templates. By specifying all DTDs and XSL templates, the application becomes completely defined from the point of view of its data exchange paths.

The rest of this section will particularize further each of these aspects.

3.1. The application

The example application performs syntactic analysis of natural language sentences (fig. 1).

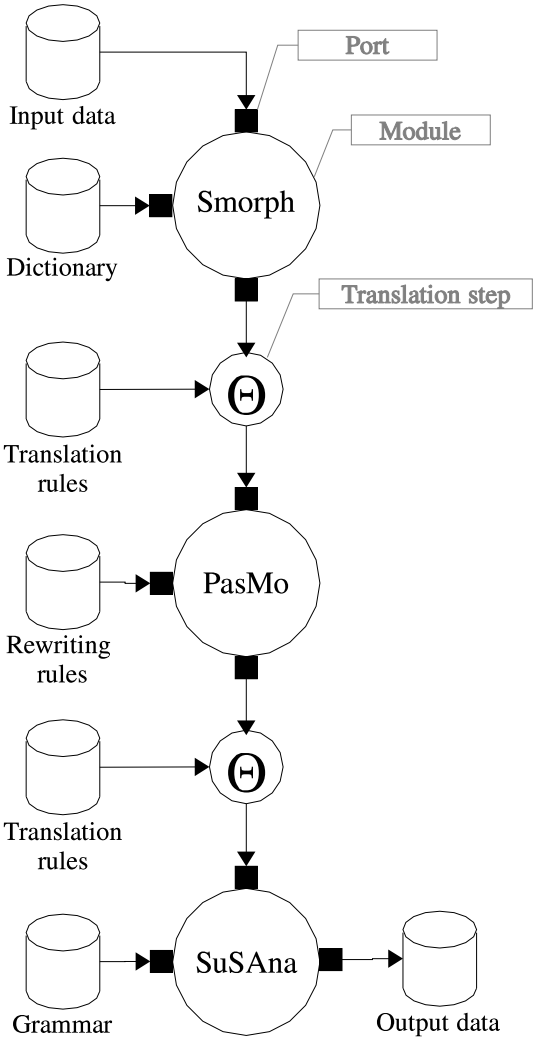


Figure 1: The example application.

The application consists of three modules: Smorph (Aït-Mokhtar, 1998) (morphological analyzer); PAsMo (Paulo, 2001) (rule-based rewriter); and SuSAna (Hagège, 2000; Batista, nd) (syntactic analyzer).

We consider only ports dealing with the data stream to be processed, thus ignoring those used for reading static data (such as dictionaries). Furthermore, in the following we will focus on the connection between Smorph and PAsMo, since the other relevant connection (that between PAsMo and SuSAna) is analogous.

3.2. The application ports

The relevant ports are Smorph's output (s) and PAsMo's input (p). To describe the data flowing through them, we need to specify just $typ(s)$ and $typ(p)$ (eq. 15 and figures 2 and 3). Smorph's output will be translated before being

```
<?xml version="1.0" encoding="iso-8859-15"?>
<!ELEMENT pasmo-in (word)*>
<!ELEMENT word     (class)*>
<!ATTLIST word  text CDATA #REQUIRED>
<!ELEMENT class    (flag)*>
<!ATTLIST class root CDATA #REQUIRED>
<!ELEMENT flag     EMPTY>
<!ATTLIST flag  name  NMTOKEN #REQUIRED>
<!ATTLIST flag  value CDATA   #REQUIRED>
```

Figure 2: DTD for PAsMo's input port, corresponding to $typ(p)$.

used by PAsMo. Note that Smorph's is a more expressive description (thus obeying condition 11), and that some information will be lost in the conversion (not a problem as long as condition 14 remains true).

```
<?xml version='1.0' encoding='iso-8859-1' ?>
<!ELEMENT smorph (item)*>
<!ELEMENT item (root)*>
<!ATTLIST item value CDATA #REQUIRED>
<!ELEMENT root (class)*>
<!ATTLIST root value CDATA #REQUIRED>
<!ELEMENT class (flags,flags)>
<!ATTLIST class type (0|mi) "0">
<!ELEMENT flags (flag)*>
<!ATTLIST flags level (1|2) #REQUIRED>
<!ELEMENT flag EMPTY>
<!ATTLIST flag name  NMTOKEN #REQUIRED>
<!ATTLIST flag value CDATA   #REQUIRED>
```

Figure 3: DTD for Smorph's output port, corresponding to $typ(s)$.

3.3. The translation step

The only relevant transformation, $\theta^{s,p}$, is the one mapping Smorph's output to PAsMo's input. It is implemented as a XSL transformation step and is completely specified by the set of XSL templates (figure 4) that map between data described according to Smorph's DTD and PAsMo's.

```
<?xml version='1.0' encoding='iso-8859-1' ?>
<xsl:stylesheet
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  version="1.0">
<xsl:output method="xml"
  encoding="iso-8859-15"
  doctype-system="pasmo-in.dtd"/>
<xsl:template match="/smorph">
  <pasmo-in>
    <xsl:apply-templates/>
  </pasmo-in>
</xsl:template>
<xsl:template match="item">
  <word text="{@value}">
    <xsl:apply-templates/>
  </word>
</xsl:template>
<xsl:template match="root">
  <class root="{@value}">
    <xsl:apply-templates
      select="class/flags/flag"/>
  </class>
</xsl:template>
<xsl:template match="flag">
  <flag name="{@name}" value="{@value}"/>
</xsl:template>
</xsl:stylesheet>
```

Figure 4: The translation specification in XSL, corresponding to $\theta^{s,p}$.

4. Related work

This work is related with several fields. The first is the field of data modeling, especially in what concerns very high-level modeling, such as the one done using UML (OMG, ndb). Specifications done in UML can be described using the XML Metadata Interchange (XMI) (OMG, 2002) specification that can then be used to specify the XSDs for the data being sent/received on a module's ports. This is useful because it allows us to describe graphically each module and its interconnections and, by extension, an entire application.

Since we plan on evolving in the direction of service specification(see sec. 5.), we have considered work in this area. One such is IBM's Web Services Flow Language (Leymann, 2001) which can be used for specifying multiple aspects of web services. This language is also layered on top of others: Web Services Description Language (WSDL) (W3C, 2001c) and Web Services Endpoint Language (WSEL) (Leymann, 2001). Although this structure closely parallels what we intend in our work, it has a different focus and does not invalidate our proposal.

The third area is that of communication systems, which typically define module interconnection architectures. An example is CORBA. Another, of particular interest for NLP, is the Galaxy Communicator (MIT, 2001; DAR, nd). This architecture is a distributed, message-based, hub-and-spoke infrastructure optimized for constructing spoken dialogue systems. It uses a plug-and-play approach that enables the combination of commercial and research components. It supports the creation of speech-enabled interfaces that scale gracefully across modalities. In this context, our proposal enables easy specification of Galaxy applications. At a different level, our specifications can be gracefully translated into hub scripts and server interface definitions.

In the context of reference architectures, such as the ones proposed by the TIPSTER (TIP, nd) or RAGS (ITRI, nd) projects, our model may prove useful in facilitating integration of external modules into the frameworks defined by those architectures. Note that, unlike most software infrastructures for Language Enginnering research and development, e.g. GATE (Cunningham et al., 1996), our model does not say anything about any module's function or impose any restrictions on their interfaces and is, thus, application- and domain-independent. This is so because the model is exclusively concerned with the data streams flowing between modules and the relations between their semantics at each end and not with the way each stream is used, i.e., the model is not directly concerned with application-related issues. In this sense, the model could be used to describe a kind of "smart glue" for use with other

architectures, e.g. in integration efforts of existing modules into GATE's CREOLE sets, or in datatype management.

Other application-development or intercommunication infrastructures may benefi t from using a high-level specifi - cation such as the one we propose here.

5. Conclusions and future directions

Our approach is useful for application development, since it focuses exclusively on the inputs and outputs of each module, without regard for module internals. This contributes to signifi cant dependency reductions, for the modules can be almost anything and run almost anywhere, as long as a communications channel (according to our restrictions) can be established between them.

We envision various directions for future work.

The fi rst is to provide higher-level service specifi cations on top of port descriptions. This would allow services to be defi ned using the descriptions of its inputs and outputs and, rather than exhaustively describing each port and its data, we would be able, at that higher abstraction level, to simply specify the name of the service. The rest would follow from lower-level descriptions.

Also, along the lines of higher-level abstractions and services, it would be interesting to try and specify automatic translation functions ($\theta_{i,j}^{m,n}$) based on service semantics. Of course, this would mean that semantics would have to be specifi ed in some way as well.

Both these approaches would help to integrate user-developed modules and help integrators to develop transformation steps that cannot be wholly automatically generated.

Another direction worth considering is the construction of module and application servers: modules or pre-built applications would be presented, e.g. via a web browser, enabling users to specify custom applications.

6. References

S. Aït-Mokhtar. 1998. *L'analyse présyntaxique en une seule étape*. Thèse de doctorat, Université Blaise Pascal, GRIL, Clermont-Ferrand.

Fernando Batista. n.d. Análise sintáctica de superfície e coerência de regras. Master's thesis, Instituto Superior Técnico, UTL, Lisboa.

Hamish Cunningham, Yorick Wilks, and Robert J. Gaizauskas. 1996. Gate – a general architecture for text engineering. In *Proceedings of the 16th Conference on Computational Linguistics (COLING96)*, Copenhagen.

DARPA, n.d. *DARPA Communicator*. See: www.darpa.mil/ito/research/com/index.html.

Caroline Hagège. 2000. *Analyse syntaxique automatique du portugais*. Thèse de doctorat, Université Blaise Pascal, GRIL, Clermont-Ferrand.

ITRI, nd. *RAGS – A Reference Architecture for Generation Systems*. Information Technology Research Institute, University of Brighton. See: http://www.itri.brighton.ac.uk/projects/rags/.

Harry R. Lewis and Christos H. Papadimitriou. 1981. *Elements of the Theory of Computation*. Prentice-Hall, Englewood Cliffs, NJ. ISBN 0-13-273426-5.

Frank Leymann, 2001. *Web Services Flow Language (WSFL 1.0)*. IBM Software Group, May. See also: xml.coverpages.org/wsfl.html.

MITRE Corporation, 2001. *DARPA Communicator*, May. See: http://fofoca.mitre.org/.

Object Management Group (OMG), 2002. *XML Metadata Interchange (XMI) Specification, v1.2*, January. See: www.omg.org/technology/documents/formal/xmi.htm.

Object Management Group (OMG), n.d.a. *Common Object Request Broker Architecture (CORBA)*. See: www.corba.org.

Object Management Group (OMG), n.d.b. *Unified Modelling Language*. See: www.uml.org.

Joana Lúcio Paulo. 2001. Aquisição automática de termos. Master's thesis, Instituto Superior Técnico, UTL, Lisboa.

NIST, nd. *TIPSTER Text Program*. See: http://www.itl.nist.gov/iaui/894.02/related_projects/tipster/.

World Wide Web Consortium (W3C), 2001a. *Extensible Markup Language*. See: www.w3c.org/XML.

World Wide Web Consortium (W3C), 2001b. *The Extensible Stylesheet Language*. See: www.w3.org/Style/XSL.

World Wide Web Consortium (W3C), 2001c. *Web Services Description Language (WSDL) 1.1*, March. See: www.w3.org/TR/wsdl.

World Wide Web Consortium (W3C), 2001d. *XML Schema*. See: www.w3c.org/XML/Schema and www.oasis-open.org/cover/schemas.html.

Towards Saving On Software Customization

Svetlana SHEREMETYVA
LanA Consulting
Madvigs Alle, 9, 2tv
1829DK Copenhagen Denmark
lanaconsult@mail.dk

Alexsei PERVUCHIN, Vladislav TROTSENKO, Alexej TKACHEV
Southern Ural State University, 76, Lenin av.
454080Chelyabinsk, Russia,
{perv,inter,tam @inf.susu.ac.ru}

Abstract

The paper suggests some ways to save on software customization when developing a family of NLP applications meeting specific domain and task requirements. The emphasis is made on the application architecture modularity and reusability of system components. A particular focus is set on an easy-to-use environment for linguist developers integrated with the architecture to facilitate the reuse and customization phase.

1 Introduction

In this paper we address the problem of rapid and low cost deployment of NLP systems by suggesting some ways to save on software customization. A wide range of literature can be found in the area of R&D of language processing software, whose goal is to facilitate future development efforts thus reducing customization cost. The issue of customization is closely related to reuse strategies and integration during development process (Prieto-Diaz, 1993; Thomas and Nejmeh, 1992). Constructing general-purpose tools that can be shared by the community is a popular topic of interest nowadays. Such tools are developed both for language acquisition and language processing. To name just a few one can mention GATE, - a tool for locating, loading and initializing components from local and non-local machines (see, e.g. Cunningham, 1999), an abstract model of thesauri and terminology maintenance OO framework (Fischer et al., 1996), grammar development environments integrated with sophisticated text-processing interfaces such as Boas acquisition tool for a quick ramp up of MT systems (see e.g., Sheremetyeva and Nirenburg, 2000), and the Advanced Language Engineering Platform,- a grammar development tool for high-level linguistic processing, (see, e.g. Bredenkamp and Henzte, 1995). It is also recognized that though many increasingly convivial, more widely distributed and hardware-independent applications softwares are currently available it is difficult to find the system that matches exactly the end-user requirements (Degoulet et al., 1994). It seems highly problematic to identify once and forever a particular locus to the dilemma of genericity versus specificity when speaking of genericity "in general" as applied to all kinds of applications. Indeed, the locus can be anything, - the system architecture, the application components, the language resources, etc. If, however, the concept of genericity is considered as applied to a family of applications, i.e., applications interleavingly sharing tasks and domains, one can probably suggest particular approaches to solve the problem. In this paper we attempt just that. The problem of customization can be considered from two perspectives: internal and external. Internal customization is responsible for improvements in a current application and for tailoring this application to the profile of a particular user. External customization refers to the effort and cost to "turn" a current application to a new one. We address these two aspects of customization by describing a cost-effective development of a specific application called AutoPat, - an application for authoring patent claims describing apparatuses in the English Language. A prototype of this application has been developed many years

ago (see, e.g., Sheremetyeva and Nirenburg, 1996) so that we shall not deal with specifications of the application but rather with a re-engineering issue. We focus here on the problem of components reusability and integration of a developer's toolkit into the application architecture. Our objective is

?? to describe a cost-effective migration from the old experimental version of AutoPat, that did not support a lot of functionalities to the AutoPat product;

?? to suggest ways to make improvements (and tailoring) in the current version of the application without extra programming effort;

?? to discuss effectiveness of AutoPat external customization to conceive and realize other specific applications of the same family.

The AutoPat "closest" family includes applications with any combination of the values of the following features:

?? Application type < authoring, machine translation, information retrieval, etc.>

?? Domain < patents with different subject matters <apparatus, method, process, etc.>

?? Document type <patent disclosure, claims>

?? Languages < English, Danish, Swedish, Norwegian, German, French, etc.>

In what follows we first present the context of the AutoPat application and the development environment. Then we shall overview reusable components and detail developers' (customization) tools. Our discussion will mainly address the improvement of the application as well as advantages of distributed environment to develop these kinds of applications.

2 System overview

AutoPat is an NLP application that consists of an interactive technical knowledge elicitation module with a sophisticated but easy-to-use interface at the user end, analysis module and fully automatic text generation module. The architecture of AutoPat with integrated development environment is given in Figure 1.

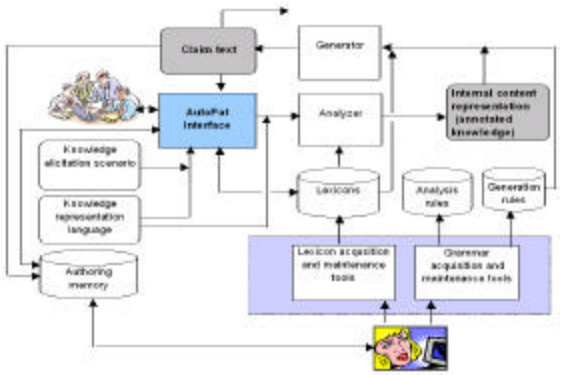


Figure 1. AutoPat overall architecture

Superficially, the architecture of our system conforms to the standard emerged in natural language generation, in that it includes the stages of content specification, text planning and surface generation (realization), as expressed, for instance in Reiter, (1994). However, there are some important differences. Unlike the typical content specification modules, our system relies on an authoring workstation environment equipped with a knowledge elicitation scenario for joint human-computer content specification. The latter starts with the user supplying natural language phrases into the system in the process of computer interview and results in the production of a content representation of a nascent claim.

A wide range of complex problems which are considered to be specific for generation may lead one to believe that generation is completely independent of analysis. This is not, however, the case in practice. The input to generation systems that is fed into the system directly by a user must first be somehow analyzed. This problem becomes especially important in those applications in which input to generation (as it is in our case) is in textual form. The stages of AutoPat processing are not strictly pipelined.The content specification interleaves with interactive semantico-syntactic analysis in that it assigns case-roles status to input phrases and memorizes their boundaries. The values of case-roles is then

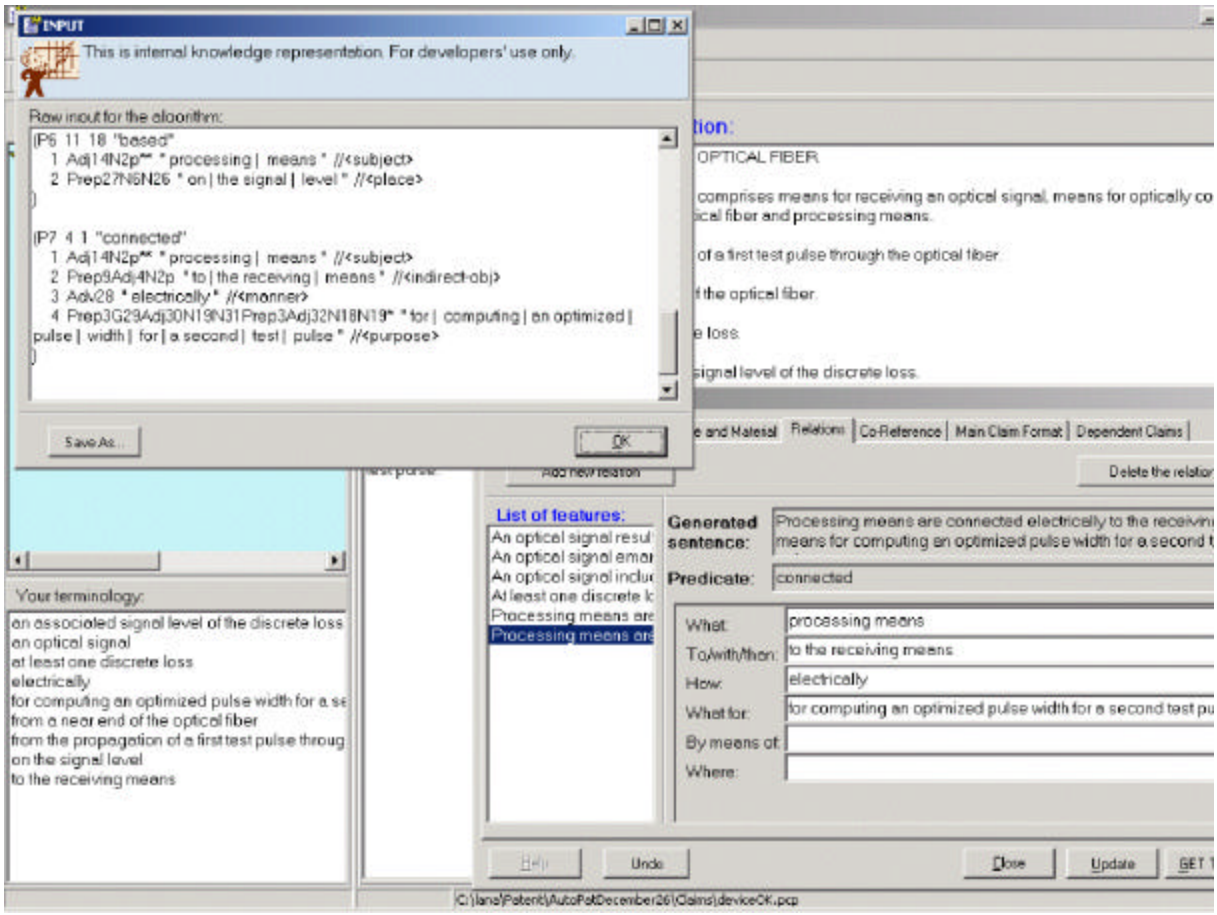


Figure 2. A screenshot of the AutoPat developer's interface window (top left corner) overlapping the user's interface for knowledge elicitation. It displays internal representation of a quantum of knowledge supplied by the user and processed by the application analyzer.

automatically unambiguously POS-tagged, assigned agreement features and interactively marked for coreference thus converting "raw" input into a shallow content representation, a claim draft. The draft is then submitted to an automatic text planner, which outputs a hierarchical structure of templates that is then input into linearizer and grammaticalizer to be converted into a legal claim text.

3 Development process: migration from experimental system to product.

3.1 Design

The first step in developing AutoPat was to define a subset of the experimental model for authoring patent claims that will be the basis of the application and the extension it will need to be turned into a product. The different functionalities of AutoPat application require, besides kernel components (such as the knowledge elicitation scenario, the knowledge representation language, lexicons, grammars and processing algorithms), a user-adaptive interface and linguistic knowledge acquisition tools to fight the well known problem of NLP applications, that of knowledge bottleneck. Developer's tools were integrated into the system architecture to facilitate the customization process and to make it cheaper.

3.2 Reuse and customization of existing components

The second step in the development process is the reuse of already developed components and their customization if they do not fit developer's needs.

3.2.1 Knowledge elicitation scenario

The knowledge elicitation scenario was almost fully reused in AutoPat, only one more step was added that of eliciting knowledge about dependent claims.

3.2.2 Knowledge representation language

Internal knowledge representation language was completely reused.

3.2.3 User Interface

The interface of the old system supported its main functions to model a professional behavior of a patent expert working with an inventor, - a knowledge elicitation interview, and to build internal content representation.

The content of the interview was almost fully reused in AutoPat, only one more step was added that of eliciting knowledge about dependent claims. Major customization dealt with lexicon acquisition functionalities and what might seem minor issues that in reality are very time consuming and thus affect the cost of application. The AutoPat interface is customized so as to support automation of tedious tasks such as typing, revising texts, making sure terminology is consistent, propagating changes through document, spell checking and lexicon acquisition. It has two different acquisition functionalities for predicate lexicon and for lexical units that fill predicate case-roles thus supporting two frames of their description (see Sections 3.2.5 and 4.2). The interface was also customized so as to better suite the user profile in terms of proficiency: the beginner has a chance to work in the Wizard Guide mode. It strictly guides the user through a step-by step procedure of describing essential features of invention and reuses experimental system interview procedure. An experienced user can work in the Professional mode that allows for more speed and flexibility when authoring a claim - the user may freely navigate among the stages of claim composition. Another new functionality allows the user to quit the program at any moment of elicitation session so that next time the user starts it s/he can resume the work where s/he left off.

3.2.3 Analyser

AutoPat reuses the architecture of the old system analyser, which consists of a submodule of interactive semantico-syntactic analysis and a submodule of automatic morphological analysis applied to the case-role fillers. The first analyser submodule is reused.

The submodule of morphological analysis is completely redone, as the old one was just a "toy" for feasibility study. Unfortunately we failed to incorporate to our system any of the described analysers as they proved to be either unavailable or not tuned to our domain. On the one hand, we tried to build our morphological analyser it in the most effort- and time-saving way. On the other hand, in view of other extensions of our AutoPat, such as multilingual generation, machine translation, information retrieval, etc., we decided to develop a reusable, possibly "extendable" morphological tool. As different types of applications need different depth of analysis our morphological submodule of the AutoPat analyzer features flexible sets of tags so that developer could vary the depth of analysis (see Sections 3.2.5 and 4.1). For example, for MT one might think of tags marking semantic information in addition to morphological, which for generation (i.e., for our immediate needs) is only necessary for predicates, but not for case-role fillers. (see Sections 3.2.5, 4.1 and 4.2). The AutoPat morphological analyser applies two levels of disambiguation procedure, one relies on context constraints, and another involves knowledge about case-roles. With one level of disambiguation it can be used as a stand-alone tool for any text from patent domain.

3.2.4 Generator

The upper level generation algorithm is reused. It consists of the same procedures: building a forest of predicate templates, linearization of this forest in a bracketed string of characters, and grammaticalization. The difference between the old version of the system and

AutoPat is that in the latter it is possible to customize algorithms at every generation step to improve the system output without extra programming effort (see Section 4.3).

3.2.5 Lexicons and grammar rules

The AutoPat lexicons are corpus-based and draw heavily on the sublanguage and on the needs of application. They include

a shallow lexicon of lexical units tagged with their class membership, which conveys morphological information (such as POS, number and inflection type) and semantic information, a concept, defining a word membership in a certain semantic class (such as object, process, substance, etc.). For example, the tag Nf shows that a word is a noun in singular (N), means a process (f), and does not end in –*ing*. This tag will be assigned, for example, to such words as *activation* or *alignment.* At present we use 23 tags that are combinations of 1 to 4 features out of a set of 19 semantic, morphological and syntactic features for 14 parts of speech. For example, the feature structure of noun tags is as follows:

```
Tag [ POS[Noun
        [object   [plural, singular]
         process [-ing, other[plural, singular]]
         substance [plural, singular]
         other      [plural, singular]]]]]
```

a deep (information-rich) lexicon of predicates. This lexicon is the main part of the AutoPat static knowledge and covers both the lexical and, crucially for our system, the syntactic and semantic knowledge. The structure of the entries of this lexicon was reused, the vocabulary was greatly enlarged. *Grammar rules* are updated (see section 4.3).

4 AutoPat Development tools

All developer's tools, including lexicon acquisition tools, have interfaces, which, on the one hand prompt acquirers to encode all the necessary features and, on the other hand, do not let them to add anything that is not relevant for the system.

4.1 Shallow lexicon acquisition tools

Shallow lexicon acquisition environment consists of several programs, including *User Interface*, for different stages of lexicon acquisition. *Web Spider* creates lists of words from a particular domain web site (5 million word corpus of US patents, in our case) in text format. *Word Sorter* sorts input wordlists in alphabetical, reverse or frequency order. *Pre-POS-Tagger* creates "dirty" lists of parts-of-speech. The human further cleans these lists. *Word Format Converter* converts lists in .txt formats into a .wdl format, - a special format used by AutoPat programs. *Word List Creator* takes unsorted files, in .wdl format sorts them in any order, merges or subtracts lists of words. *Word List Editor* maintains tagged lists of words allowing for editing, adding, deletion and search of the words. *Tag Editor* edits number and content of tags assigning them to certain groups of lexemes in the final morphological lexicon.

User Interface is used for shallow lexicon acquisition in the course of automatic spell checking. A word typed in by the user is highlighted as misspelled in two cases: when it is really misspelled or when it is not found in the underlying lexicon. The main distinguishing feature of the AutoPat spell checker is that in addition to providing hints to correct a word, it also provides for a pop-up menu of features for a word (in case the user considers it is correct) to be put into the AutoPat shallow lexicon with proper description.

4.2 Predicate lexicon acquisition tools

The main tool for predicate acquisition is graphical Predicate Lexicon Interface. It is directly linked to the main application engine, which relies on linguistic knowledge contained in the lexicon. The interface allows for editing any of lexicon fields, search any word by its prefix or semantic class, propagate changes from one field to another. The interface program has a built-in morphological generator that automatically generates all the word forms of the predicate necessary for generation. The interface has a standing menu of semantic classes and case-roles to select from when acquiring a new word. The acquirer can

customize the menu of semantic classes. Most of the fields of a new predicate entry are automatically filled with default fillers after the semantic class is acquired. The interface is programmed so as to keep acquirer "on the right road" by means of different hints and waning messages. The user can also acquire a predicate through the *User Interface* by simply typing it in a pop-up box and selecting its semantic class in the menu. The grammar forms of a new word are automatically generated in a word box for the user to check and edit if necessary, other information is assigned to a new predicate automatically by default depending upon a semantic class selected by the user. Every new word thus introduced by the user is flagged so that later a linguist could check its entry through the predicate dictionary interface closed for the user.

4.3 Grammar acquisition tools

Grammar acquisition tools include 7 compilers. Compilers 1-4 belong to the AutoPat analyzer, while compilers 5-7 compile rules for the AutoPat generator. All compilers have front-end interfaces providing rule writing help. The formal language for writing rules is very simple and has an IF-THEN-ELSE-ENDIF structure (see Figure 3). Every compiler has another interface to test the rules. Compilers for the analysis rules allow downloading any text files, not necessarily the user's input into AutoPat. That means that these compilers can be used as stand alone programs. In fact the whole morphology analysis module can be used as off-the-shelf-tool separately from AutoPat. *Compiler-1* is used to create tag disambiguation rules, which are applied to the Tagger output. These rules only use context information, which might be a tag or a lexeme within a 5-word window with a tag in question in the middle. The output of this compiler together with the output from the *Tagger* is fed to *Disambiguator* and used for the first disambiguation pass. *Compiler-2* is used to create or edit the second set of tag disambiguation rules that use syntactic knowledge about the case-roles filled by the analyzed strings. *Disambiguator* uses the output of this compiler at the second pass. *Compiler-3* creates rules, which determine whether singular and plural forms of the nouns belong to the same lexeme and can be considered as coreference candidates. *Compiler-4* creates rules for determining agreement features between the predicate and its first case-role. *Compiler-5* is used to write rules for linerazation of the claim plan tree of predicate templates. They specify the order of the words in every predicate template and the location of the templates relative one another in the nascent claim. These rules are more often subject to changes than any other rules. They are fed into *Linearizer* that substitutes the tree of templates with a bracketed string of tags. *Compiler-6* is for writing cohesion rules that delete some of the tagged strings from *Linearizer* output, insert commas and assign morphological features to predicates. Condition part of the rules uses specific knowledge provided by the *Linearizer* and by the predicate lexicon. *Compiler-7* is used for writing rules for inserting determiners before noun phrases in the final claim text. These rules should recognize coreferential phrases, which may be parts of other phrases or worded differently.

In AutoPat three types of rules are not directly linked to any compiler for updating but are "welded in" the programs. They are tagging and semantico-syntactic rules in the analysis module, and text planning rules in the generation module. Tagging rules are very simple and only suggest look-ups in the morphological lexicon. These rules can indirectly be updated through editing tag sets and morphological lexicon. Results of this knowledge update can be displayed in a. special developers' interface. Syntactico-semantic rules rely on interactive knowledge elicitation procedure and consist in looking up a predicate (selected by the user at the knowledge elicitation stage) at the predicate lexicon, presenting the user with a selected predicate template, assigning a case-role status (place, manner, etc.) to every phrase put by the user into a corresponding slot of the template and registering the boundaries of these phrases. Though these types of rules are not editable, the output of syntactico-semantic analyzer can still be checked through the developers

interface built into the users' interface (see Figure 2) and its output can be edited indirectly by editing predicate lexicon. Text planning rules are very complex. They include algorithms of grouping and sorting conceptually close predicate templates into a forest of trees relying on semantic, legal, stylistic and rhetoric domain knowledge built into the system.

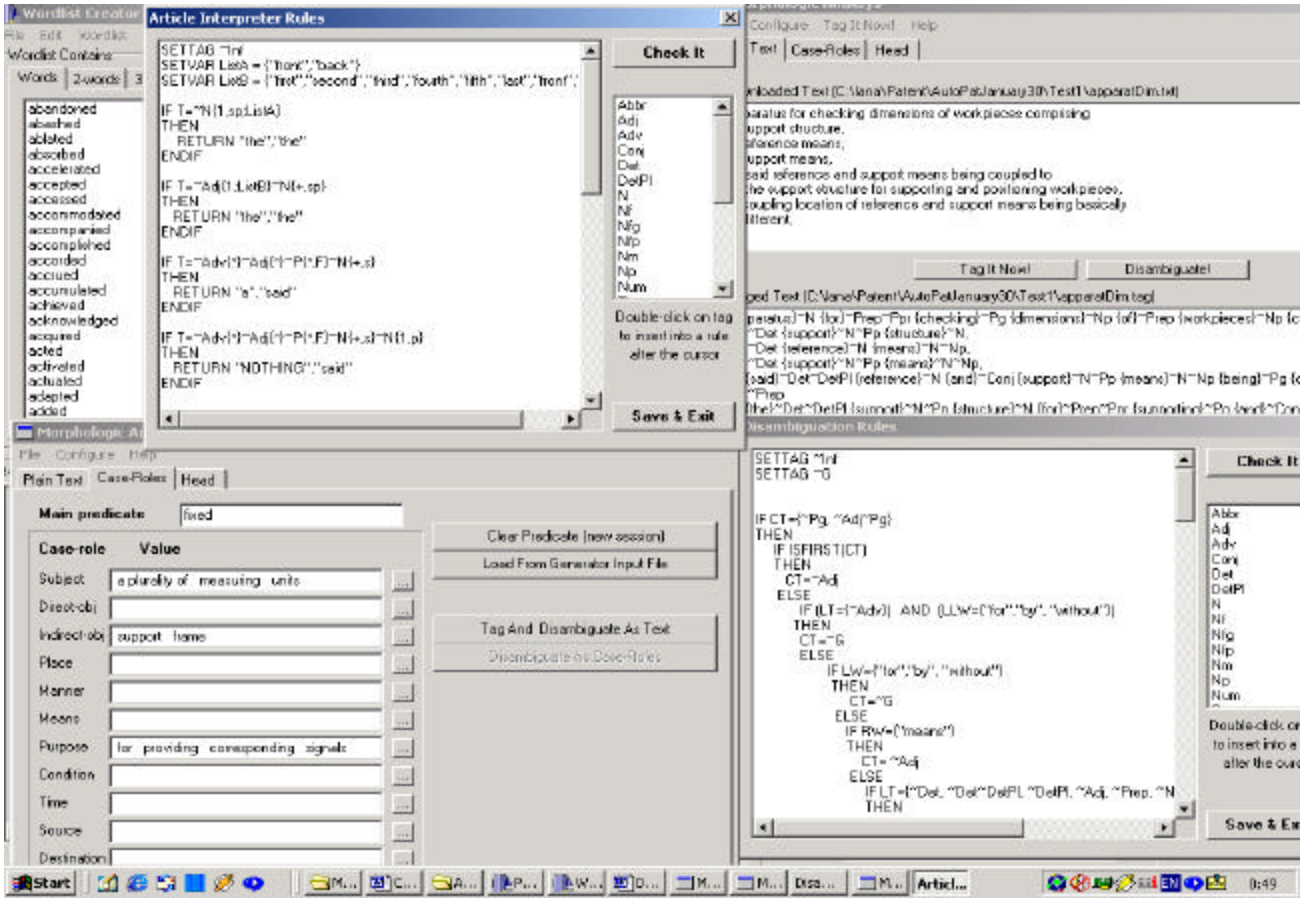


Figure 3. A screen shot of different compilers interfaces

These rules are not editable. But the developer can still update the structure of this tree by updating the predicate lexicon. A special interface was built for the developer to follow the stages of construction of the internal meaning representation and intermediate outputs of every generating procedure. In fact the rules for building a text plan is language independent, they depend only on semantic properties of predicates which could be treated as universal for different languages.

5. Discussion and conclusion

In this paper we addressed the problem of saving on software customization when developing a family of NLP applications sharing domain and task requirements or when updating applications once created. We illustrated the approach on the example of migrating from a prototype system for authoring patent claims to an AutoPat product. The migration was performed in two steps. The first step was the analysis of the aplication, the improvement of old components, such as generator, and the realization of new

components, such as morphological analyzer and a new user interface. The second step of migration which was described in this paper was to create developers tools for customization of application and integrate them into the system.

We were unable to compare the effectiveness of our development tools to other such tools due to their unavailability. Most of developer's tools are components of commercial products and are presented as black boxes, only used internally. This makes them unsuitable for research purposes (see, for example, a similar complain in (Lezius, 1998)).

The application development process described in the paper and targeted at saving on software customization emphasizes reuse and integration. At the level of each component, the AutoPat developer can access specific tools to perform reuse and customization. Integration is about the extent of compatibility of these tools and how seamlessly they can facilitate the development of applications. The development process of AutoPat-product validated the effectiveness of both the tools and their integration into the system. Programmers' work on AutoPat was finished long before the system could be considered a product. After manual acquisition of a training amount of knowledge for programming work the linguist completed the task of creating product-size and -quality knowledge without extra programming effort. We are planning to reuse the same tools for other applications of the same family (see Introduction) including syntax parsing, machine translation and automatic indexing. For example, we have already started the work on machine translation of patent claims where all the English lexicons and tools (e.g. interfaces) for their acquisition will be reused though augmented with new relevant for MT functionalities.

Acknowledgements

This research and development has been supported by Zacco A/S, - international Intellectual Property Rights consulting firm, Denmark, Sweden, and Norway.

References

Bredenkamp and Henzte, 1995. Some aspects of HPSG implementation in the ALEP formalism. Working Papers in Language Processing No 46.

Cunningham, H 1999. JAPE: a Java Annotation Pattern Engine. Research Memorandum CS-99-06, Department of Computer Science, University of Sheffield

Degoulet P, Jean FC, Engelmann U, Meinzer HP, Baud R, Sandblad B, Wigertz O, Le Meur R, Jargermann CA. 1994. The component-based architecture of the HELIOUS medical software engineering environment. *Comp. Methods and Programs Biomed. 45, Suppl.*

Fischer,D., W.Mohr, and L.Rostek, 1996. A Modular, Object-Oriented and Generic Approach for Building Terminology Maintenance Systems. In TKE'96: *Terminology and Knowledge Engineering.* Frankfurt.

Lezius W., Rapp R., and Wettler M. 1998. A freely Available Morphological Analyzer, Disambiguator and Context Sensitive Lemmatizer for German. *Proceedings of the COLING-ACL'98 conference.* Monreal, Canada, August

Prieto-Diaz R. 1993 Status report: software reusability. *IEEE Software 10(3).*

Reiter, E.B. 1994. Has consensus natural language generation architecture appeared and is it psycholinguistically plausible? In *Proceedings of the 7th International Workshop on Natural Language Generation*

Sheremetyeva, S and S. Nirenburg. 2000. Towards A Universal Tool For NLP Resource Acquisition. *Proceedings of LREC-2000 (Second International Conference on Language Resources and Evaluation)* Athens, Greece., June

Sheremetyeva S. and S. Nirenburg. 1996. Interactive Knowledge Elicitation in a Patent Expert's Workstation. *IEEE Computer. Vol.7.*

Thomas I., and Nejmeh B.1992. Definitions of tool integration for environments. IEEE Software. 9(2).

Resource integration and customization for automatic hypertext information retrieval in a corporate setting

Maria Nava

Université de Paris-Sorbonne
Institut des Sciences Humaines Appliquées
96 boulevard Raspail,
F-75006 Paris, France

Electricité de France R&D
Dept. SINETICS/TAIC
1, avenue du Général de Gaulle
F-92141 Clamart, France

maria.nava@edf.fr

Abstract

In this paper, we describe our experience in reusing and customizing existing tools to meet new information retrieval needs in a corporate setting.
The problem was to supply an authoring aid to handle customers enquiry letters by exploiting a textual case base.
We decided to integrate, and go as far as possible with, a terminology extractor and a context exploration platform. They were previously developed through an academic and industrial collaborative research.
We have found a method to generate an information retrieval hypertext structure on a large collection of homogeneous documents by creating links between noun phrases that are pertinent for navigation. Noun phrases are selected by automatic extraction and filtered on the basis of the linguistic context class where they appear, also determined automatically.
We have tried to point out the peculiar features that made possible the reuse and integration of existing resources, to produce a relatively new solution to a fairly constrained real-world problem.

1. Introduction

Our work is motivated by an novel information retrieval (IR) need formulated in a corporate setting, at *Electricité de France R&D* (EDF R&D, the research and development department of the French national electricity board). The general problem was to supply an authoring aid to help EDF employees handle customers enquiry letters.

Starting from a textual case base and software available, we were invited to study a flexible and cost effective solution that would respect the employees savoir-faire and experience, and add value to existing tools.

Our approach aims at identifying the context where interesting NPs occur in the enquiry letters, in order to enhance the selection of pertinent cross-document links. Context identification is based on spotting linguistic markers of the expression of enquiries and on the exploitation of a structured lexicon that we can extract automatically from the textual case base.

2. The starting point

The initial scenario presented a number of constraints to be respected, concerning both the nature of the IR solution and the technical implementation.

2.1. Two corporate memory corpora

Two corpora were used to carry out a linguistic analysis, train our system for marker identification and test processing performance.

2.1.1. A large corpus of stored letters

A corpus of about 2000 customer letters, in French, was first made available by *EDF R&D*. The collection contains inquiries, intervention requests and complaints. Even when a complaint is not formulated explicitly, generally the writer's intention is to point at some sort of problem that needs fixing.

The corpus is homogenous from the point of view of the general subject matter and purpose of the letters. On the other hand, the variety of speech acts performed by the writers lends a challenging heterogeneity to the texts, interesting but problematic for automatic processing.

The corpus can be introduced in the corporate memory as a case base, and connected to customer profile and commercial strategy databases for global IR about a single customer case.

Unfortunately, letters in this corpus were not associated to the answers they had actually received.

2.1.2. A smaller corpus of letters and related answers

A second corpus of about 200 question-answer pairs is used for testing and discussing evaluation issues. It is a collection of letters that were sent directly to EDF branch managers to solicit special treatment on peculiar issues.

This gives the letters a somewhat special status, which is reflected in their style, vocabulary and structure.

We have used this smaller, more personal collection to put our system to test, point out its limitations, and try to explain them.

2.2. A terminology extraction tool: Lexter

The acquisition and exploitation of a structured lexicon are carried out automatically by the Lexter[1] system (Bourigault *et al.*, 1996), developed at EDF R&D in the framework of a PhD research project. Lexter was designed to extract noun phrases (NPs) from a corpus of texts (in French). Extraction is based on the hypothesis that eligible NPs must exhibit the syntactic pattern of candidate terms, as established by terminology theory. For example : *definite article + noun + preposition + noun* is an observed candidate term pattern. NPs are not extracted by direct pattern matching, but they are isolated by spotting their syntactic boundaries, like, for instance, verbs. The "terminological hypothesis" is not without consequence for our work, as it will be pointed out in the conclusive section.

Extracted NPs are then automatically organized in a structured network of head-expansion relations.

Lexter accounts for morphological variants and head-modifier relations of nouns and NPs, that are grouped into families. It also supplies simple distributional figures, such as frequency of a candidate term in the corpus or candidate term head-modifier productivity within the structured network.

Lexter also stores the whole corpus divided into paragraphs, along with a pointer to the location of each candidate term in the text. This feature was initially designed to supply the terminologist with a linguistic context for validation.

Extraction and corpus-related information is stored in a relational database. We have taken advantage of all Lexter features and results for the generation of hypertext links, as described below.

2.3. A context exploration tool: ContextO

The identification of context classes where candidate terms appear is based on the contextual exploration method (Desclés *et al.*, 1997) implemented in the ContextO platform (Ben Hazez & Minel 2000). The system was designed and is still developed at the LaLICC laboratory (*Langage, Logique, Informatique, Cognition et Communication*) of the Sorbonne University in Paris .

The exploration engine deployed by ContextO is based on the identification of markers of a large number of linguistic functions, as observed in the general language. Markers are acquired through a "manual" linguistic analysis of a corpus of texts, to model the expressions of linguistic functions, depending on the application. They are subsequently organized in semantic classes with object-model relations. Markers are stored in a knowledge base (a relational database, the same as Lexter' s), whichsi accessed by the contextual exploration engine of ContextO, a Java application. Markers are exploited by specialized agents, performing specific tasks. A number of tasks were already available; for example, the identification of static relations (*is-a*, *has*, etc.), causal relations, thematic focus, citations, definitions, etc. For the time being, ContextO exploits markers from French and Spanish, but could easily be adapted to other languages.

The study of our own corpus of letters has helped us find a number of linguistic structures regularly associated to the expression of complaints, justifications or requests. Each letter contains linguistic markers indicating a focus on certain speech acts that help the writer organize argumentative discourse.

For the first tests, the database contained about 200 markers organized into 24 functional classes ("complaint", "demand", "justification", etc.).

3. HyTEC, a new tool born from customization

Hypertext generation based on automatically extracted key-words usually produces an overwhelming number of non-pertinent links. Any NP can actually constitute an anchor for too large a set of heterogeneous links, a serious limitation to the effectiveness of IR.

By exploiting the features of the existing tools, we have designed a system, HyTEC (*Hypertext from TErms in Context*), capable of generating a IR hypertext structure on a large collection of homogeneous documents by selecting only those NPs that are pertinent for navigation.

Our work can be placed in the domain of IR automatic hypertext (Agosti *et al.*, 1997; Allan, 1997), where paragraph (and document) linking is based on IR similarity measures, and is typed.

The specification of our IR hypertext system is based on a real-world application, that is, browsing a large textual case base made of customer enquiry letters, along with the associated reply letters. The aim of the navigation in the document base is to help finding consistent answers to any new incoming letter.

As our document base is liable to frequent updating, we found it interesting that the hypertext structure be generated at each IR session. Therefore, the document base is dynamically indexed by a short content-sample text at the beginning of the session.

A new browsing session is booted by the content of the incoming letter, which supplies content elements to compute a thematic similarity with enquiry letters stored in the corporate memory.

Navigation allows to gather information on similar cases that have already been solved and reuse written material to compose a response to handle the problem.

3.1. Identifying the context of lexical expressions

Textual similarity is computed from what we call the “pragmatic profile” of an input letter. We want to identify the discursive context of NPs in order to select only the most interesting ones and create links to similar NPs appearing in the case base, in comparable discursive contexts.

Our research is based on the articulation of two principles:

1. The exploitation of a lexicon structured by grammatical relations, extracted automatically from the whole text collection;

1 © EDF R&D.

2. The identification of linguistic markers indicating the expression of requests, complaints, justifications and other discourse acts that are relevant in our working context.

These two principles are implemented in the two different NLP systems, that offer complementary functions and results, that we have integrated.

3.2. Computing lexical links between texts

Our hypothesis is that the co-occurrence of a candidate term and a focusing structure selects a portion of text interesting for our similarity search in the case base.

The search for pertinent markers is a means to refine link generation on a number of texts already selected by their lexical components, extracted by Lexter.

In order to reduce the number and, at the same time, to keep only the most pertinent links, we have decided to maintain only the links between NPs. NPs represent a form of mutual contextualization of lexical elements and allow a more precise automatic indexing than simple nouns (Evans & Zhai, 1996). For example, instead of retaining the simple word *electricity*, we will first choose expressions like *electricity bill* or *electricity meter* (as translated from French) as content carriers, because we feel they are thematically more precise.

We have then integrated this domain-specific lexical information, extracted automatically by Lexter, and semantic and pragmatic context information supplied by markers of the general language, identified by ContextO. The lexical information triggers context analysis to create a "signature", a context-tag / NP relation, that is used for indexing and filtering.

Even if the actual language we use is French, the same principle may as well be illustrated with an example in English, like

> Due to *temporary money problems*, I'd be happy if I could *pay the bill* by installments.

Context analysis is triggered by the phrases in italics (*pay the bill* would be a nominal form in French). As markers like *I'd be happy if* (demand) or *Due to* (justification) would be found in a particular context (by context exploration rules), the sentences would be tagged as belonging to a pertinent context class.

Links between portions of texts are computed by matching signatures formed by NPs that are flagged with a semantic tag indicating a context class.

4. Similarity search results

The results obtained by testing the system on three sample entry letter are summarized in Table 1.

The performance of our system on sample texts shows that the simple association of NPs and their conditions of use can effectively improve retrieval precision, when compared to results obtained by generating links between NPs alone.

	Before context analysis		**After context analysis**	
Samples	Initial number of links	Non pertinent	Non pertinent links eliminated	Pertinent links eliminated
1	158	81	71	8
2	93	23	16	11
3	78	32	24	5

Table 1: Results for three sample input letters on the main corpus

For instance, consider the following input text (as translated from French), where extracted NPs are in italics and context markers are in bold:

> Dear Sirs,
>
> Earlier this month, I have received an invoice from you, concerning the **use of gas and electricity**, whose amount *I do not agree* with. As the **big amount** you are asking for apparently concerns only a 2-month period, I have taken down the numbers shown on my **gas meter**. The meter indicated 00613, but your invoice reports 00878. I know this number represents an estimation.
>
> On the other hand, if we consider the **huge difference** between **your estimation** and my **actual gas consumption**, *I refuse* to **pay the amount** you are asking for and *invite you* to send me a **new invoice**, reporting figures closer to reality.

Context identification allowed to retain a target text like :

> Dear Sir,
>
> I am the tenant of the apartment located X Street in TheTown, belonging to Mr and Ms Y. Since I have taken up the place in 1996, I have only received invoices reporting estimations of my electricity consumption.
>
> Before I was here, the place was unoccupied. I'll take the liberty to tell you that at present, the electricity meter indicates 36,637.
>
> ***I'd be grateful if*** you could send me an invoice corresponding to **the actual consumption**.

Notice that the NP *real … consumption* was also included in a focused sentence in the input letter (*On the other hand …, I refuse ...*). Notice also that the target text focuses on the NP *electricity meter* (*I'll take the liberty*

to…), that could also be found in the input letter under the form *gas meter*.

We have found that it is not necessary to carry out context indexing for the input letter to improve precision: it is enough to search the context of NPs in the candidate target texts. On the other hand, to execute a relational indexing (NP + context tag) both for the input and the target texts allows precise link typing, which makes navigation easier.

The same search session as above allowed to eliminate a number of target texts, that had been retrieved be found on the basis of simple NP matching, but were not pertinent, like:

> Sirs ;
>
> I take the liberty to draw your attention to the dangerous situation menacing all the families living in our building.
>
> We experienced important damages due to water overflow last summer. To day, the leakage, which has not been stopped yet, affects the wall bearing our **electricity meters** and wiring.
>
> [*long descriptive text snipped*]
>
> If you consider that there actually is no danger, *I'd be grateful* if you could send us an official written declaration about this, *etc*.

In this target text, there is no co-occurrence of context markers and extracted NPs, as compared to the input letter. Therefore it was not retained by HyTEC, which improves retrieval precision.

Eventually, we are left with 1) non-pertinent targets that have been retained, but also 2) pertinent candidates that have not been retrieved.

In the first case, co-occurrence of markers and NP has been identified in a sentence or paragraph, yet the rest of the letter relates events or circumstances that are different from those found in the input letter. However, the noise caused by uninteresting letters is very low, considering the number of searched texts. In this case, retrieval precision would probably improve if we could rely on a global text model, accounting for lexical and discursive chaining.

In the second case, in spite of global similarity between the input letter and a possible candidate target, content proximity has not been identified. The most frequent cause of this kind of failure is that pertinent markers focus on *synonyms* of extracted NPs. Improved recall rates should be attained cost-effectively by adding a relatively small number (the corpus is homogeneous) of synonymic relations to the NP network. We are planning to test the integration of a tool (SynoTerm) that automatically supplies Lexter with candidate synonyms from general language resources (digital dictionaries) (Hamon, 2000).

5. Generation of a hypertext structure

The results of link computation are presented in the form of a hypertext structure generated on-the-fly, directly exploiting the data structure in Lexter's relational database.

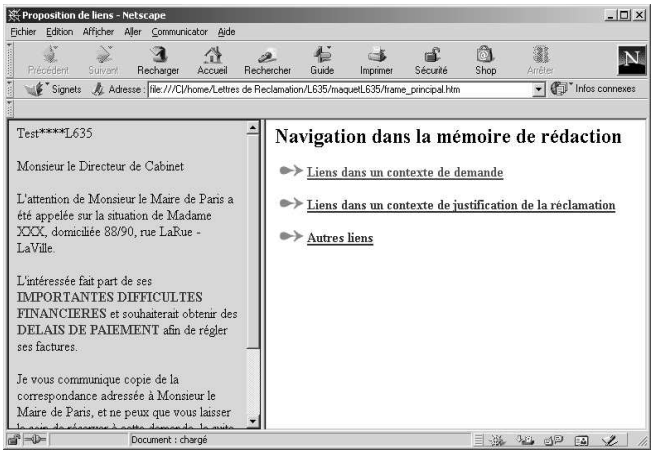


Figure 1: Navigating in context classes

The demonstration window (Figure 1) shows the text of the input letter (left) with salient NPs highlighted and (right) a choice of links to pertinent context classes (complaint formulation, enquiry, justification, etc.).

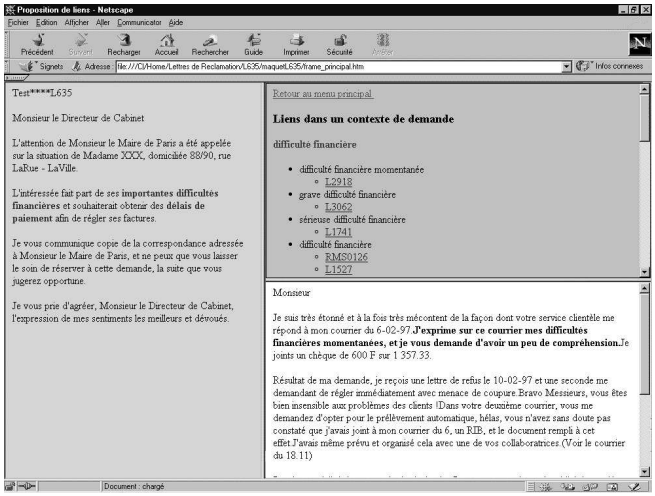


Figure 2: From typed links to target paragraphs

Once a context class has been selected, the links to target texts appear (Figure 2).

6. Evaluating task performance

The results of the first experiments are encouraging in terms of precision/recall ratio (Nava & Garcia, 2001). However, we feel that traditional evaluation measures are not completely adapted to the task, as it is often a delicate matter to decide whether two letters are even loosely connected.

As we are currently testing the system on a more extensive input letter set, a more flexible evaluation protocol is under study. It will possibly include an improved link type taxonomy and link weighing.

7. Methodological issues about customization

In this paper, we have shown how we have reused, customized and integrated two different NLP tools.

Lexter and ContextO belong to two different paradigms, which are, we believe, complementary.

Lexter and ContextO were not designed to be integrated. Lexter is a corpus-based extraction tool, ContextO is a knowledge-rich filtering system. However, we found that their results are complementary, and their coupling has provided benefits that reach beyond the simple application of cascading NLP processes.

In our case, we have observed that facility of integration and customization are related to a number of features, ranging from modularity and separation of linguistic resources and procedures, to implementation by off-the-shelf technology.

7.1. Lexter

We have taken advantage of the following:

1. Corpus-based extraction without domain-specific resources (dictionary, thesaurus, etc.);
2. Shallow morpho-syntactic structuring;
3. Access to the full-text source;
4. Simple distributional data (frequency, head-modifier productivity in the morpho-syntactic network);
5. Extraction results stored and organized in an off-the-shelf relational database (Microsoft Access).

On the other hand, considering our particular application, we have experienced an important limitation due to the fact that Lexter is basically an extractor of candidate terms. This is certainly well suited for technical, domain-specific text processing; but for our collection (customers letters), this constraint is rather restrictive. Given the source, purpose and style of the texts, we would have been happier with additional lexical information, like, for example, verbal phrases (which are generally ignored by the classical terminology theory and applications). In informal writing, an expression like *pay the bill* is often preferred to *bill payment*.

7.2. ContextO

ContextO was *designed* to facilitate the acquisition and reuse of linguistic knowledge, based on the following:

1. Separation of linguistic knowledge and search engine;
2. State-of-the-art object model of text, linguistic data and tasks;
3. Independent specialized agents exploiting the knowledge base;
4. Portable Java engine implementation accessing an off-the-shelf relational database (Microsoft Access).
5. Exploitation of markers related to structures of the general language.

It must be noted, however, that if the marker collection is largely domain-independent, it is sensitive to style and textual genre. Moreover, certain semantic classes (for example, thematic markers) are more generally reusable than others (for example, static relation markers).

Prospective work includes the adaptation of our approach to automatic, corpus-based terminology structuring.

8. Acknowledgements

Our PhD research is financed by *EDF R&D*.

9. References

Agosti, M., Crestani, F. and Melucci; M., 1997. On the use of information retrieval techniques for the automatic construction of hypertext. *Information Processing and Management*, 33(2):133-144.

Allan, J., 1997. Building hypertext using information retrieval. *Information Processing and Management*, 33(2):145-159.

Ben-Hazez, S. and Minel, J.L., 2000. Designing tasks of identification of complex linguistic patterns used for semantic text filtering. *Proceedings of RIAO 2000*, Paris, France, 1560-1568.

Bourigault, D., Gonzalez-Mullierand, I. and. Gros C., 1996. Lexter, a Natural Language Tool for Terminology Extraction. *Proceedings of the 7th EURALEX International Congress*, Göteborg, Sweden, 771-779.

Desclés, J.P., Cartier, E., Jackiewicz; A. and Minel, J.L., 1997. Textual processing and contextual exploration method. Proceedings of *CONTEXT '97*, Rio de Janeiro, Brazil, 189-197.

Evans, D.A. and Zhai C., 1996. Noun-phrase analysis in unrestricted text for information retrieval. Proceedings of the *34th Annual Meeting of the Association for Computational Linguistics*, Santa Cruz, California.

Hamon, T., 2000. *Variation sémantique en corpus spécialisé : Acquisition de relations de synonymie à partir de ressources lexicale*s. PhD Thesis, Université de Paris Sorbonne.

Nava, M. and Garcia, D., 2001. Automatic hypertext information retrieval in a corporate memory using noun phrases in context. In R. Mitkov (ed.), *Proceedings of Recent Advances in Natural Language Processing 2001 (RANLP '01)*, Tzigov Chark, Bulgaria.

Lexicalized Grammar Specialization for Restricted Applicative Languages

Patrice Lopez, Christine Fay-Varnier, Azim Roussanaly

LORIA, INRIA Lorraine and Universities of Nancy
BP 239, 54506 Vandœuvre-lès-Nancy, France {lopez, fay, azim}@loria.fr

Abstract

In the context of spoken interfaces, we present a practical methodology and an implemented workbench called EGAL (Lexicalized Tree Grammar Extraction) dedicated to design and test restricted languages used in specific task-oriented applications. A complementary methodology is proposed to process the extraction of these applicative languages from a general LTAG grammar and a training corpus. Additional results allow us to estimate the representativeness of the training corpus. An application of the system is presented for the tuning of a LTAG grammar dedicated to a spoken interface on the basis of a Wizard of Oz corpus.

1. Introduction

1.1. Motivations

In the case of a spoken dialogue system, the quality of the human computer interaction largely depends on the ability of the computer to understand spontaneous utterances normaly used by humans. The practical development of a spoken interface for a restricted domains implies that we perform the tunning of existing lexicon and grammar to a particular application. This paper proposes a methodology and an implemented workbench called EGAL (Lexicalized Tree Grammar Extraction) dedicated to design and test restricted natural languages used in specific task-oriented applications. This workbench is a sub-component of a general platform for designing spoken language systems and addresses software designers who are non-experts in natural language processing.

Specializing a grammar for restricted domains supposes at least the two following tasks:

- Cutting down the existing lexicon and grammar.
- Adding new words and new syntactic constructions.

In recent years, the development of large covering lexicalized grammars could be observed. Complementary, studies about the use of this kind of formalism for parsing spoken language have been performed. To address spoken disfluencies and robustness constraints in the context of human computer interaction, additional mechanisms have been proposed which often depend on the application domain. At the lexical and syntactic level, the following adaptations are required:

- Model spoken phenomena that could be considered agrammatical or rare in written language but frequent in spontaneous speech such as ellipsis or interpolated clauses (Price et al., 1989).
- Use robust parsing techniques to take into account the variability of the input.
- Specialize a lexicon and a grammar dedicated to text to a specific kind of dialogue and a specialized domain.

This paper addresses the last point. The specialization of a general hand written grammar to a specific domain is not a trivial task. Probabilistic methods and grammar inference as (Bod, 1995) can be seen as an alternative to this problem. Still a linguistically motivated hand written grammar provides a precise understanding of the occuring phenomena and reusability. In particular, this kind of grammar allows us to take into account the important ambiguity of the syntactic level. This ambiguity is one of the main differences between natural language that we want to process and regular languages which are just an approximation of natural language. Moreover, probabilist methods need very large annotated training corpora. Their development can require the same amount of effort as the writing of a wide-covering grammar.

We present in this paper a methodology and an implemented system called EGAL (Lexicalized Tree Grammar Extraction), able to perform an assisted specialization of a general grammar in order to obtain an applicative sublanguage from a corpus. When the specialized grammar has been obtained, a parsing module allows the evaluation of the grammar on a test corpus and the choice between various parsing algorithms and strategies. The partial and complete derivations can be visualized and compared following different criteria. The methodology also allows us to obtain information about the representativeness of the initial training corpus. Finally, the lexicalized grammar and the parser can be integrated in concrete HCI systems.

The proposed workbench can be applied to various domains. Our main goal is to design generic and portable spoken systems that can process spontaneous language. To illustrate our methodology and system, we have chosen a target application and collected an experimental Wizard of Oz corpus from which we have extracted a lexicon and a specialized grammar. We have finally evaluated the representativeness of the resulting grammar.

1.2. Lexicalized Tree Adjoining Grammars

The lexicalization of a syntactic formalism consists of the association of a set of appropriate syntactic contexts to each entry of the lexicon. Lexicon and grammar are merged in a single entity called *syntactic lexicon*. Lexicalization provides at least two main advantages: First the ability to describe syntactically each specific lexical entry allows us to choose the required complexity of the syntactic structures with flexibility. Even for restricted domains, too much generalization in syntactic descriptions generally results in

unexpected border effects. Secondly the lexicalization allows parsing heuristics since a lot of syntactic ambiguity problems become lexical ambiguities which are easier to process (Abeillé, 1991).

The choice of the formalism is essential for the representation and the understanding of linguistic phenomena. It is also important to consider its applicability for NLP applications. The Lexicalized Tree Adjoining Grammars (LTAG) (Joshi and Schabes, 1992) is interesting for parsing and generation thanks to the lexicalization property and extended domain of locality. Linguistic studies and large-covering grammar developments for example in English and French have shown the practical interest of these properties. Moreover probabilistic models based on LTAG as stochastic TAG (Srinivas, 1997) or supertagging (Srinivas, 1997), allow optimizations for the processing of lexical and syntactic ambiguities on the basis of preferential choices. These properties make the LTAG formalism interesting for spoken utterances understanding (Halber, 1998) and generation in spoken systems (Becker et al., 2000).

Still the lexicalization has some drawbacks, in particular the task of designing of the grammar. Still work in progress, the English grammar of the XTAG system (Doran et al., 1994) already took ten years of development, the French grammar (Abeillé et al., 1994) more than seven years. A large covering grammar can include several thousand of elementary tree patterns called *schemata* (Candito, 1999) and a syntactic database that gives for each lemma the set of corresponding trees or tree families. Considering a given application, the use of the whole general grammar would lead to a prohibitive number of hypotheses. Moreover our goal is to avoid the development of a new grammar for each new application.

Work on the use of LTAG for dialogue systems for both parsing and generation of a sublanguage has been done recently, but the tuning of a general grammar to a specific application and domain remains a problem for the practical application of such a lexicalized formalism. The extraction of sublanguage grammars for LTAG has been discussed in (Doran et al., July 1997). But the proposed solution was based on successive manual approximations by experts. No practical methodology was proposed. No significant features have been identified that could help to perform more efficiently this task or that could lead to a software engineering solution.

2. Collection methodology

2.1. Restricted language

A restricted language can be defined as a set of utterances linked by a restricted domain, used for a particular function and generated by a specific grammar and vocabulary (Deville, 1989). Two factors limit the general language: The kind of discourse or dialogue which is realized and the application domain of the system. A restricted language is not only a subset of the whole language since an application can use technical terms which are only relevant for the domain. Moreover even in limited domains, the size of the vocabulary and the syntactic constructions change as the application evolves. Consequently a system has to propose a methodology to add new words and new syntactic contexts for the structures that would not be covered by a general grammar.

The practical advantages of the restricted language definition are a reduction of the combinatoric complexity of the processing and the ability to use a hand-written grammar (which is for example not realistic for dictation systems).

In the case of spoken dialogue systems, we claim that the systems should not understand words out of the corresponding restricted language because such words do not belong to the competence of the system. The lexicalized grammar defines here the norm of the applicative language, i.e. what is acceptable or not. Since domain restricted applications should not understand every user's request, they eventually have to lead additional dialogues with the user in the case of out of domain words.

2.2. Wizard of Oz experiments

The Wizard of Oz experiments are now widely used as a first step of the design of a spoken dialogue system. This experiment consists in the simulation of a spoken dialogue system in order to get a set of possible user interactions for a given application. The resulting corpus (which has a subjective representativeness) becomes a reference for the linguistic modeling. In other restricted domain applications, such as automatic thematic classification of e-mail in e-commerce, a similar step is necessary.

One of the main problems related to this kind of corpus is its representiveness for the application sublanguage we want to model. If the principle of restricted language is relevant, we can expect that by increasing the size of the training corpus, we will reach a size such that any addition will not result in a significant increase in the vocabulary or the size of the grammar.

Our approach consists first of obtaining a corpus which is classically divided in two parts. The first part is used to design the grammar of the restricted language (*training corpus*). The second one is dedicated to test (*test corpus*).

We have presented the different aspects which are essential for the kind of system we want to build: WoZ Experimental approach in order to obtain a corpus, specialization/designing of a lexicalized grammar dedicated to spoken language understanding, test of the resulting grammar and representativeness evaluation of the training corpus.

We have not found any existing workbench for lexicalized grammar which would combine all these aspects.

3. Presentation of the workbench

The general organization of a lexicalized tree grammar dedicated to parsing relies on three main knowledge sources:

- A morpho-syntactic database which associates an inflected form, a syntactic category and a set of morphological features.
- A syntactic database which associates a given lemma to a set of elementary trees representing the valid syntactic context for this lemma.
- A set of schemata (Candito, 1999).

The grammar designing/tuning module of the system is based on these three kinds of databases (see figure 1).

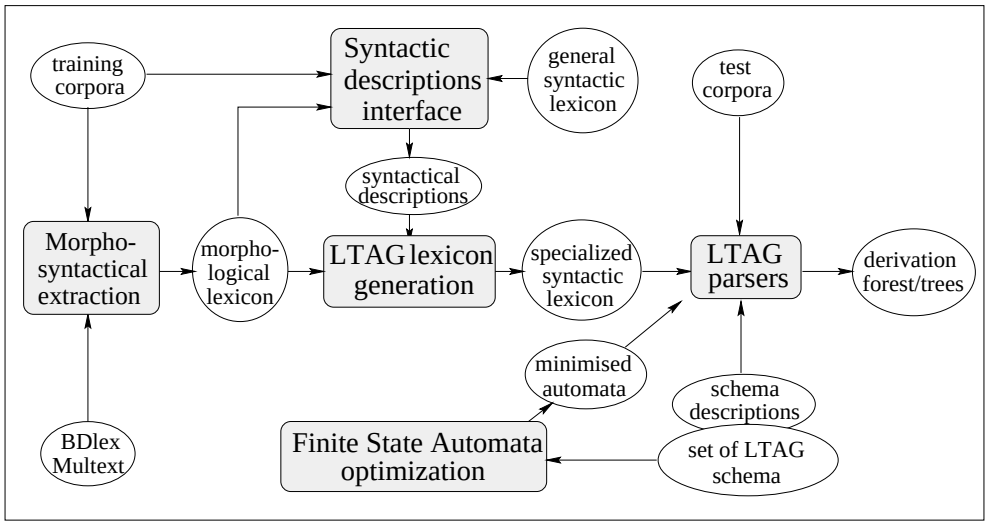


Figure 1: Overall presentation of the EGAL workbench.

3.1. Assisted generation of the lexicons

Morpho-syntactic extraction Given a training corpus, this step just corresponds to the exploitation of existing morpho-syntactic databases, Multext and BDlex (Ide and Véronis, 1994), by extracting the required information for all the words used in the corpus. This process has been implemented with an automaton-based compilation of the morpho-syntactic databases.

Set of schemata We assume that we already have a set of schemata (non-lexicalized elementary trees). For instance this schema can come from an existing hand-written grammar or from an automatic tree generation system as proposed by (Candito, 1999). A graphical editor allows the design of new schemata or the modification of existing ones.

Syntactic descriptions The goal of this module is to identify the syntactic properties associated with a lemma in order to select its correct syntactic structures. This identification is not an automatic process since resources able to enumerate all the possible predicative structures for a given lemma are not available. This result is obtained on the basis of a graphical interface dedicated to non-grammarian users.

The main idea is to associate a term of syntactic features to characterize (i) the various possible syntactic contexts covered by the general grammar (i.e. the various LTAG schemata), (ii) each lemma of a given corpus on the basis of a linguistic test suite illustrated by examples. The unification of these two structures characterizes then the precise subset of the acceptable syntactic constructions for each lemma.

The definition of our syntactic feature set is based on linguistic studies of French (mainly (Abeillé, 1991)). The current system uses nineteen syntactic features for the characterization of a verbal context (for example arity, passive, subject-verb inversion, support verb, equi-verb, reflexive, auxiliary,...) and a frame of possible prepositions. An alternative would be to use the syntactic features corresponding to the metagrammar described in (Candito, 1999) and the corresponding grammar generation system: In this case the description term corresponding to the schema that would be obtained automatically with the generation of the schemata.

For each syntactic feature we create a linguistic test composed by a question labeling the set of possible values and a set of examples. The tests are stored in a declarative way in a XML document. This XML document is then used by a generic test interface that allows a user to fill the frame for each lemma in a friendly way. The result of these tests consists of a feature term which is the syntactic description of the lemma.

For example the two following questions begin the French linguistic tests for verbs:

- Which auxiliary is used with the verb? (one between *être* and *avoir*)
- Can the verb be used in an intransitive/transitive/ditransitive context?

The tests continue until the complete frame of syntactic features and the preposition frame are specified.

The unification of the terms associated to the different schemata and the term obtained for a given lemma gives the correspondence between an entry of the lexicon and the subset of schemata that can be anchored by this entry. For instance on figure 2, the tree schema can be used with the lemma *enlever* since the two syntactic descriptions can be unified. This lexicalization process is uniform with the lexicalization performed on the basis of morphological features (for instance infinite verbs only lexicalize infinitive contexts).

This module can be used in two different ways:

- Completion of the whole list of linguistic tests in order to characterize completely a lemma for all its possible uses.
- Characterization of the syntactic contexts observed in the training corpus.

For the proposed methodology, the second possibility must be chosen. The list of utterances (in the training corpus)

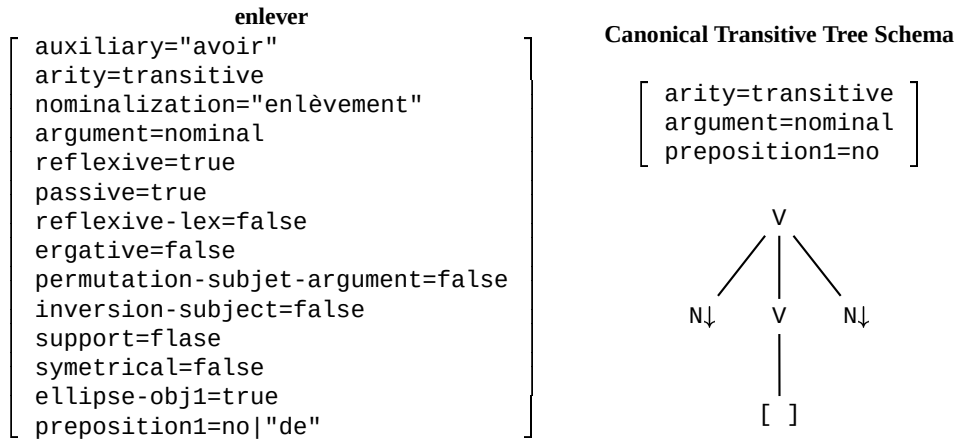


Figure 2: Two examples of syntactic descriptions: one for the French lemma *enlever*, one for a transitive tree schema.

which contain the lemma and the linguistic tests are proposed simultaneously to the user by the graphical description tool. In our methodology, contrary to the classical approach for cutting down the grammar, we specify each entry of the lexicon in terms of its category and also in terms of its correct syntactic contexts. The resulting grammar is really a *lexicalized* subgrammar.

We do not use the principle of tree family used by the XTAG system because of the small size of the lexicon and for reasons of computational efficiency. With tree families, the final selection of trees associated to an entry of the lexicon is obtained dynamically by unification at the time of instantiation. Here the correct trees are already predefined and listed in the syntactic lexicon.

A complementary tool for linguists allows the design of linguistic tests. We note that:

- The descriptions obtained by filling the features frame are independent from the lexicalized formalism. For instance, one could use HPSG lexical types.
- This module allows us to integrate easily new words to a system by characterizing the inflected forms which are not recognized during the morphological extraction. Moreover a very important point is that adding new words with this tool can be done by a non-linguist user if the linguistic tests are correctly written.

Automatic generation of the specialized LTAG syntactic lexicon This step produces the syntactic lexicon by exploiting information from the three databases described before. We add to each entry of the morphological lexicon the list of LTAG schemata which can be lexicalized. This list is obtain by

- The unification of the morphological features of the flexed form with the morphological features of the node to be anchored.
- The unification of the syntactic feature term that describes the corresponding lemma with all the syntactic feature terms of the schemata.

The links to schema are simply noted with external references using the XML links mechanisms. The final anchoring is classically done as a pre-parsing process.

3.2. Parsing test workbench

After the generation of a grammar for an applicative sublanguage given a training corpus, this module aims to test the results on a second test corpus. It allows us:

- To visualize the parsing results (both partial and complete ones).
- To check the generated grammar and possibly change manually some data in the syntactic lexicon or the set of schemata.
- To test and to compare various parsing heuristics and strategies.
- To study out of grammar phenomena.

This workbench implements two chart parsing algorithms and several parsing heuristics:

- A bottom-up connection driven algorithm that delivers extended partial results (Lopez, 23 25 February 2000).
- An implementation of the top-down Earley-like algorithm of (Schabes, 1994).

The bottom-up parser gives complete and partial parses with or without unification of features structures used in Feature Based LTAG. These different kinds of results aim to test the grammar by identifying the step involved in the failure of the parsing.

3.3. Technical choices

The implementation have been made in Java for portability reasons. All the involved data are encoded in the highly portable formalism XML. A specific application of XML dedicated to resources used with LTAG has been developed called TagML (Tree adjoining grammar Markup Language) (Lopez and Roussel, 2000). TagML allows an efficient representation of these data in term of redundancy. For instance it is possible to encode only one time substructures that are redundant in several schemata. Similarly it is also possible to share feature equations occuring in several schemata. All these redundancies imply redundant

computation that could be avoided. This standard representation allows easy resource exchanges with our research partners and allows the sharing and the comparison of tools. The DTD allows us to check the consistency of the whole grammar. Every parser that respects this encoding norm can be integrated to the parsing workbench very easily.

The Java sources, classes and documentation of the parsing test workbench, including editors, are freely available on request. The other modules should also be packaged and available at the time of the conference.

4. Grammar of the GOCAD corpus

4.1. A target application: GOCAD

The GOCAD application aims to model geological surfaces. The protocol and the Wizard of Oz experiment used with this application are presented in (Chapelier et al., 1995). This experiment allowed us to obtain a corpus which has been encoded following the TEI specifications[1]. This corpus of transcribed French spoken utterances is presented in Table 1.

4.2. LTAG for the applicative restricted language

The corpus has been divided in a training corpus (80% of the utterances) and a test corpus (20%). The size of the LTAG grammar obtained with the EGAL system is presented Table 2. The total number of links to schema is a good metric for the whole size of the syntactic lexicon.

Given this specialized lexicalized grammar, the average time for parsing is 167 ms per utterance with an average lenth of utterances of 6.42 words per utterance on Sun Ultra 1. It is difficult to compare with results obtained with the complete French LTAG grammar because first the covering of this complete grammar is really limited for this corpus (124 unknown words). Moreover, for technical reasons, this grammar has been designed for the XTAG system which is very difficult to install (SunOS 4 only for instance) and use. For indication, the parsing of sentences of 10 to 15 words can take more than ten minutes.

4.3. Representativeness of the training corpus

The morphological extraction phase and the generation of the syntactic lexicon for GOCAD are fast (less than one second for the first one, less than ten seconds for the second on an average workstation). Consequently it is possible to realize systematic tests to study the evolution of the generated data. The method consists of first randomly selecting utterances from the whole corpus and then generating the corresponding LTAG grammar. This allows us to study the evolution of the size of the grammar given the number of links to a schema in function of the number of utterances taken into account. A decrease of the slope of the curve indicates an improvement of the coverage. A horizontal asymptote would mean that the coverage of the grammar is perfect for the target sublanguage. The Figure 3 gives the evolution observed for the GOCAD corpus: The number of new structures obtained by considering the last two hundred utterances is very low and we can conclude that the final generated grammar is a good approximation of the GOCAD sublanguage.

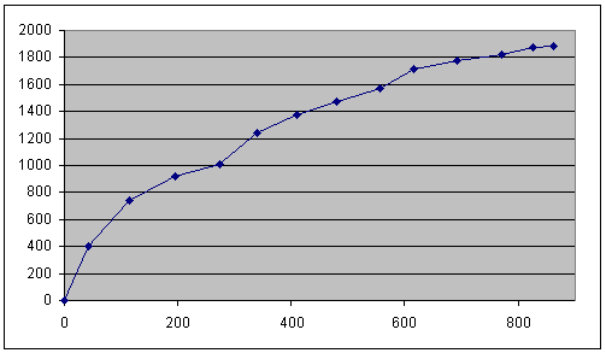


Figure 3: Evolution of the size of the generated LTAG grammar (number of links to schema) as a function of the size of the training corpus (number of utterances)

Such a result can be very useful to estimate the size of the corpus needed to reach a satisfactory covering rate. Covering 100% of the utterances is not our objective since in our approach only utterances corresponding to the competence of the spoken system need to be understood.

5. Future direction

We plan to see how the workbench scales up to other corpora and applications different than spoken interfaces. Our second goal is to extend the specialization workbench to cover multilinguality. One difficulty that arises is that the syntactic features used for the description of tree schemata and lemmas can be different from one language to another. It would mean that only a subset of these features has a real multilingual validity and could be used for parallel specialization of multilingual syntactic ressources. Syntactic features depending on the language might be limited if we only restrict them to pairs of languages, i.e. not considering all the languages at the same time.

6. References

Anne Abeillé, Béatrice Daille, and A. Husson. 1994. FTAG : An implemented Tree Adjoining grammar for parsing French sentences. In *TAG+3*, Paris.

Anne Abeillé. 1991. *Une grammaire lexicalisée d'arbres adjoints pour le français*. Ph.D. thesis, Université Paris 7.

Tilman Becker, Anne Kilger, Patrice Lopez, and Peter Poller. 2000. Multilingual generation for translation in speech-to-speech dialogues and its realization in verbmobil. In *ECAI'2000, Berlin, Germany*.

Rens Bod. 1995. *Enriching Linguistics with Statistics : Performance Models of Natural Language*. Ph.D. thesis, University of Amsterdam.

Marie-Hélène Candito. 1999. *Structuration d'une grammaire LTAG : application au français et à l'italien*. Ph.D. thesis, University of Paris 7.

Laurent Chapelier, Christine Fay-Varnier, and Azim Roussanaly. 1995. Modelling an Intelligent Help System from

[1]This corpus is available on the Silfide server (`http://www.loria.fr/projets/Silfide/`)

Number of user utterances	number of words	average number of words/utterance.
862	5535	6,42

Table 1: GOCAD corpus

number of inflected forms	number of schemata	number of links to schema
526	71	1776

Table 2: Size of the LTAG grammar corresponding to the training GOCAD corpus.

a Wizard of Oz Experiment. In *ESCA Workshop on Spoken Dialogue Systems*, Vigso, Danemark.

Guy Deville. 1989. *Modelization of task-Oriented Utterances in a Man-Machine Dialogue System*. Ph.D. thesis, University of Antwerpen, Belgique.

Christy Doran, Dania Egedi, Beth Ann Hockey, B. Srinivas, and Martin Zaidel. 1994. XTAG System - A Wide Coverage Grammar for English. In *COLING*, Kyoto, Japan.

C. Doran, B. Hockey, P. Hopely, J. Rosenzweig, A. Sarkar, F. Xia, A. Nasr, O. Rambow, and B. Srinivas. July 1997. Maintaining the forest and burning out the underbrush in XTAG. In *Workshop on Computational Environments for Practical Grammar Development (ENVGRAM '97)*, Madrid.

Ariane Halber. 1998. Grammatical factor and spoken sentence recognition. In *Workshop on Text, Speech and Dialog, Brno*.

Nancy Ide and Jean Véronis. 1994. Multext (multilingual tools and corpora). In *14th Conference on Computational Linguistics (COLING'94), Kyoto, Japan*.

Aravind K. Joshi and Yves Schabes. 1992. Tree Adjoining Grammars and lexicalized grammars. In Maurice Nivat and Andreas Podelski, editors, *Tree automata and languages*. Elsevier Science.

Patrice Lopez and David Roussel. 2000. Predicative LTAG grammars for Term Analysis. In *TAG+5*, Paris, France.

Patrice Lopez. 23-25 February, 2000. Extended Partial Parsing for Lexicalized Tree Grammars. In *International Workshop on Parsing Technology, IWPT 2000*, Trento, Italy.

Patti Price, Robert Moore, Hy Murveit, Fernando Pereira, Jared Bernstein, and Mary Dalrymple. 1989. The integration of speech and natural language in interactive spoken language systems. In *Proceeding of Eurospeech*, Paris, France.

Yves Schabes. 1994. Left to Right Parsing of Lexicalized Tree Adjoining Grammars. *Computational Intelligence*, 10:506–524.

Bangalore Srinivas. 1997. *Complexity of lexical descriptions and its relevance to partial parsing*. Ph.D. thesis, University of Pennsylvania, Philadelphia.

A Versatile Knowledge Management Package

Hurskainen Arvi
University of Helsinki, Box 59
FIN-00014 University of Helsinki, Finland
Arvi.Hurskainen@helsinki.fi

Abstract

It is suggested that for the knowledge management system to be accurate, flexible and have wide coverage, the customer should have access to the whole chain of analysis modules, whether general or language-specific. The customer should not be made dependent on pre-coded texts. The analysis package should provide possibilities to perform various levels of encoding, disambiguation, etc. A knowledge management environment, tested with text of 8 million words, is briefly described.

1. Introduction

Sophisticated and powerful systems for knowledge management are not necessarily easy to use, while user-friendly systems are found among such applications that perform fairly simple tasks. Can these expectations be met within one single system? And if not, which of these two should be given preference? There are big differences among end users in the preparedness to use time for learning the system.

I personally, coming from the research community and serving primarily specialised communities outside the academia, am interested in maximizing the usability of a knowledge management package, which consists of language resources as well as of various language-specific and general-purpose applications. I shall discuss a system that analyses the language cumulatively, starting from morphology (Koskenniemi 1983; Hurskainen 1992) and extending to disambiguation (morphological and semantic) and syntactic analysis (Karlsson 1990, 1995a, b; Karlsson et al.; Tapanainen 1996, 1999; Tapanainen and Järvinen 1994; Voutilainen et al. 1992; Voutilainen and Tapanainen 1993). Maximum versatility of the system is achieved by including into the package all components of analysis, so that the user can perform a large number of different tasks within one single system (Hurskainen 1999a). The cumulative language analysis package can be conceived as a chain of phases, where each phase takes the output of the previous phase as input and performs a defined operation. Therefore, each phase of analysis is a potential cutting point, which can be a basis for a customized application. The result of a morphological analyser, for example, is the phase, on which a spelling checker can be built. A further phase in the chain provides a basis for more advanced language proofing tools, such as phrase structure checkers. These are examples of such customisation where the end user has very few choices.

We may, however, conceive of a system where the end user has access to each phase of the analysis chain and to all its intermediate analysis results. Such possibly useful intermediate phases are the list of tokens (tokeniser), morphological analysis with all possible grammatically correct interpretations of each word-form (useful for studying homonymy), heuristic morphological guesser for assigning analysis for unrecognised words, morphological disambiguation (non-ambiguous morphological interpretation of word-forms), semantic disambiguation (non-ambiguous word sense interpretation of word-forms), syntactic analysis (shallow syntactic parsing or full dependency trees [Tapanainen and Järvinen 1997; Järvinen and Tapanainen 1997; Tapanainen 1999]), etc.

If the user has access to the whole package, he or she is able to make maximum use of the power of each of the programs, and can use texts of one's own choice as input. In order to utilize the versatility of the system, the user environment should facilitate processing in pipe, such as found in Unix/Linux environments. Also several general-purpose utilities, filters and programming languages of Unix add to the usefulness of the system. In other words, a comprehensive analysis system working in a powerful environment facilitates the maximum number of applications.

It becomes obvious from the above that this is not a hit-a-button system, and my impression is that really useful systems cannot be made very simple. But they can be made manageable and fairly easy to learn and use, provided that correct and detailed information is given about the properties and function of each module. Below I shall explicate the ideas given above by describing a language-specific knowledge management system applied to Swahili.

2. Components of the system

The knowledge management system has the following components:

(1) A program that cuts and marks the text into suitable pieces for further processing.

(2) Text normaliser that formalises the text suitable for computational analysis.

(3) Tokeniser that identifies each token in text and verticalises the text.

(4) Morphological analyser that gives each token one or more analyses.

(5) Heuristic guesser that utilizes morphological features and assigns an analysis to unrecognised words.

(6) Morphological disambiguator that resolves morphological ambiguity on the basis of context.

(7) Semantic tagger that tags the word-forms semantically with the help of a look-up dictionary.

(8) Semantic disambiguator that resolves semantic ambiguity with the help of (a) context-sensitive rules, and (b) by utilizing the technique of Self-Ordering Map (SOM).

(9) Syntactic analyser. Two versions are provided, one for shallow syntactic parsing (Constraint Grammar), and another one for constructing full syntactic trees (Dependency Grammar).

(10) General lexical database manager for transforming the result suitable for general dictionary compilation.

(11) Domain-specific database manager for preparing domain-specific dictionaries.

(12) Information extraction based on linguistic analysis.

The system is so constructed that each module is built on top of the previous one. For example, the heuristic guesser (5) can be applied only after the phases 1-4 have been processed. Because all modules, except for (1) and (3), are language-specific, access to the whole package has to be provided for the user.

The system has been under development since 1985, and currently it has major components for morphological analysis (Hurskainen 1992), morphological and semantic disambiguation (Hurskainen 1996), as well as for shallow syntactic mapping (Hurskainen 1999a).

3. Where is the bottleneck?

Although it is generally accepted that knowledge-based systems enhance greatly the performance of knowledge management, very few such systems exist. The bottleneck might not be only the lack of sufficiently advanced modules in the system, but rather the fact that the system is necessarily fairly complex, and that rarely the designer(s) of the system have access to the source code of all the components of the system. It often turns out that the developer of a module finds himself struggling with problems that actually should have been solved by the developer of the previous module in the chain. And if the previous module cannot be corrected, the developers are left struggling with problems in the wrong environment. Work with rule-based disambiguation and syntactic parsing is an example of such work, where the developers should have a possibility to correct or change the morphological parser. There are always bugs in the morphological parser, and such bugs are often found in testing disambiguation rules. Also the coding system might need improvement, and it would be best to fix it in the morphological parser and not patch it up after the analysis has already been performed.

4. Shell scripts vs. user-defined processing

The system sketched above makes it possible to construct the processing chain in several ways, depending on needs. As it is well known, command calls of varying complexity may be stored into shell script files. For example, shell scripts for performing various phases in the process may be constructed so that one script performs phases 1-4, another 1-5, another one again 1-6, etc. It is important to note that the system allows the processing of raw text, and with each call the processing is taken that far as is defined in the shell script.

What is defined in shell scripts may be called also directly from the command prompt. Command calls are, however, usually so complex that the use of shell scripts has become a common practice. However, the shell script does not necessarily always give the desired result. For that reason it is often useful to combine shell scripts and command calls as one consequtive chain of commands. What is important is that the user may build one's own working environment by using the possibilities offered by the operating system (Unix/Linux). This does not exclude the possibility that the basic package distributed to the customers contains a model environment for those who do not have interest or time to develop their own more flexible environment.

5. What format should the source text have?

Corpus texts available to the customers are often encoded in some way. If they do not have part-of-speech tagging, or even more sophisticated encoding, they are encoded at least on the level of document structure, by using SGML, TEI, or XML encoding. It depends very much on the application which type of encoding is useful. If, for instance, the application is aimed at retrieving whole documents according to selected criteria, the document structure encoding is useful, and often sufficient. On the other hand, if the application aims at fine-grained knowledge management, the encoding of the document structure provides only a kind of background level, and each token has to be encoded in respect to several levels of linguistic analysis. Let us look in more detail the usefulness of various formats of source text.

(a) SGML, TEI, and XML encoding

One of these encoding systems, increasingly XML, is used in encoding corpora. Although such encoding is not very useful for some applications, it is not harmful either, because it helps in including or excluding sections of the documents in processing.

(b) Corpora with part-of-speech tagging

For a long time corpora tagged with POS codes have been principal sources of corpus-based linguistic investigation. Tagged corpora have the advantage of transparency and fairly good reliability, because the code set as well as the actual encoding of text is visible, and the encoding has been checked. Its disadvantage is that it is a frozen document. Both the text itself and the encoding are in fixed format, and it allows very little calibration.

(c) Corpora with analysis programs

If the aim is to establish an accurate and efficient knowledge management environment, it is hard to see any other possibility than to provide the customer with a package that contains both the source texts and the analysis programs. And in fact the programs are even more important, because they give the user a possibility to use any texts. In order to make the system user-friendly, the analysis system should be able to accept text in the format where texts are usually available.

When the user has access to all programs in the package, it makes it possible to maintain the same texts in various formats. It is obvious that the ordinary text format is useful in retrieving context for keywords. It is also very likely that the user would like to have the text also in some kind of pre-processed format, so that there is no need to process all phases each time when information is searched. At least a tokenised format, or sentence-per-line format, is a useful format to preserve. In that case, proc-

essing could be started from phase 4. The heaviest part of processing is the section composed of phases 5-9, because they contain the actual linguistic analysis and disambiguation. Because this takes some time, depending on the size of the corpus, there is a temptation to analyse the corpus and preserve the analysed version of corpus as a source for further processing. The disadvantage is that the analysed files tend to become fairly large, in Swahili for example 14 times the size of the original file. The disambiguation process downsizes the result by about 50 per cent. However, if the disambiguated text is saved in zipped format, its size is about the same as the original unzipped text. Therefore, there are practical possibilities to maintain even large texts in analysed format.

6. The optimal compromise

It depends on the skills of the user what kinds of use can be made of the language management system. Testing the performance and accuracy of dictionaries (Hurskainen 1994, 1999b) requires much more from the user than the production of concordances, for example. Taking into account the fact that most users value the ease of use, the various phases of analysis described in (2) above can be packed to the following components:

(a) **Tokeniser** includes the phases 1-3
(b) **Morphological** analyser includes the phases 4-5
(c) **Disambiguator** includes the phases 6-8
(d) **Syntactic analyser** includes the phase 9

These components can be compiled as runtime versions, in which case they cannot be modified by the user but their use is easy and fast. This solution includes the most important language analysis components but still leaves the user freedom to process the result of each phase for one's own needs.

7. Summary

A full language analysis package adapted to a suitable platform enhances an accurate, powerful and versatile working environment for a number of applications and user-defined tasks. The system described here relies heavily on language-specific and rule-based components. Although the basic components are generic, domain-specific applications can be built on the basis of features in the analysis. In Appendix are some extracts from text in various formats.

Appendix

Original format

Nyakati hizi kila mtafiti analazimika kupambana na wafadhili ili apate mbinu za kufanya kazi ya utafiti. (*These times every researcher is forced to fight with sponsors in order to get means to do research work.*)

Morphologically analysed format without disambiguation

```
"<*nyakati>"
        "wakati" N 11/10-PL AR ' time '
"<hizi>"
        "hizi" V IMP AR SV ' put to shame '
        "hizi" V <kwisha AR SV ' put to shame '
        "hi-i" PRON DEM :hV 9/10-PL ' this '
"<kila>"
        "kila" N 7/8-SG IND A [C] [X]
        "kila" ADJ A-UNINFL AR ' every '
"<mtafiti>"
        "tafiti" V SBJN VFIN 1/2-SG3-OBJ AR SV SVO ' do research '
        "tafiti" V SBJN VFIN 1/2-PL2-SP AR SV SVO ' do research '
        "mtafiti" N 1/2-SG AR ' researcher '
"<analazimika>"
        "lazimika" V 1/2-SG3-SP VFIN PR:na AR SV ' be necessary , be forced to ' STAT
"<kupambana>"
        "pambana" V INF SV SVO ' decorate ; fight with ' REC
"<na>"
        "na" CC @CC
        "na" AG-PART
        "na" NA-POSS
        "na" PREP
"<wafadhili>"
        "fadhili" V SBJN VFIN 1/2-PL2-OBJ AR SV SVO ' be generous , donate '
        "fadhili" V SBJN VFIN 1/2-PL3-OBJ AR SV SVO ' be generous , donate '
        "fadhili" V SBJN VFIN 1/2-PL3-SP AR SV SVO ' be generous , donate '
        "fadhili" V 1/2-SG2-SP VFIN PR:a AR SV SVO ' be generous , donate '
        "fadhili" V 3/4-SG-SP VFIN PR:a AR SV SVO ' be generous , donate '
        "fadhili" V 11-SG-SP VFIN PR:a AR SV SVO ' be generous , donate '
        "fadhili" V 1/2-PL3-SP VFIN PR:a AR SV SVO ' be generous , donate '
        "mfadhili" N 1/2-PL AR ' donor , patron'
"<ili>"
        "ili" CONJ **CLB AR ' so that , in order to '
"<apate>"
        "pata" V SBJN VFIN 1/2-SG3-SP SV SVO ' get '
"<mbinu>"
        "mbinu" N 9/10-NI-SG ' means '
        "mbinu" N 9/10-NI-PL ' means '
"<za>"
        "a" GEN-CON 9/10-PL
"<kufanya>"
        "fanya" V INF SV SVO ' do, make '
"<kazi>"
        "kazi" N 9/10-0-SG ' work '
        "kazi" N 9/10-0-PL ' work '
"<ya>"
        "a" GEN-CON 3/4-PL
        "a" GEN-CON 9/10-SG
        "a" GEN-CON 5/6-PL
        "a" 5/6-PL-SP
"<utafiti>"
        "tafiti" V SBJN VFIN 3/4-SG-OBJ AR SV SVO ' do research '
        "tafiti" V SBJN VFIN 11-SG-OBJ AR SV SVO ' do research '
        "tafiti" V SBJN VFIN 1/2-SG2-SP AR SV SVO ' do research '
        "tafiti" V SBJN VFIN 3/4-SG-SP AR SV SVO ' do research '
        "tafiti" V SBJN VFIN 11-SG-SP AR SV SVO ' do research '
        "utafiti" N 11-SG AR HC ' research '
"<.$>"
```

Constraint Grammar parsing, including morphological and semantic disambiguation

```
"<*nyakati>"
        "wakati" N 11/10-PL AR ' time ' @TIME
"<hizi>"
        "hi-i" PRON DEM :hV 9/10-PL ' this ' @<ND
"<kila>"
        "kila" ADJ A-UNINFL AR ' every ' @AD-A>
"<mtafiti>"
        "mtafiti" N 1/2-SG AR ' researcher ' @SUBJ
"<analazimika>"
        "lazimika" V 1/2-SG3-SP VFIN PR:na AR SV ' be forced
to ' STAT @FMAINV
"<kupambana>"
        "pambana" V INF SV SVO ' fight with ' REC
"<na>" @-FMAINV-n
        "na" PREP @PREP>
"<wafadhili>"
        "mfadhili" N 1/2-PL AR ' donor ' @I-OBJ
"<ili>"
        "ili" CONJ **CLB AR ' in order to ' @CS
"<apate>"
        "pata" V SBJN VFIN 1/2-SG3-SP SV SVO ' get '
@FMAINVtr>
"<mbinu>"
        "mbinu" N 9/10-NI-PL ' means ' @OBJ
"<za>"
        "a" GEN-CON 9/10-PL @<NOM
"<kufanya>"
        "fanya" V INF SV SVO ' do ' @-FMAINV-n
"<kazi>"
        "kazi" N 9/10-0-SG ' work ' @OBJ
"<ya>"
        "a" GEN-CON 9/10-SG @<NOM
"<utafiti>"
        "utafiti" N 11-SG AR HC ' research ' @<P
"<.$>"
```

Syntactically analyzed format (FDG)

1	Nyakati	wakati	tmp:>5	@TIME N 11/10-PL AR ' time '
2	hizi	hizi	det:>1	@<ND PRON DEM :hV 9/10-PL ' this '
3	kila	kila	det:>4	@ADJ> ADJ A-UNINFL AR ' every '
4	mtafiti	mtafiti	subj:>5	@SUBJ N 2-SG AR ' researcher '
5	analazimika	lazimika	main:>0	@FMAINV V 2-SG3-SP VFIN PR:na AR SV ' be forced to ' STAT
6	kupambana	pambana	mod:>5	@-FMAINV-n V INF SV SVO ' fight with ' REC
7	na	na	ha:>8	@ADVL PREP ' with '
8	wafadhili	mfadhili	obj:>6	@I-OBJ N 2-PL AR ' sponsor '
9	ili	ili	pm:>10	@CS CONJ AR**CLB ' in order to '
10	apate	pata	cnt:>6	@FMAINVtr> V SBJN VFIN 2-SG3-SP SV SVO ' get '
11	mbinu	mbinu	obj:>10	@OBJ N 9/10-NI-PL ' means '
12	za	za	mod:>11	@<NOM GEN-CON 9/10-PL
13	kufanya	fanya	pcomp:>11	@-FMAINV-n V INF SV SVO ' do '
14	kazi	kazi	obj:>13	@OBJ N 9/10-0-SG ' work '
15	ya	ya	**:>16	@<NOM GEN-CON 9/10-SG
16	utafiti	utafiti	attr:>14	@<P N 11-SG AR HC ' research '
17	.			

Disambiguated format, sorted in order of frequency

```
 1006    "<wakati>" "wakati" N 11-SG AR ' time '
 1050    "<kwamba>" "kwamba" CONJ **CLB ' that '
 1057    "<*mungu>" "*mungu" PROPNAME AN HUM ' God '
 1086    "<vya>" "a" GEN-CON 7/8-PL
 1117    "<ya>" "a" GEN-CON 3/4-PL
 1272    "<ili>" "ili" CONJ **CLB AR ' so that '
 1504    "<watu>" "mtu" N 1/2-PL ' man '
 1563    "<alisema>" "sema" V 1/2-SG3-SP VFIN PAST SV SVO '
say '
 1759    "<kama>" "kama" ADV AR ' like , such as '
 1821    "<na>" "na" PREP ' with , by '
 1827    "<wa>" "a" GEN-CON 3/4-SG
 1879    "<cha>" "a" GEN-CON 7/8-SG
 1896    "<na>" "na" AG-PART ' by '
 2177    "<hiyo>" "hi-o" PRON DEM :hV ASS-OBJ 9/10-SG ' this '
 2708    "<wa>" "a" GEN-CON 1/2-SG
 2923    "<wa>" "a" GEN-CON 1/2-PL
 3063    "<kuwa>" "kuwa" CONJ **CLB ' that '
 3252    "<za>" "a" GEN-CON 9/10-PL
 3343    "<wa>" "a" GEN-CON 11-SG
 3533    "<ya>" "a" GEN-CON 5/6-PL
 3641    "<la>" "a" GEN-CON 5/6-SG
 4819    "<ni>" "ni" DEF-V:ni ' be '
 4917    "<katika>" "katika" PREP ' in, at '
 5798    "<kwa>" "kwa" PREP ' at, to, for '
 7812    "<ya>" "a" GEN-CON 9/10-SG
12835    "<na>" "na" CC ' and '
```

Disambiguated format, sorted according to word-form

```
   17    "<kinywaji>" "kinywaji" N 7/8-SG DER:ji ' drink '
    7    "<kinywa>" "kinywa" N 7/8-SG HC ' throat '
    8    "<kinywani>" "kinywa" N 7/8-SG HC LOC LOC ' throat '
    1    "<kiofisi>" "ofisi" ADV ADV:ki 9/10-0-SG DER:i ' office '
  163    "<*kiongozi>" "kiongozi" N 7/8-SG DER:zi AN ' leader '
  189    "<kiongozi>" "kiongozi" N 7/8-SG DER:zi AN ' leader '
   12    "<kioo>" "kioo" N 7/8-SG 'mirror'
   12    "<kiota>" "kiota" N 7/8-SG ' nest '
    6    "<kipaji>" "kipaji" N 7/8-SG ' talent , gift'
   12    "<kipande>" "kipande" N 7/8-SG ' piece '
   21    "<kipato>" "kipato" N 7/8-SG DER:o ' income '
    2    "<kipato>" "pato" ADV ADV:ki 5a/6-SG DER:o ' income '
   30    "<kipaumbele>" "kipaumbele" N 7/8-SG ' priority '
    2    "<kipawa>" "kipawa" N 7/8-SG ' income '
    5    "<kipengele>" "kipengele" N 7/8-SG ' point , aspect '
```

Disambiguated format, sorted according to lemma

```
   12    "<kiota>" "kiota" N 7/8-SG ' nest '
   23    "<vipaji>" "kipaji" N 7/8-PL ' gift '
    6    "<kipaji>" "kipaji" N 7/8-SG ' gift '
    8    "<vipande>" "kipande" N 7/8-PL ' piece '
   12    "<kipande>" "kipande" N 7/8-SG ' piece '
    3    "<vipandikizi>" "kipandikizi" N 7/8-PL DER:zi HC ' graft '
    6    "<vipato>" "kipato" N 7/8-PL DER:o ' income '
   21    "<kipato>" "kipato" N 7/8-SG DER:o ' income '
    2    "<vipaumbele>" "kipaumbele" N 7/8-PL ' priority '
   30    "<kipaumbele>" "kipaumbele" N 7/8-SG ' priority '
    3    "<vipawa>" "kipawa" N 7/8-PL ' gift '
    2    "<kipawa>" "kipawa" N 7/8-SG ' gift '
   11    "<vipengele>" "kipengele" N 7/8-PL 'point , aspect '
    5    "<kipengele>" "kipengele" N 7/8-SG ' point , aspect '
```

References

Hurskainen, A. (1992). A Two-Level Computer Formalism for the Analysis of Bantu Morphology: An Application to Swahili. Nordic Journal of African Studies 1(1): 87-122.

Hurskainen, A. (1994). Kamusi ya Kiswahili Sanifu in test: A computer system for analyzing dictionaries and for retrieving lexical data. Afrikanistische Arbeitspapiere 37 (Swahili Forum I): 169-179.

Hurskainen, A. (1996). Disambiguation of morphological analysis in Bantu languages. COLING-96, Proceedings of the 16th International Conference on Computational Linguistics, Copenhagen, August 5-9, 1996. Pp. 568-573.

Hurskainen, A. (1999a). SALAMA: Swahili language manager. Nordic Journal of African Studies 8(2): 139-157.

Hurskainen, A. (1999b). Salim K. Bakhressa, Kamusi ya Maana na Matumizi. Nairobi: Oxford University Press. Review article. Journal of African Languages and Linguistics 20.

Järvinen, T. & Tapanainen, P. (1997). Timo Järvinen and Pasi Tapanainen. A Dependency Parser for English. Technical Reports, No. TR-1. Department of General Linguistics. University of Helsinki, 1997.

Karlsson, F. (1990). Constraint Grammar as a framework for parsing running text. In Hans Karlgern (ed.), COLING-90. Papers presented to the 13th International Conference on Computational Linguistics. Vol. 3, pp. 168-173, Helsinki, 1990.

Karlsson, F. (1995a). Designing a parser for unrestricted text. In Karlsson et al (eds.), Constraint Grammar: A Language?Independent System for Parsing Unrestricted Text. Mouton de Gruyter, Berlin. Pp. 1-40.

Karlsson, F. (1995b). The formalism and environment of Constraint Grammar Parsing. In Karlsson et al (eds.), Constraint Grammar: A Language?Independent System for Parsing Unrestricted Text. Mouton de Gruyter, Berlin. Pp. 41-88.

Karlsson, F., A. Voutilainen, J. Heikkilä & A. Anttila (eds.) (1994). Constraint Grammar: A Language-Independent System for Parsing Unrestricted Text. Mouton de Gruyter, Berlin, 1994.

Koskenniemi, K. (1983). Two-level morphology: A general computational model for word-form recognition and production. Publications No. 11. Department of General Linguistics, University of Helsinki, 1983.

Tapanainen, P. (1996). The Constraint Grammar Parser CG-2. Publications No. 27. Department of General Linguistics, University of Helsinki, 1996.

Tapanainen, P. (1999). Parsing in two frameworks: finite state and functional dependency grammar. PhD dissertation. Language technology, Department of General Linguistics, University of Helsinki. http://ethesis.helsinki.fi/julkaisut/hum/yleis/vk/tapanainen/

Tapanainen, P. & Järvinen, T. (1994). Syntactic analysis of natural language using linguistic rules and corpus-based patterns. COLING-94. Papers presented to the 15th International Conference on Computational Linguistics. Vol. 1, pp. 629-634, Kyoto, 1994.

Tapanainen, P. & Järvinen, T. (1997). A non-projective dependency parser. In Proceedings of the 5th Conference of Applied Natural Language Processing, March 31st - April 3rd, Washington D.C., USA. Pp. 64-71.

Voutilainen, A., J. Heikkilä & A. Anttila, (1992). Constraint Grammar of English - A Performance-Oriented Introduction. Publications No. 21. Department of General Linguistics, University of Helsinki, 1992.

Voutilainen, A. & Tapanainen, P. (1993). Ambiguity resolution in a reductionistic parser. In Proceedings of the Sixth Conference of the European Chapter of the Association for Computational Linguistics. EACL-93. pp. 394-403, Utrecht, Netherlands, 1993.

Challenges in MT customization on closed and open text styles

Rémi Zajac
SYSTRAN Software, Inc.
zajac@systransoft.com

Abstract

This paper reports work in progress on two on-going customization projects at Systran. One project targets on-line technical support documentation. This project falls in a domain that has been (and still is) a favorite target for high-quality MT applications. The second project targets open style (on-line) texts on a large set of small domains. We outline and contrast customization issues for these two projects, and present the customization process based on an automated analysis of monolingual corpora.

1. Introduction

Manual versus automatic customization

Customization of MT systems is a problem that has not received much attention. Typically, customization is reduced to the (manual) development of a simple domain-specific dictionary. Complex lexical entries, involving complex subcategorization patterns for example, are excluded; a fortiori, syntactic customization is excluded too.

Most previous work on automated customization make use of a parallel corpus, for example Yamada et als. (1995) and Su et als. (1995, 1999). Of course, example-based systems may be considered fully customized systems (Richardson et als. 2001, Pinkham et als. 2001).

Yamada et als. (1995) present a method to adapt a rule-based MT system to a new domain by using aligned sets of sentences. The method involves the comparison of the MT parse tree (presumably after transfer) with the parse tree of the manually produced translation. A side effect of the comparison is the automatic generation of either bilingual dictionary entries or transfer rules. The interest of the method is not clear since the technical description is rather sketchy and since there is no discussion on the influence of the bilingual corpus on quality improvement. The method seems to be implemented only for simple bilingual lexical equivalences.

Su et als. (1995, 1999) suggest that customizing an MT system can be reduced to learning probabilistic parsing parameters. They use probabilistic learning techniques to select the best parse of a non-deterministic parser. The best parse is the one that gives a translation that is closest to the manually translated sentence (or the one which produces a parse tree that is closest to the parse tree for the manually translated sentence, the paper is unclear on this point). The method does not seem to be implemented.

The best current approaches, providing highest quality results, to fully automatic customization are using example-based techniques built on a substrate of a comprehensive rule-based system as in the MSR-MT project (Richardson et als. 2001, Pinkham et als. 2001). In this approach, there is no distinction between lexical and syntactic customization. What is learned is, in essence, a set of lexicalized transfer rules that may cover entire sentences.

Customization projects at Systran

Systran has recently started several customization projects, for example for on-line technical support documentation as in the Autodesk project (Senellart et als. 2001b). In all these projects, there is no bilingual corpus available. An essential part of the effort is the development of an automated methodology and tools to speed-up customization and to lower costs. A parallel effort that also supports customization projects is directed at the restructuration of the MT architecture towards better modularity and declarativity, and improved performances (Senellart 2001a).

The paper is organized as follows. The next section gives an overview of two on-going customization projects at Systran and outlines specific customization issues. One project falls in a domain that has been (and still is) a favorite target for high-quality MT applications: (on-line) technical support documentation. The second project targets open style (on-line) texts on a large set of small domains. The next two sections present the customization process. Section 3 describes the assessment of customization needs for a given application. This assessment translated into a customization plan, and the customization process itself is described in Section 4. We conclude on several open issues.

2. Two customization projects at Systran

MT applications have been traditionally divided into dissemination and assimilation applications. An

assimilation scenario is an analyst working of foreign documents in open domains and open styles (technical documents as well as web postings and email). These kinds of applications use a generic MT engine with a very large lexical coverage and implementing a non-specific language model that will succeed in providing an average quality on most texts and fail to provide a high quality on most texts. Dissemination applications target the translation of technical documents, typically technical user manuals, for publication. Technical documents cover closed domains and closed styles. Dissemination applications use specific MT engines with targeted lexical coverage and implementing a tailored language model that succeeds in providing a good quality on most texts.

It should be clear that, at present, manual customization could only be envisaged in the dissemination scenario, on a closed domain and a closed text style. For this kind of scenario, some rule-based MT systems have demonstrated high-quality translations. However, no system has ever been able to provide high-quality translation on open domains and open style documents. The consensus in the MT community seems to be that on these kinds of texts, MT can only be used for assimilation scenarios. However, this consensus is based on past experience with rule-based systems: it is still an open question whether example-based or statistical-based systems may achieve high-quality on open domains and open style.

Scenario 1: on-line technical support documentation

The source corpus for this project is medium size (tens of thousands documents, about 2M words). The document style is technical and homogeneous and documents are written by technical writers following specific style guidelines and using an in-house glossary. Sub-types of documents are well defined, for example, FAQs and procedures. The domain is homogeneous covering a single family of products, but very complex, with a high number of concepts and a high number of relationships between concepts. The corpus evolves slowly over time as new documents are added for new versions of existing products. There may be completely new products (still within the same product family) with new concepts and new terminology to cover. Therefore, a continuous monitoring of the document database is necessary to keep track of the emergence of new concepts and terms. At that point, a focalized customization effort is needed for these new documents.

This type of application has been a favorite target for high-quality MT in the past and still remains a favorite (See e.g., Richarson et als. 2001, Pinkham et als. 2001). Translation problems may be reduced to some extend by using a writing style guide. For example, the lexicon can be limited to common words (new words for new concepts only). Grammar and style variation can also limited by the use of technical writing guidelines. Such guidelines could be implemented in a controlled language checker. The idea is the same as for other controlled languages, but with a more modest aim: not solving all MT problems but limiting MT problems to a narrower range.

The main challenge is how to describe the terminology of a large and complex domain: the ontology of the domain is narrow but deep, complex and very specific. For this project, we use a mix of terminology extraction tools (see e.g., Jacquemin 2001). Another issue will be tracking emergence of new concepts as the document database changes over time, and update the terminology accordingly.

Scenario 2: fast changing on-line postings

This customization project blurs the distinction between MT for assimilation and MT for dissemination. In this project, the corpus is very large, millions of postings, with several thousands of new postings every day. The style is very relaxed and makes use of a large range of colorful expressions, with plenty of misspellings and grammar errors, highly variable punctuation usage, as well as uncommon abbreviations. The corpus can be divided into a large number of unrelated sub-domains. Within each domain, the terminology is relatively restricted with a limited number of concepts and a limited number of relationships between concepts. However, there is a very large number of proper names referring to specific products and entities.

In a posting, we can identify several sections that can be categorized into different styles. For example, a description of an object, contractual sections dealing e.g., with payment options or shipping, etc. However, there are always sections that cannot be fitted into any well-defined slot and must be considered free open text. These sections are the most challenging since they are typically argumentative in nature, conveying opinions and trying to convince the reader to adhere to these opinions. These sections are also the ones exhibiting free informal syntax and creative use of language. A specific issue here is to automatically segment a posting into sections that correspond to homogeneous styles and select most appropriate translation parameters for each style.

Each posting addresses a specific domain, and each domain is shallow and relatively simple. Each domain can be managed using simple thesaurus-like management tools à la Wordnet. There are however two main challenges. One is the number of domains (hundreds). Another is the novelty factor that requires constant tracking and customization to match changes in language. In particular, we need to track the emergence of new words (neologisms as well as new names and abbreviations) and new expressions.

3. Evaluation of Customization Needs

Customization assumes a base MT system. The first step in a customization project is to measure the gap between

the quality of this base system and the quality of the targeted customized system in order to evaluate as precisely as possible the customization needs, and to develop a customization plan.

If a bilingual corpus is available, the customization needs could be estimated by evaluating the performance of the base system against the corpus. The translation of the source corpus produces a baseline translation that can be compared and evaluated against the manual translation (target side of the bilingual corpus). An in-depth evaluation of mismatches provides a detailed catalog of customization requirements for both lexical customization and grammatical customization. This evaluation can also assign mismatches to specific sub-grammars of the MT system: NP analysis, verb transfer, relative clause generation, etc. This of course can only be done manually on a small set of documents only.

An indirect but more economical way of evaluating the customization needs is to:

- Measure the performance of the system on a known source baseline corpus, and to
- Evaluate the distance between the baseline corpus and the source corpus.

By using a set of quantitative linguistic indicators, it is possible to estimate the amount of customization needed to achieve a pre-set quality target. The following paragraphs give an overview of an automated customization evaluation process that includes the establishment of a baseline and the construction of a terminological (domain) profile, a lexical profile and a syntactic profile for the source corpus. These profiles are compared to the baseline in order to provide a quantitative estimate of the customization needs (Underwood & Longejan 1999).

A terminological profile of a corpus provides an estimate of the closure of the vocabulary of the corpus as well as the complexity of the domain. The vocabulary closure is measured by counting the number of new terms that appear when a new text is seen (term growth curve). If this curve flattens out rapidly (few new terms appear in newly seen documents), the vocabulary is essentially closed. In such a case, the customized system will probably require little lexical maintenance after delivery.

The complexity of the domain is estimated using the number of technical terms belonging to the domain and the number of interconnections between these terms. The number of interconnections between terms can be estimated by counting the number of syntactic relations between technical terms occurring in the same sentence: predicate-arguments relationships (predicate-object, but also predicate-subjects), and head-modifiers relationships.

New usages of existing words can be detected only as a failure in parsing or translation: parsing and translation failures are collected and sorted by shared lexical units: any lexical unit that occurs in several parsing or translation failures is a potential source of failure and should be investigated.

The syntactic profile of a corpus provides an estimate of the customization work needed on grammars for parsing, transfer and generation. The base system is evaluated on a standardized test suite where test items are categorized by linguistic classes of phenomena. This evaluation provides a detailed account of the strengths and weaknesses of the base system in terms of linguistic categories. We then run the system on the source corpus and extract a frequency profile of morphosyntactic phenomena. This frequency profile is matched to the baseline profile in order to build a customization plan, and to estimate the level of quality that can be achieved for a given level of effort.

4. Customization Process

The customization loop

The source corpus is segmented into translation units (sentences) and the translation units are translated, sorted and stored in the development database. Customization then proceeds along two parallel lines: one customization for terminology and lexical elements, and another for grammar and style. Customization plans are directly derived from terminological, lexical and syntactic profiles. Since any change in a component of the system may have unforeseen impact on other components, and in order to ensure constant progress and test for regression, testing is done continuously and in parallel to development. Continuous testing uses the development database, allows to focus on the main customization issues, to deal rapidly with any potential regression, and to measure progress.

Words that are not in the system dictionaries are extracted during the initial assessment. This initial step also produces a list of lexical units that may be sources of parsing or translation failures, and are therefore candidates for revision. Terminology lists are built using terminology extraction tools. Initial customization proceeds using these lists. As the systems dictionaries are updated, the test database is translated with the updated dictionaries, and new translations are compared with the initial ones. Any translation that shows a difference is added into the review set.

The initial assessment produces a frequency list of morphosyntactic structures that appear in the source corpus. Given that the baseline evaluation identifies the weak areas of the system, this list is converted into a customization plan where the most frequent weak areas are dealt with first (modulo dependencies between grammar modules).

Testing

Testers review new translations as the system is updated. Two different kinds of testing are done, one for terminological and lexical customization, and another for grammar and style. New translations are sorted according to various criteria, including coverage of terminology and difference in translation. For example, new terms added in the dictionaries should be matched and translated for all translation units containing these terms. Two lists are built: one containing matched terms (for simple checking) and one containing unmatched terms (to identify potential dictionary coding problems). A similar testing process is used for structural customization. For example, after working on relative clauses, all sentences containing relative clauses are extracted and divided into a list of changed translations and a list of unchanged translations.

When a translation is changed, it should show an improvement in quality: progress is tracked for any changed translation and quality of new translations is evaluated and recorded. Tracking the quality improvement rate allows us to estimate the cost-effectiveness of the customization effort.

5. Conclusion

The customization method presented in this paper is directly challenged by fully automatic methods using example-based techniques, including for example MSR methods (Richardson et als. 2001, Pinkham et als. 2001). Although manual customization is obviously feasible and can reach acceptable quality, one important issue is the cost-effectiveness of the method: a manual method should be cheaper than using automated customization with a bilingual corpus. Therefore, it should be cheaper or equivalent to the cost of translating a bilingual training corpus (this obviously depends on the minimal size required by the training algorithm). We assume that it may be cheaper when there are multiple target languages as the initial work of analyzing the source corpus and extracting terminology and other specific linguistic pattern can be shared among all target languages. Another important issue is the evolution of the source document database: we need to develop specific methods for tracking changes in language and for updating the language resources at a minimal cost.

To evaluate the accuracy of the estimates of customization effort, and to evaluate the speed and cost-effectiveness of the customization methodology, we are recording a set of quantitative indicators to help us provide accurate estimations. During a customization project, we are tracking cost of creating/customizing lexical entries together with the quality impact of these new of customized entries on the whole corpus. We do the same for grammar customization. Finally, the quality of the MT system is evaluated before customization, and a post-customization evaluation provides a measure of the improvement in quality that has been achieved. Experience over several projects should help us find the most relevant indicators, and obtain accurate estimates from detailed corpus analyses.

6. References

Jacquemin, Christian. 2001. *Spotting and Discovering Terms trough Natural Language Processing*. The MIT Press.

Lalaude, Myriam, Veronika Lux, Sylvie Regnier-Prost. 1998. "Modular controlled language design". CLAW-98, Pittsburgh, PA. Pp103-113.

Pinkham, Jessie, Monica Corston-Oliver, Martine Smets, Martine Petterano. 2001. "Rapid assembly of a large-scale French-English MT system". MT Summit VIII, Santiago de Compostela, Spain. Pp277-282.

Richardson, Stephen, William Dolan, Arul Mezenes, Jessie Pinkham. 2001. "Achieving commercial-quality translation with example-based methods". MT Summit VIII, Santiago de Compostela, Spain. Pp293-298.

Senellart, Jean, Peter Dienes, Tamas Varadi. 2001a. "New generation Systran translation system". MT Summit VIII, Santiago de Compostela, Spain. Pp311-316.

Senellart, Jean, Mirko Plitt, Christophe Bailly, Francoise Cardoso. 2001b. "Resource alignment and implicit transfer". MT Summit VIII, Santiago de Compostela, Spain. Pp317-324.

Su, Keh-Yih, Jing-Shin Chang. 1999. "A customizable, self-learnable parameterized MT system: the next generation". MT Summit VII, Singapore. Pp182-190.

Underwood, Nancy L., Bart Jongejan. 1999. "Profiling Translation Projects". TMI-99, Chester, England. Pp139-149.

Yamada, Setsuo, Hiromi Nakaiwa, Kentaro Ogura, Satoru Ikehara. "A method for automatically adapting an MT system to different domain". TMI-95, Leuven, Belgium. Pp303-310.

Locating and Reusing Sundry NLP Flotsam in an e-Learning Application

Anju Saxena and Lars Borin

Department of Linguistics, Uppsala University,
Box 527, SE-751 20 Uppsala, Sweden
and
Computational Linguistics, Department of Linguistics,
Stockholm University, SE-106 91 Stockholm, Sweden

anju.saxena@ling.uu.se, lars.borin@ling.su.se

Abstract

We describe the background and motivation for an e-learning project—*IT-based Collaborative Learning in Grammar*—where NLP resource reuse has become an important issue. The resources are of several kinds: POS-tagged and syntactically annotated corpora (treebanks), parsing systems and grammar writer's workbenches, and visulization and manipulation tools for linguistically annotated corpora. Our experience thus far has been that although there are a number of such resources available e.g. on the Web, as a rule, numerous incompatibilities and lack of standardization at all levels—markup formats, linguistic annotation schemes, grammatical framework, software APIs, etc.—make the reuse of these resources into a non-trivial endeavor.

0. Preamble: the Setting

It is generally acknowledged that the goal of teaching grammar—especially at the university level—should not primarily be that students memorize definitions of concepts and grammatical constructions, but rather that they understand and learn to recognize different structural patterns. This can hardly be achieved without giving students practical training in the skill of grammatical analysis. Research has shown that hands-on problem-solving is more stimulating and thought-provoking than when the information and results are handed down to the pupils during lectures. Further, our experience has been that students learn about grammatical constructions and phenomena more actively when these constructions are discussed by comparing the system found in their native language with that of another language. An added factor contributing to an active student participation is the choice of the material forming the basis for exercises and group activities, which should preferably be as natural as possible.

With these pedagogical considerations in mind, we formulated a project for realizing a new format for teaching courses in grammar in Linguistics and Computational Linguistics (the ability to reason about grammar and to carry out grammatical analyses of language utterances being necessary prerequisites for all linguistic studies of language and thereby part of the core curriculum of these subjects). In the proposed format interactive practical training and corpus-based exercises comprise an integral part of the students' learning process, giving them the opportunity and incentive to participate more actively in their own learning process. Using IT as a tool for collaborative work allows the students to choose the problem-solving strategy which suits them best, as well as the time and place to work on the problem. A corpus of natural language material for grammatical analysis contributes to a more active participation, as it not only presents the grammatical constructions in their context, but also gives students a greater freedom to approach the material and conduct the investigation from a perspective which suits their individual learning styles. A text corpus consists of naturally occurring language in its natural physical context, since it is made up of complete texts or large text fragments, as opposed to the made-up or isolated single sentences or phrases often used to illustrate grammatical points in linguistics textbooks. This accompanying physical context makes it possible to investigate the textual, discourse-level, functions of the grammatical phenomena.

An outline of the proposed training material is presented below. It has a modular architecture, composed of four types of modules (see Figure 1, below):

1. 'Encyclopedia' module, containing descriptions of grammatical concepts and constructions. Its content will be attuned to the contents of the course and the interactive exercises (as, in their turn, the exercises will be adapted to the 'encyclopedia' contents), and at appropriate places, there will be hyperlinks to interactive exercises dealing with the current topic.

2. 'Text corpus' module, containing at least (a) POS-tagged and syntactically annotated corpora of Swedish, and (b) an annotated corpus of a foreign language. For (a), we will use the SUC and Talbanken annotated Swedish corpora (see below); for (b), we will use a corpus of Kinnauri (a Tibeto-Burman language spoken in India) narratives available on the web (`http://www.ling.uu.se/anjusaxena/corpus.html`; see figure 2), which is hyperlinked to a morpheme dictionary. Further, with the help of a graphic interface students will be able to see a 'map' of how and where one particular morpheme or a word occurs in the corpus (see Olsson and Borin (2000)), providing support in their work on the functions of grammar. The students will work with the

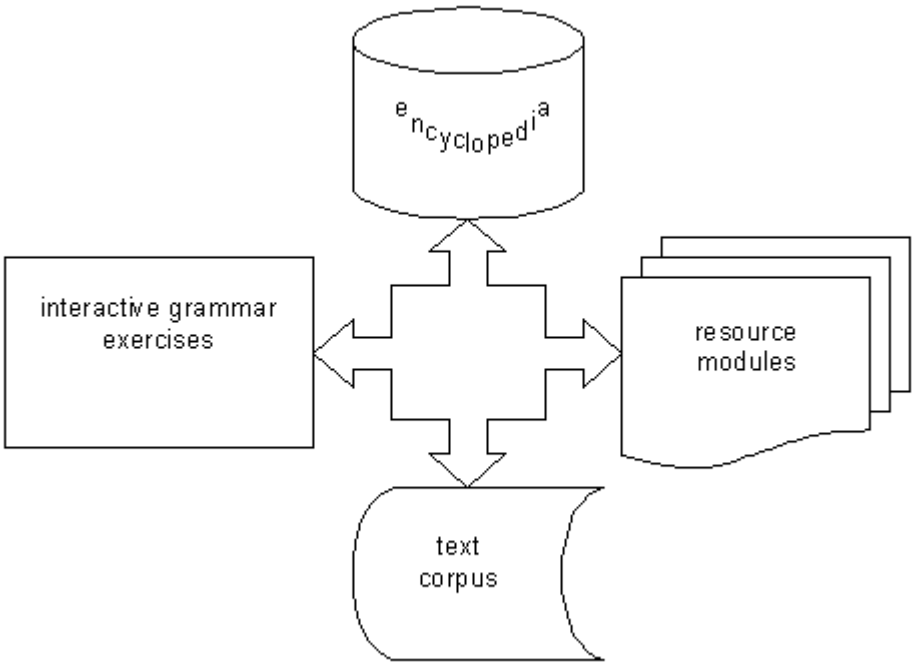


Figure 1: Organization of the proposed IT-supported grammar training application

same corpus as part of their group activities and as part of their examination.

3. 'Interactive exercise' module. Our aim here will be to provide students with a set of exercises, with basic tools for computer-mediated student cooperation in virtual work- groups (a 'spreadsheet' for problem-solving; optional 'step-by-step questions' for the grammatical topic covered; grammar rule writing exercises to be discussed in more detail below), with hyperlinks to the 'encyclopedia', to the 'resources' (see below) and to the annotated corpus of a foreign language (which, in turn, will be hyperlinked to the dictionary; see Saxena (2000)). As part of each theme, students will first discuss the construction during the lecture session, then again while examining the construction in the corpus, and finally also while comparing the results of the corpus-based analysis with the Swedish system and then discussing it in the group. This learning method where the same construction is examined from a number of mutually reinforcing practical and theoretical viewpoints will, hopefully, provide the students with support and incentive in their learning process. Further, the same corpus will be used in grammar courses in first and second semesters, providing grounds for deeper analyses in the second semester than would have been the case.

4. 'Resource' modules will provide a pool of resources for further reading and relevant links to other sites.

The architectural organization of the software proposed here has several advantages, the two most significant ones being extensibility and 'conceptual decentralization'. Extensibility means that new functions can be easily integrated in the application. 'Conceptual decentralization' is especially significant as it allows the possibility of adjusting to individual learning styles. For example, if the student prefers to start out with the 'encyclopedia' material and go from there to the appropriate exercises, when she feels the need to do so, she has that choice. At the same time, the application allows the possibility of starting out at other entry points, e.g., 'interactive exercises', with the option of calling up the relevant 'encyclopedia' material at each instant.

1. The NLP Resource Customization Problem

NLP resource customization has become an issue in this project mainly in connection with module 3 (interactive grammar exercises). It has been our aim from the conception of the project to rely mostly on standard WWW and open-source software—i.e., software which is generally free and where the source code is freely available and modifiable by the user—for implementing the modules. This design philosophy has the advantage of making the application maximally platform-independent, as well as providing a familiar interface—a standard web browser—for students and faculty.

One of the exercises that we have planned for module 3 builds upon a combination of a syntactically annotated corpus (a treebank) and a grammar writer's workbench. The basic premise of the exercises is a further refinement of the idea presented by Borin and Dahllöf (1999). We propose to use grammar rules written by students (using an existing grammar development tool) as search expressions in the

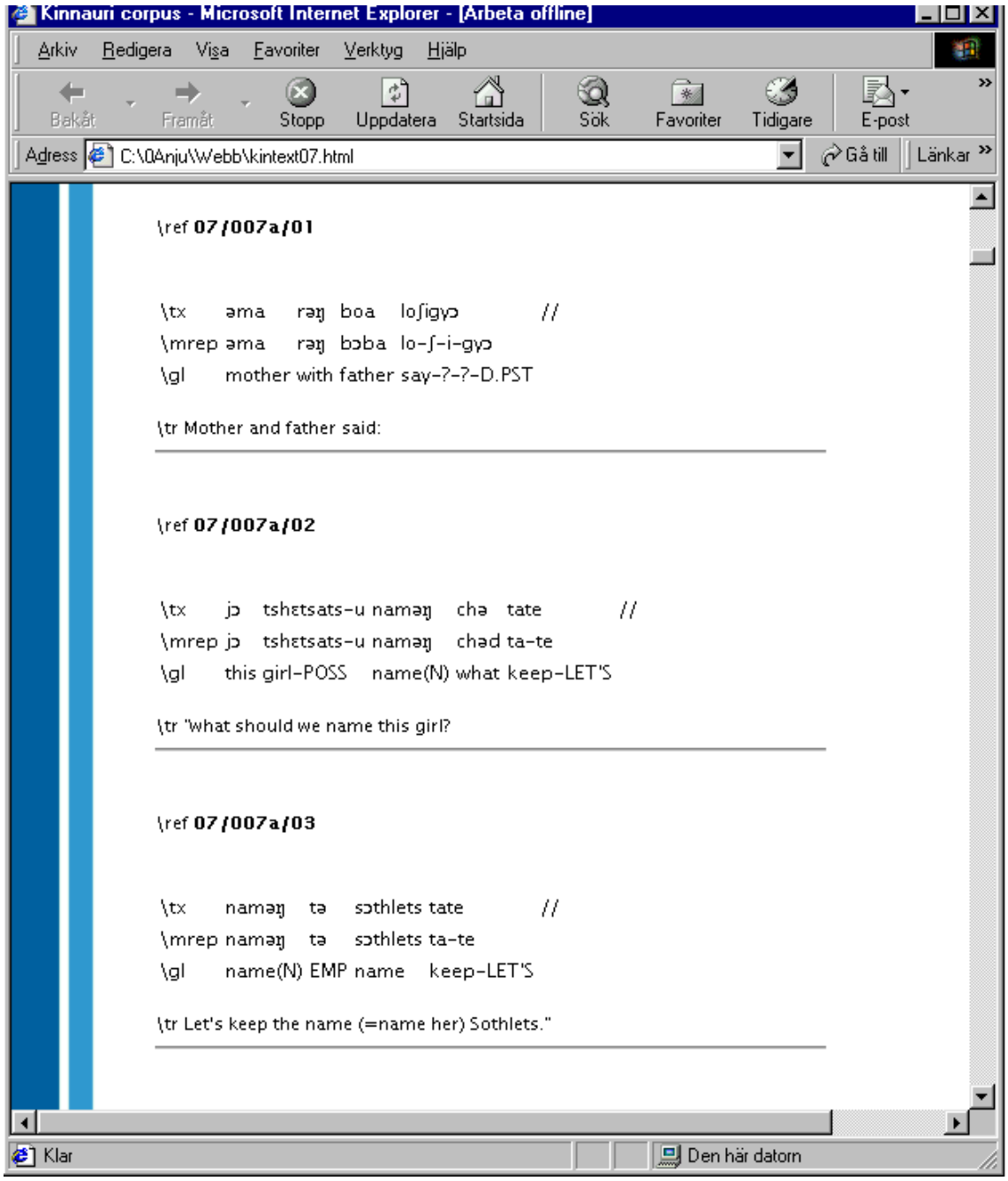


Figure 2: The Kinnauri corpus – Web format

treebank. In its simplest form, the result of the search would be expressed as precision and recall. Given an NP rule formulated by a student, we could automatically tell how many of the (maximal) treebank POS sequences matching the rule actually make up NPs, how many are not NPs, and how many NPs in the treebank are not described by the rule. There are all kinds of conceivable elaborations of this basic scheme, which could be seen as a more linguistically sophisticated parallel to the use of (unannotated) text corpora and concordancing software in so-called data-driven language learning (Flowerdew, 1996).[1] For the Computational Linguistics students, there is the additional advantage of being able to work from the very beginning of their studies with the same kind of tools and resources that they will be using 'for real' after graduating, in their professional life.

What we have found already in this beginning stage of the project, however, is that there are some serious obstacles to using available NLP resources.[2] Mostly, the issues that have arisen in this connection concern (lack of) compatibility and standardization of NLP resources. Some of

[1]The basic idea here is similar to the ICECUP FTF (Fuzzy Tree Fragment) grammatical query system for parsed corpora (Wallis and Nelson, 2000), but with a diffent use and target audience in mind.

[2]Here, we use "NLP resources" as a cover term for both *language resources* and *processing resources* in the terminology adopted by Cunningham (2002).

```
<text id=kl01>
<body>
<p>
<s id=kl01-001>
<c lem='-' msd='FI' n=1>-</c>
<w lem='vilken' msd='DH@0P@S' n=2>Vilka</w>
<w lem='djävla' msd='AQP00N0S' n=3>djävla</w>
<w lem='optimist' msd='NCUPN@IS' n=4>optimister</w>
<c lem=',' msd='FI' n=5>,</c>
<w lem='frusta' msd='V@IIAS' n=6>frustade</w>
<name type=person>
<w lem='Lasse' msd='NP00N@0S' n=7>Lasse</w>
</name>
<c lem='.' msd='FE' n=8>.</c>
</s>

<suctext id=kl01>
<p>
<s id=kl01-001>
<d n=1>-<ana><ps>MID<b>-</d>
<w n=2>Vilka<ana><ps>HD<m>UTR/NEU PLU IND<b>vilken</w>
<w n=3>djävla<ana><ps>JJ<m>POS UTR/NEU SIN/PLU IND/DEF NOM<b>djävla</w>
<w n=4>optimister<ana><ps>NN<m>UTR PLU IND NOM<b>optimist</w>
<d n=5>,<ana><ps>MID<b>,</d>
<w n=6>frustade<ana><ps>VB<m>PRT AKT<b>frusta</w>
<name type=person>
<w n=7>Lasse<ana><ps>PM<m>NOM<b>Lasse</w>
</name>
<d n=8>.<ana><ps>MAD<b>.</d>
</s>
```

Figure 3: Alternative SUC annotation formats

the issues are:

- Differences in fundamental storage and text markup formats. The three corpora that we are considering for use in the project have three different storage formats: (1) The basic format of Saxena's Kinnauri narrative corpus is as a Shoebox database (Buseman and Buseman, 1998) (see figure 4), from which a web version in HTML hyperlinked to a morpheme lexicon was semiautomatically derived (see figure 2); (2) The Stockholm Umeå Corpus (SUC; Ejerhed and Källgren (1997)) comes in an SGML corpus format as specified by the Text Encoding Initiative (TEI; `http://www.tei-c.org/`), and further, there are two different grammatical annotation formats, Parole/EAGLES format (see Monachini and Calzolari (1996)) and SUC format (see figure 3); (3) The Talbanken syntactically annotated corpus of Swedish (Einarsson, 1976a; Einarsson, 1976b; Teleman, 1974) is in an 80-column punch card format with only capital letters (see figure 5).

```
\ref 07/007a/01
\tx @ma r@N boa loshigyO //
\mrep @ma r@N bOba lo-sh-i-gyO
\gl mother with father say-?-?-D.PST
\tr Mother and father said:

\ref 07/007a/02
\tx jO tshEtsats-u nam@N ch@ tate //
\mrep jO tshEtsats-u nam@N ch@d ta-te
\gl this girl-POSS name(N) what keep-LET'S
\tr "what should we name this girl?

\ref 07/007a/03
\tx nam@N t@ sOthlets tate //
\mrep nam@N t@ sOthlets ta -te
\gl name(N) EMP name keep-LET'S
\tr Let's keep the name (=name her) Sothlets."
```

Figure 4: The Kinnauri corpus – Shoebox format

- Differences in POS tagging and syntactic annotations between corpora. The SUC and Talbanken Swedish corpora, although both are POS tagged, use different tagsets, with e.g. SUC having two and Talbanken three subclasses of nouns, and SUC, but not Talbanken, marking number in nouns, etc. Tagset incompatibilities, even within a language is a problem that has been noted in the literature (e.g. by Atwell et al. (2000)), and there has been some work on tools for automatic tagset mapping (e.g. Teufel (1995)). The problems are compounded when several languages are involved,[3] which would be desirable in our setting, where the linguistic subdisciplines of Contrastive Linguistics and Language Typology rely on explicit comparisons between languages at various linguistic levels. As stated above, we know from experience that students learn about grammatical constructions and phenomena more actively when these constructions are discussed by comparing the system found in their native language with that of another language. Preferably, the other language should be one that the students do not know already, as they then will be better able to concentrate on the analysis of 'pure' form. This is why we intend to use the Swedish and Kinnauri corpora together in our first application.
- Differences in POS categories, syntactic categories and grammatical framework between the corpora on

[3]The problem of crosslinguistic mapping of part-of-speech tags has not been extensively discussed in the computational linguistics literature (see Borin (2000); Borin (Forthcoming 2002); Borin and Prütz (2001)), but in general linguistics, there is an extensive literature on the issue of crosslinguistic properties of part-of-speech systems and the universality of proposed parts of speech, which is very relevant in this context (e.g., Anward et al. (1996); Itkonen (2001); Pawley (1993)).

```
P21803012001     0000                    <<      GM          010
P21803012002     *DET                    POOP    SS          010
P21803012003     RÖR                     VVPS    FV          010
P21803012004     SIG                     POXP  AAOO          010
P21803012005     ALLTSÅ                  ABKS    +A          010
P21803012006     OM                      PR      OAPR        010
P21803012007     FALL                    NN      OA          010
P21803012008     1000                    RC      OAET        010
P2180301200910002DÄR                     ABRA    RA          010
P2180301201010002ORSAKEN                 NNDD    SS          010
P218030120111002TILL                     PR      SSETPR      010
P2180301201210002PATIENTENS              NNDDHHGGSSETDT      010
P218030120131002SYMTOM                   NN      SSET        010
P218030120141002INTE                     ABNA    NA          010
P218030120151002PRIMÄRT                  AJ      AA          010
P218030120161002ÄR                       AVPS    FV          010
P2180301201710002ÅDERFÖRKALKNING         VN  SS  SP          010
P2180301201810002100                     +F      +F          010
P218030120191002UTAN                     ++MN    ++          010
P21803012020111002I                      ABMN    +A          010
P218030120211002STÄLLET                  ID      +A          010
P218030120221002BEROR                    VVPS    FV          010
P21803012023111002PÅ                     PR      OAPR        010
P21803012024111002EN                     EN      OADT        010
P218030120251002SANNOLIK                 AJ      OAAT        010
P218030120261002STÖRNING                 VN      OA          010
P21803012027111002I                      PR      OAETPR      010
P2180301202811002CIRKULATIONEN           VNDD    OAET        010
P218030120291002AV                       PR      OAETETPR    010
P218030120301002DEN                      PODP    OAETETDT    010
P218030120311002VÄTSKA                   NN      OAETET      010
P2180301203211002110                     RC      OAETETET    010
P218030120331106SOM                      PORP    SS          010
P2180301203411106OMGER                   VVPSSM  FV          010
P2180301203511106HJÄRNAN                 NNDD    OO          010
P21803012036     .                       IP      IP          010
```

Figure 5: The annotation format in the Talbanken treebank

the one hand and the grammar writing tools and parsers on the other. Thus, the Talbanken corpus uses a fairly traditional Swedish functional grammatical framework, where e.g. NPs are not directly recoverable, but only indirectly, through a combination of syntactic function and lexical category of the head word, while it seems that many, perhaps the majority, of the grammar writing tools freely available on the Web presuppose a phrase structure framework.

- Differences in implementation language, storage model, API, documentation and source code availability, etc. of potentially suitable software. For an excellent overview of these issues, see Olsson (2002).

Thus, we have been forced from the outset to discuss seriously how we are to integrate existing NLP resources in our application, as well as how to make the application itself extensible, so that e.g. new language corpora or new annotations can be added.[4]

2. Taking Stock and Looking Ahead

We are attempting to reuse NLP resources originally meant for NLP research—both *language resources* (notably annotated text corpora) and *processing resources* (the most important being parsers and grammar writing tools)—in an e-learning application for IT-based collaborative learning in grammar courses for Linguistics and Computational Linguistics university students. At the moment, we are locating and evaluating[5] NLP resources, mainly on the web, for the corpus-based interactive grammar exercises. As the corpora are in place already, we are now evaluating tools for the manipulation and visualization of corpus data, parsing systems, and grammar writing environments (workbenches), which raises a number of compatibility/standardization issues that need to be resolved. These compatibility/standardization issues point in two directions simultaneously, as it were:

1. backwards: How can we integrate in our application, with the least amount of effort, existing NLP resources of the kind that we need?

2. forwards: How can we ensure that we ourselves, as well as others, will be able in the future to modify the existing NLP resources, or add new ones, in the framework that we define?

The preliminary answers to these two questions are as follows.

There does not seem to be a simple answer to the first question. Generally, we think that it is more desirable to be able to reuse existing language resources—i.e., texts and corpora, lexicons, and the like—than processing

[4]Courses in Hindi and Turkish at Uppsala University will be used as testbeds during the third year of the project, based on relevant Hindi and Turkish corpus resources.

[5]The evaluation is to be mainly pedagogical, i.e. we will ask ourselves whether a particular resource will be suitable for the pedagogical framework that we have adopted for teaching grammar. However, usability—as the term is used in Human–Computer Interaction research—will also be an important evaluation criterion, as well as the the estimated effort needed to adapt the resource for our needs. See Hammarström (Forthcoming 2002) for details.

resources—in our case first and foremost grammar writing and processing environments—for the pragmatic reasons that

- constructing an annotated corpus from scratch is likely to be a much larger effort than building a grammar writing environment;
- standardization efforts have progressed further particularly in the realm of POS tagged language corpus resources than in the case of language processing resources (Monachini and Calzolari, 1996; Bird et al., 2000; Ide et al., 2000; Cotton and Bird, 2002) (and treebank formats; see Atwell et al. (2000)), although, as a rule, their use in computer-assisted language learning applications has not been considered in this connection (Borin, 2002).

Hence, we aim at being able to handle at least POS-tagged corpora using the EAGLES/Parole tag scheme and marked-up according to the TEI/CES SGML or TEI/XCES XML language corpus formats (thus recognizing, e.g., the SUC Parole format without special preprocessing).

As for the second question, it too, is easier to answer for language resources. Here, we will harmonize the underlying corpus formats with other ongoing projects in our departments,[6] while simultaneously endeavoring to conform to standards that are being worked out in the NLP community. This means that we will undertake the conversion of the Kinnauri and Talbanken corpora into this format, and that in due course we plan to make the corpora generally available in the new format.

As far as 'grammar writer's workbenches' are concerned, we have not yet been able to find a ready-made environment user-friendly enough (for our Linguistics students) and bug-free enough to be immediately useful for our purposes. Thus, it seems likely that we will have to put in some development effort in this area. If this turns out to be the case, the most likely kind of workbench that we will modify or build, will be one within the general paradigm of unification-based feature structure grammar. The evaluation of these systems is still ongoing, however (Hammarström, Forthcoming 2002).

3. Acknowledgements

The work described here forms part of the project *IT-based Collaborative Learning in Grammar*, a collaboration between the universities in Uppsala and Stockholm, funded by the Swedish Agency for Distance Education (DISTUM), for the three years 2002–2004. Anju Saxena is the principal investigator for the project. See also `http://www.ling.uu.se/anjusaxena/distum.html`.

[6] We will strive to be compatible with the corpus format developed in the CROSSCHECK (`http://www.nada.kth.se/theory/projects/xcheck/`), SVANTE (`http://www.ling.uu.se/lars/SVANTE/`) and *ASU availability* projects, in all of which formats and tools for Swedish *learner corpora* (see Granger (1998)) are being developed. The basic corpus format will adhere closely to XCES, with 'standoff' linguistic annotation (Ide et al., 2000).

4. References

Jan Anward, Edith Moravcsik, and Leon Stassen. 1996. Parts of speech: a challenge for typology. *Linguistic Typology*, 1(2):167–183.

Eric Atwell, George Demetriou, John Hughes, Amanda Schiffrin, Clive Souter, and Sean Wilcock. 2000. Comparing linguistic annotation schemes for English corpora. In Anne Abeille, Torsten Brants, and Hans Uszkoreit, editors, *Proceedings of the Workshop on Linguistically Interpreted Corpora. LINC-2000*, pages 1–10. Held at the Centre Universitaire, Luxembourg, August 6, 2000.

Steven Bird, David Day, John Garofolo, John Henderson, Christophe Laprun, and Mark Liberman. 2000. ATLAS: a flexible and extensible architecture for linguistic annotation. In *Proceedings of LREC 2000*, pages 1699–1706, Athens. ELRA.

Lars Borin and Mats Dahllöf. 1999. A corpus-based grammar tutor for Education in Language and Speech Technology. In *EACL'99. Computer and Internet Supported Education in Language and Speech Technology. Proceedings of a Workshop Sponsored by ELSNET and The Association for Computational Linguistics*, pages 36–43, Bergen. University of Bergen.

Lars Borin and Klas Prütz. 2001. Through a glass darkly: Part of speech distribution in original and translated text. In Walter Daelemans, Khalil Sima'an, Jorn Veenstra, and Jakub Zavrel, editors, *Computational Linguistics in the Netherlands 2000*, pages 30–44. Rodopi, Amsterdam.

Lars Borin. 2000. Enhancing tagging performance by combining knowledge sources. In Gunilla Byrman, Hans Lindquist, and Magnus Levin, editors, *Korpusar i forskning och undervisning. Corpora in Research and Teaching*, pages 19–31, Växjö Universitet, Växjö. ASLA, ASLA.

Lars Borin. 2002. Where will the standards for intelligent computer-assisted language learning come from? In *Proceedings of LREC 2002 workshop on International Standards of Terminology and Language Resource Management*. To appear.

Lars Borin. Forthcoming 2002. Alignment and tagging. In Lars Borin, editor, *Parallel Corpora, Parallel Worlds. Selected Papers from a Symposium on Parallel and Comparable Corpora at Uppsala University, Sweden, 22–23 April, 1999*. Rodopi, Amsterdam.

Alan Buseman and Karen Buseman, 1998. *The Linguist's Shoebox for Windows and Macintosh.* Summer Institute of Linguistics, Waxhaw, North Carolina:.

Scott Cotton and Steven Bird. 2002. An integrated framework for treebanks and multilayer annotations. In *Proceedings of LREC 2002*, Las Palmas. ELRA. To appear.

Hamish Cunningham. 2002. GATE, a general architecture for text engineering. *Computers and the Humanities*, 36:223–254.

Jan Einarsson. 1976a. Talbankens skriftspråkskonkordans. Corpus on CD-ROM.

Jan Einarsson. 1976b. Talbankens talspråkskonkordans. Corpus on CD-ROM.

Eva Ejerhed and Gunnel Källgren. 1997. Stockholm Umeå

Corpus version 1.0, SUC 1.0. Department of Linguistics, Umeå University.

John Flowerdew. 1996. Concordancing in language learning. In Martha C. Pennington, editor, *The Power of CALL*, pages 97–113. Athelstan, Houston, Texas.

Sylviane Granger, editor. 1998. *Learner English on Computer*. Longman, London.

Harald Hammarström. Forthcoming 2002. Overview of IT-based tools for learning and training grammar. Project report, IT-based Collaborative Learning in Grammar. Department of Linguistics, Uppsala University.

Nancy Ide, Patrice Bonhomme, and Laurent Romary. 2000. XCES: an XML-based encoding standard for linguistic corpora. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC2000)*, pages 825–830, Athens. ELRA.

Esa Itkonen. 2001. Concerning the universality of the noun vs. verb distinction. *SKY Journal of Linguistics*, 14:75–86.

Monica Monachini and Nicoletta Calzolari. 1996. Synopsis and comparison of morphosyntactic phenomena encoded in lexicons and corpora. a common proposal and applications to European languages. EAGLES Document EAG-CLWG-MORPHOSYN/R.

Leif-Jöran Olsson and Lars Borin. 2000. A web-based tool for exploring translation equivalents on word and sentence level in multilingual parallel corpora. In *Erikoiskielet ja kännösteoria – Fackspråk och översättningsteori – LSP and Theory of Translation. 20th VAKKI Symposium*, pages 76–84, Vaasa, Finland. University of Vaasa.

Fredrik Olsson. 2002. *Requirements and Design Considerations for an Open and General Architecture for Information Refinement*. Number 35 in Reports from Uppsala University, Department of Linguistics, RUUL. Uppsala University, Department of Linguistics.

Andrew Pawley. 1993. A language which defies description by ordinary means. In W. A. Foley, editor, *The Role of Theory in Language Description*, pages 87–129. Mouton de Gruyter, Berlin.

Anju Saxena. 2000. Corpora of lesser-known languages on the internet: A pedagogical tool for the teaching of syntax. Paper presented at the workshop on IT inom språkundervisningen. Uppsala University. `http://www.ling.uu.se/anjusaxena/symposium0303.html`.

Ulf Teleman. 1974. *Manual för grammatisk beskrivning av talad och skriven svenska*. Liber, Lund.

Simone Teufel. 1995. A support tool for tagset mapping. In *Proceedings of SIGDAT 1995. Workshop in connection with EACL 95*, Dublin. Association for Computational Linguistics.

Sean Wallis and Gerry Nelson. 2000. The FTF home pages. WWW: `http://www.ucl.ac.uk/english-usage/ftfs/faqs.htm`. Accessed on 10 April 2002.

We thank The MITRE Corporation for their administrative support of the workshop.
We thank the Advanced Research and Development Activity (ARDA) Northeast Regional Research Center (NRRC) for their technical support in the organization of the workshop.

nrrc.mitre.org

The MITRE Corporation
202 Burlington Road
Bedford, MA 01730

MITRE
www.mitre.org

Question Answering: Strategy and Resources

Workshop Program

Tuesday May 28, 2002

Palacio de Congreso de Canarias

8:00 a.m. Welcome and Introduction
Mark Maybury, The MITRE Corporation, USA

8:15 a.m. Invited Keynote – What's the Next Big Thing in Question Answering?
John Lowe, UC Berkeley and formerly Ask Jeeves, Inc.

QA Evaluation

9:00 a.m. The Evaluation of Question Answering Systems:
Lessons Learned from the TREC QA Track
Ellen M. Voorhees, National Institute of Standards and Technology, USA

9:25 am Why are People Asking these Questions?
A Call for Bringing Situation into Question-Answering System Evaluation
Elizabeth D. Liddy, Syracuse University, USA

9:50 am A Curriculum-based Approach to a QA Roadmap
John Prager, IBM, USA

10:15 am Evaluating QA Systems on Multiple Dimensions
Eric Nyberg and Teruko Mitamura, Carneige Mellon University, USA

10:40 am Evaluation Roadmap Discussion
All

11:00 – 11:20 a.m. Morning Break

11:20 am QA Roadmap
All

13:00 p.m. Lunch and Demos

Inference

14:30 p.m. Inference in Question Answering
Bonnie Webber, University of Edinburgh, Scotland
Claire Gardent, CNRS-LORIA, France, and
Johan Bos, University of Edinburgh, Scotland

Applications

14:55 p.m. The Challenge of Technical Text
Fabio Rinaldi, James Dowdall, and Michael Hess,
University of Zurich, Switzerland

15:20 p.m. Question Answering in the Infosphere: Semantic Interoperability and Lexicon Development
Paul Thompson and Steven Lulich, Dartmouth College, USA

Multilingual and Multiperspective QA

15:45 p.m. Summarization Based Japanese Question and Answering System for Newspaper Articles
Yohei Seki and Ken'ichi Harada, Keio University, Japan

16:10 p.m. Multiple Perspective and Temporal Question Answering
James Pusteyovsky, Brandeis University, USA,
Janice Wiebe, University of Pittsburgh, and
Mark Maybury, The MITRE Corporation, USA

16:35 - 17:00 p.m. Afternoon Break

17:00 p.m. Final Group Roadmap Session on Question Answering
All

19:00 p.m. Close

Lunch Time Demos

Question Answering system for POLISH (POLINT)
Zygmunt Vetulani, Adam Mickiewicz University, Poland

QA from Technical Manuals
Fabio Rinaldi, James Dowdall, and Michael Hess, University of Zurich, Switzerland

Statistical Web-based Question Answering
Drago Radev, University of Michigan, USA

Table of Contents

Table of Contents (Concluded)

Preface

Effective question answering is crucial for proper human-system interaction, and systems that can answer questions help to realise the artificial intelligence dream of a machine as a collaborative agent. Question answering draws on many capabilities including information retrieval, language processing, and human computer interaction. Effective question interpretation and answer generation require technologies that index, retrieve, transcribe, extract, translate, and summarize. Question answering can occur in multilingual, multimedia, and multiparty environments. The applicability of question answering ranges across all domains and tasks including learning, playing and conducting business.

Topics in the call for papers, listed in its entirety at www.lrec-conf.org/lrec2002/lrec/wksh/QuestionAnswering.html, included but were not limited to:

- Roadmaps for question answering language resources (LR) and scientific algorithm developments
- Existing question answering language resources
- Guidelines, standards, specifications, models and best practices for question answering LR
- Methods, tools, and procedures for the acquisition, creation, management, access, distribution, and use of question answering LR
- LR and evaluation and benchmarking of question answering systems and algorithms for tasks including:
  - Advanced question analysis
  - Answer discovery and integration
  - Answer explanation and presentation generation
  - Interactive question answering
- LR and evaluation methods for advanced question answering challenges, including but not limited to:
  - Question answering from heterogeneous (structure, unstructured, semi-structured) sources.
  - Multimedia (e.g., text, graphics, audio, video) and Multimodal (i.e., auditory, visual) question answering
  - Multilingual question answering
  - Answering questions from multiple perspectives (e.g, political/economic/legal, local/national/international)
- Question answering components, architectures or instrumentation that facilities evaluation

This one day workshop aims to refine a roadmap (www-nlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc) for question answering applications and the methods for the creation and evaluation of resources for the next decade in support of these systems. The workshop will draw upon research in the TREC Q&A track, the AQUAINT program, and efforts planned for the ARDA Northeast Regional Research Center (NRRC). Participants will help formulate grand challenge problems, discuss possible data sets and/or evaluation metrics/methods, articulate the role of and necessary advances in resources and evaluation to solve these challenges, as well as strategize jointly about the most effective and efficient path forward. Possible joint products arising from the workshop include:

- A list of existing resources and ones under development (with planned release dates)
- Joint formulation of a Q&A roadmap, motivated by ARDA's roadmap (www-nlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc)
- List of evaluation methods and benchmarks of question answering systems
- List of unresolved research problems and/or areas in question answering
- Shared knowledge of research groups and efforts

Table 1 below lists the papers included in the workshop, the primary focus of the article, question answering issues addressed in the papers, and the kinds of sources focussed on.

TABLE 1. Overview of Contributions

Primary Focus	Title	Technical Issues Addressed	Sources	Author(s)
Evaluation	The Evaluation of Question Answering Systems: Lessons Learned from the TREC QA Track	Evaluation, benchmarking, TREC, existing resources	Newspapers	Ellen Voorhees
Evaluation	Why are People Asking these Questions? A Call for Bringing Situation into Question-Answering System Evaluation	Evaluation, Application	Statistical tables; Mechanical Engineering Papers; Web Sites	Elizabeth Liddy
Evaluation	A Curriculum-based Approach to a QA Roadmap	Question answering, roadmap, evaluation, resources, computational linguistics	documents	John Prager
Evaluation	Evaluating QA Systems on Multiple Dimensions	Ambiguity resolution, evaluation methodology	TREC QA track corpora (Future: Chinese and Japanese newswire)	Eric Nyberg, Teruko Mitamura
Inference	Inference in Question Answering	Inference, question-answering, test suites	documents	Bonnie Webber, Claire Gardent, Johan Bos
Applications	The Challenge of Technical Text	Question Answering, Technical Domains, Technical Terminology, XML	Technical Manuals	Michael Hess, James Dowdall, Fabio Rinaldi
Applications	Question Answering in the Infosphere: Semantic Interoperability and Lexicon Development	question answering systems, query optimization, semantic interoperability, lexicons, connectivistic databases	Sensors (Plans to address documents)	Steven Lulich, Paul Thompson
Multiperspective and Temporal	Multiple Perspective and Temporal Question Answering	question answering systems, multiperspectives, temporal expressions, events	Newspapers	James Pustejovsky, Jan Wiebe, Mark Maybury
Multilingual	Summarization Based Japanese Question and Answering System for Newspaper Articles	Japanese Q A System, summarization technique, information fusion from multiple newspaper articles	Newspapers	Yohe Seki, Ken'ichi Harada
Multilingual	Question Answering system for POLISH (POLINT) and its language resources	question answering, language resources, grammars, dialogue corpora, Polish language	Question-answer corpus	Zygmunt Vetulani

Any international workshop demands the selfless contributions of many individuals. We first thank the authors and participants for their important contributions. We next thank the Organizing Committee for their time and effort in providing detailed and high quality reviews and counsel. And we thank Paula MacDonald at MITRE for her tireless and excellent administrative workshop support.

Workshop Organiser

Mark Maybury
The MITRE Corporation
maybury@mitre.org

Workshop Program Committee

Sanda Harabagiu
University of Texas at Austin
sanda@cs.utexas.edu

Liz Liddy
University of Syracuse
liddy@syr.edu

John Prange
Advanced Research and Development Activity (ARDA)
jprange@nsa.gov

Karen Sparck Jones
University of Cambridge
sparckjones@cl.cam.ac.uk

Ellen Voorhees
National Institute of Standards and Technology (NIST)
ellen.voorhees@nist.gov

Author Index

Invited Keynote

What's the Next Big Thing in Question Answering?

John B. Lowe*
UC Berkeley / LACITO Paris / Formerly of Ask Jeeves, Inc. and W3C AC

Question answering as a computational craft has been around just long enough to have a colorful history and a track record of successes and failures. This checkered past provides object lessons and touchstones in the quest for an effective roadmap for further research.

Early attempts to answer questions by computer -- valiant, creative, and ambitious -- enjoyed limited success due to a number of constraints both foreseen and unforeseen. The importance of certain now well-understood principles governing conversation (e.g. Austin 1962, Grice 1957, 1969, Searle 1969, Dreyfus 1972, 1979) and indeed linguistics generally (Harris 1995, Lakoff 1989) were only dimly appreciated three or four decades ago. Computational resources, both hard and soft, were scarce -- NLP and IR accessories (tokenizers, POS taggers, and parsers, for example) which today are taken for granted often did not exist or had to be re-invented in each instance. While the early research program did not always realize its ambitious goals, a large number of approaches were tried and to some extent evaluated. Much was learned.

The advent of the web and other technological developments of the mid- to late-nineties injected new vigor into the question-answering field. For the first time in a long time commercial and intellectual opportunity was seen in open-domain question answering and a number of companies, both startups and established firms, rushed into the fray.

Yet another wave of twenty-first century technology promises to both enable and challenge future QA systems. The first of these is the so-called Semantic Web. A gleam in the eye of the web inventor Tim Berners-Lee and others for some time now (Dertouzos 2001), the Semantic Web is to be partially enabled by Web Services, another initiative which is now the subject of a turf war between major players in information services.

If the mark of a mature research programme is a group of focused researchers working together within an accepted paradigm judged on the basis of impartial evaluation criteria then the question answering field is mature. Nevertheless, even the best systems today handle only a few classes of the known range. Furthermore, the prospects for general solutions are anxiously dependent on developments in other fields as disparate as linguistic semantics, sociolinguistics, and knowledge representation (KR).

The roadmap presented as part of this workshop demonstrates the maturity of the field. It also indicates that question answering is at a crossroads and how important it is to pick the right path. As part of my talk, I will critique some of the major points and suggestions made therein, with an eye to clarifying their achievability and the consequences of success.

* Department of Linguistics
1203 Dwinelle Hall
University of California at Berkeley
Berkeley, CA 94720-2650
voice: (510) 643-9910
fax: (208) 567-2107
email: jblowe@socrates.berkeley.edu

The Evaluation of Question Answering Systems: Lessons Learned from the TREC QA Track

Ellen M. Voorhees

National Institute of Standards and Technology
100 Bureau Dr. STOP 8940
Gaithersburg, MD 20899-8940
ellen.voorhees@nist.gov

Abstract

The TREC question answering (QA) track was the first large-scale evaluation of open-domain question answering systems. In addition to successfully fostering research on the QA task, the track has also been used to investigate appropriate evaluation methodologies for question answering systems. This paper gives a brief history of the TREC QA track, motivating the decisions made in its implementation and summarizing the results. The lessons learned from the track will be used to evolve new QA evaluations for both the track and the ARDA AQUAINT program.

1. The TREC QA Task

TREC is a workshop series designed to provide the infrastructure required for large-scale evaluation of text retrieval and related technologies (National Institute of Standards and Technology, 2002). A "track" for the investigation of question answering systems was introduced into TREC-8 in 1999, and has been run each year since then for a total of three times to date.

The original motivation for the track was to foster research that would move retrieval systems closer to *information* retrieval systems rather than *document* retrieval systems. Document retrieval systems' ability to work in any domain was considered an important feature to maintain. At the same time, the technology that had been developed by the information extraction community appeared ready to exploit. Thus the task for the TREC-8 QA track was defined such that both the information retrieval and the information extraction communities could work on a common problem. The task was very similar to that used in the MURAX system (Kupiec, 1993), which used an on-line encyclopedia as a source of answers for closed-class questions, except that the answers were to be found in a large corpus of documents rather than an encyclopedia. Since the documents consisted mostly of newswire and newspaper articles, the domain was essentially unconstrained. However, only closed-class questions were used, so answers were generally entities familiar to information extraction systems.

Participants were given a document collection and a test set of questions. The questions were fact-based, short-answer questions such as *How many calories are there in a Big Mac?* and *Where is the Taj Mahal?*. Each question was guaranteed to have at least one document in the collection that answered it. For each question, participants returned a ranked list of five [*document-id*, *answer-string*] pairs such that each answer string was believed to contain an answer to the question. Answer strings were limited to either 50 or 250 bytes depending on the run type. Human assessors read each string and made a decision as to whether or not the string contained an answer to the question in the context provided by the document. Individual questions received a score equal to the reciprocal of the rank at which the first correct response was returned (or 0 if none of the five responses contained a correct answer). The score for a run was the mean of the individual questions' reciprocal ranks.

2. Evaluation

The TREC QA evaluations have been based on the assumption that different people will have different ideas of what constitutes a correct answer. This assumption was demonstrated to be true during the TREC-8 evaluation. For TREC-8, each question was independently judged by three different assessors. The separate judgments were combined into a single judgment set through adjudication for the official track evaluation, but the individual judgments were used to measure the effect of differences in judgments on systems' scores. Assessors had legitimate differences of opinion as to what constituted an acceptable answer even for the deliberately constrained questions used in the track. Two prime examples of where such differences arise are the completeness of names and the granularity of dates and locations.

Fortunately, as with document retrieval evaluation, the relative scores between QA systems remain stable despite differences in the judgments used to evaluate them (Voorhees and Tice, 2000). The lack of a definitive answer key does mean that evaluation scores are only meaningful in relation to other scores on the same data set. Absolute scores *do* change if you use a different set of judges, or a different set of questions. However, this is an unavoidable characteristic of QA evaluation. Since assessors' opinions of correctness differ, the eventual end users of the QA systems will have similar differences of opinion, and an evaluation of the technology must accommodate these differences.

A [*document-id*, *answer-string*] pair was judged correct if, in the opinion of the NIST assessor, the answer-string contained an answer to the question, the answer-string was responsive to the question, and the document supported the answer. If the answer-string was responsive and contained a correct answer, but the document did not support that answer, the pair was judged "Not supported" (except in TREC-8 where it was marked correct). Otherwise, the

pair was judged incorrect. Requiring that the answer string be responsive to the question addressed a variety of issues. Answer strings that contained multiple entities of the same semantic category as the correct answer but did not indicate which of those entities was the actual answer (e.g., a list of names in response to a who question) were judged as incorrect. Certain punctuation and units were also required. Thus "5 5 billion" was not an acceptable substitute for "5.5 billion", nor was "500" acceptable when the correct answer was "$500". Finally, unless the question specifically stated otherwise, correct responses for questions about a famous entity had to refer to *the* famous entity and not to imitations, copies, etc. For example, two TREC-8 questions asked for the height of the Matterhorn (i.e., the Alp) and the replica of the Matterhorn at Disneyland. Correct responses for one of these questions were incorrect for the other.

One of the problems of judging entire strings for correctness is that the resulting judgments do not create a reusable test collection. The primary way TREC has been successful in improving document retrieval performance is by creating appropriate test collections for researchers to use when developing their systems. While creating a large collection can be time-consuming and expensive, once it is created researchers can automatically evaluate the effectiveness of a retrieval run. Unfortunately, different QA runs very seldom return exactly the same answer strings, and it is quite difficult to determine automatically whether the difference between a new string and a judged string is significant with respect to the correctness of the answer. Word recall (Breck et al., 2000) and answer patterns (Voorhees and Tice, 2000) have been suggested as ways of approximating a reusable test collection. These approximations have been well-correlated with human judgments in tests to date, but they mis-judge broad classes of responses. Since the mis-judged classes are frequently the cases that are difficult for the original systems being evaluated, the approximations are likely to be less useful as QA systems continue to improve. Nonetheless, they are currently helpful for providing quick feedback as to the relative quality of alternate question answering techniques.

3. Retrieval Results

The most accurate of the TREC-8 systems were able to answer more than 2/3 of the questions. When an answer was found at all, it was likely to be highly ranked. Not surprisingly, allowing 250 bytes in a response is an easier task than limiting responses to 50 bytes. Indeed, traditional passage retrieval techniques are effective when a response as long as 250 bytes is acceptable (Singhal et al., 2000).

Most participants used a version of the following general approach to the question answering problem. The system first attempted to classify a question according to the type of its answer as suggested by its question word. For example, a question that begins with "who" implies a person or an organization is being sought, and a question beginning with "when" implies a time designation is needed. Next, the system retrieved a small portion of the document collection using standard text retrieval technology and the question as the query. The system performed a shallow parse of the returned documents to detect entities of the same type as the answer. If an entity of the required type was found sufficiently close to the question's words, the system returned that entity as the response. If no appropriate answer type was found, the system fell back to best-matching-passage techniques.

The absolute value of the scores for TREC-9 systems was lower than for TREC-8, but in fact the systems were significantly improved (the TREC-9 task was much more difficult as described below). The improvement in QA systems came from refinements to the individual steps of the general strategy described above rather than an entirely new approach. TREC-9 systems were better at classifying questions as to the expected answer type, and used a wider variety of methods for finding the entailed answer types in retrieved passages. Many systems used WordNet (Fellbaum, 1998) as a source of related words for the initial query and as a means of determining whether an entity extracted from a passage matched the required answer type.

Many systems continued to refine this approach in the TREC 2001 track. However, the TREC 2001 track also saw a resurgence of approaches that relied on simpler pattern matching methods using very large corpora (generally the web) rather than sophisticated language processing. The idea exploited in the massive data approach is the fact that in a large enough data source a correct answer will usually be repeated often enough to distinguish it from the noise that happens to occasionally match simple patterns.

4. Creating a Question Set

The manner in which the test set of questions was assembled has had a big effect on the results of the QA evaluations. In TREC-8, the majority of the questions were created expressly for the track, and thus tended to be back-formulations of a statement in a document. In TREC-9, the questions were selected from an Encarta log that contained actual questions, and a raw Excite log. Since the raw Excite log did not contain many grammatically well-formed questions, NIST staff used the Excite log as a source of ideas for actual questions. All the questions were created without looking at any documents. The resulting test set of questions was much more difficult than the TREC-8 set, mainly because the TREC-9 set contained many more high-level questions such as *Who is Colin Powell?*. For the TREC 2001 track, the source of the questions was again web logs, this time from Microsoft and AskJeeves who automatically filtered their raw logs to select queries containing question words. NIST did additional human filtering of the logs, selecting a final set of 500 questions. Except for some tweaking of the spelling and punctuation, the questions were as they appeared in the log.

NIST has made no attempt to control the relative number of different types of questions in the test set from year to year. Instead, the distribution of question types in the final test set has reflected the distribution in the source of questions. The TREC 2001 test set contained a dramatically greater proportion of definition questions than the previous years. While a large fraction of definition questions is "real" in that the filtered MSNSearch and AskJeeves logs contain many definition questions, there are easier ways to find the definitions of terms than searching for a concise

definition in a corpus of news articles. As a result, NIST intends to exert somewhat more control over the distribution of question types in future tracks.

5. Other Tasks

Each of the TREC QA tracks have differed slightly from one another in ways other than the manner in which the test set of questions was assembled. To investigate whether QA systems are robust to the variety of different ways a question can be phrased, the TREC-9 question set contained 500 questions drawn from the logs, plus an additional 193 questions that were syntactic variants of an original question. For example, the test set contained four variants for the question *What is the tallest mountain?*: *What is the world's highest peak?*, *What is the highest mountain in the world?*, *Name the highest mountain.*, and *What is the name of the tallest mountain in the world?*. Systems that parsed questions into a common representation generally had fewer differences in their responses to question variants than did systems that relied on templates to classify questions by answer types. Overall, however, most variant sets showed little variability in the average score obtained by the different participants, indicating that the difficulty of obtaining the underlying information being sought dominated the results. For the few variant sets that did have a wide range of average scores, the difference was usually caused by different word choices in the variants. For example, the original question *Where was Poe born?* had a much higher average score than any of the variants that all asked for Poe's birthplace.

The TREC 2001 track contained three tasks, the main task, the list task, and the context task. The main task was similar to the previous tracks except questions were not guaranteed to have an answer in the document collection. Recognizing that there is no answer is a challenging task, but it is an important ability for operational systems to possess since returning an incorrect answer is usually worse than not returning an answer at all. The majority of the systems did not attempt to do no-answer processing.

The list task was designed to require systems to assemble an answer from information located in multiple documents. Such questions are harder to answer than the questions used in the main task since information duplicated in the documents must be detected and reported only once. The test set of questions consisted of 25 questions constructed by NIST assessors, each of which specified a target number of instances of a particular kind of information to be retrieved. For example, *What are 9 novels written by John Updike?* was one of the question used in the task. Systems returned an unordered list of [*document-id*, *answer-string*] pairs where each pair represented a single instance. The list could contain no more than the target number of instances. Each individual instance was judged as in the main task. The evaluation metric used was average accuracy, where the accuracy for a single question was the number of distinct correct instances retrieved divided by the target number of instances. The best performing system had an average accuracy of 76%, suggesting that the list task as defined is feasible with current technology.

The context task was intended to test systems' ability to track discourse objects (context) through a short series of questions. However, system performance was so dominated by whether the system could answer the particular type of question posed that differences in ability to track context were not detectable. More research is needed to create an evaluation that actually measures a system's ability to track context.

6. Future Evaluations

The TREC QA track has stimulated research on open-domain question answering and has created a foundation on which future evaluations can build. The data used in the TREC tracks, including questions, answer patterns, sentences containing answers, and evaluation scripts are available on the TREC web site (National Institute of Standards and Technology, 2002).

To date, the TREC QA track has used only factoid questions. This allows the evaluation of the answers to be judged using a binary decision of correct/incorrect. While assessors' opinions as to correctness differ even for this basic question type, evaluation is at least stable in that the relative quality of different QA systems is not materially affected by such differences in opinion. Answers to other types of questions require a more fined-grained scoring procedure: answers that are explanations or summaries or biographies or comparative evaluations cannot be meaningfully rated as simply right or wrong. The appropriate dimensions along which such answers should be judged, scoring mechanisms that reflect quality in those dimensions, and the stability of evaluations using those scoring mechanisms all need to be investigated.

The impact the way in which the test set of questions was assembled has had on system effectiveness in TREC illustrates the balancing of tensions required to create an effective test. One the one hand, careful selection of questions allows specific features of QA systems to be tested, enabling crisper conclusions to be drawn. On the other hand, such selection generally reduces the realism of the test. Designed tests usually lack the diversity of subject matter, vocabulary, and sentence constructions that are represented in large samples of naturally occurring questions. Such diversity can be particularly important to include in initial evaluations when the features that affect performance on the task are not well understood.

The TREC track will continue, with the goal of increasing the kinds and difficulty of the questions that systems can answer. The main task in the TREC 2002 will focus on having systems retrieve the *exact* answer. In past tracks, responses could contain extraneous information and still be judged correct provided the extraneous information was not distracting. Such fuzziness in the definition of correct was used in the first track when it was unclear what the systems' abilities were, and it has remained. However, the fuzziness is masking true differences in systems in the final scores. Forcing system to be precise will not only allow scores to better distinguish among technologies, but also improve QA technology.

An evaluation effort related to the TREC QA track is the new AQUAINT (Advanced QUestion and Answering for INTelligence) program sponsored by ARDA (Ad-

vanced Research and Development Activity), a research center within the U.S. Department of Defense (see `http://www.ic-arda.org/`). The main focus of AQUAINT is to move beyond factoid questions, including the investigation of scoring mechanisms for complex answer types. Within the first year of AQUAINT (2002), AQUAINT contractors and NIST will run pilot studies to experiment with different measures.

7. References

Eric Breck, John Burger, Lisa Ferro, Lynette Hirschman, David House, Marc Light, and Inderjeet Mani. 2000. How to evaluate your question answering system every day ...and still get real work done. In *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC-2000)*, volume 3, pages 1495–1500.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press.

Julian Kupiec. 1993. MURAX: A robust linguistic approach for question answering using an on-line encyclopedia. In Robert Korfage, Edie Rasmussen, and Peter Willett, editors, *Proceedings of the Sixteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 181–190. Special issue of the SIGIR FORUM.

National Institute of Standards and Technology. 2002. The Text REtrieval Conference web site. `http://trec.nist.gov`.

Amit Singhal, Steve Abney, Michiel Bacciani, Michael Collins, Donald Hindle, and Fernando Pereira. 2000. AT&T at TREC-8. In E.M. Voorhees and D.K. Harman, editors, *Proceedings of the Eighth Text REtrieval Conference (TREC-8)*. NIST Special Publication 500-246. Electronic version available at `http://trec.nist.gov/pubs.html`.

Ellen M. Voorhees and Dawn M. Tice. 2000. Building a question answering test collection. In *Proceedings of the Twenty-Third Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 200–207, July.

Why are People Asking these Questions? :
A Call for Bringing *Situation* into Question-Answering System Evaluation

Elizabeth D. Liddy
Center for Natural Language Processing
School of Information Studies
Syracuse University
Syracuse, New York 13210
315-443-5484 (v) 315-443-5806 (f)
liddy@syr.edu; www.cnlp.org

Introduction

I believe that in order for the field of Question-Answering (QA) to evolve to the stage where it will provide maximum utility, the environment in which a QA system is to be used should become a parameter in the evaluation of QA systems. That is, the current evaluation paradigm is becoming restrictive and may well push development in a single direction that will not produce systems that will prove useful in multiple environments. Even a quick review of the potential scenarios in which QA can be utilized suggests two key facts: 1) what is considered *'a useful answer'* in one context might not be useful in another, and; 2) currently permissible methods that systems can utilize to determine correct answers are not feasible in many real world QA environments. This paper will advance this position and suggest a range of situational dimensions that should be considered for inclusion in the QA evaluation roadmap.

QA Evaluation

While there was significant early research in Question Answering in the fields of logic and linguistics (Belnap, 1963; Belnap & Steel, 1976), automatic QA was first focused on in a large-scale evaluation framework in the TREC Conferences, beginning with TREC-8 in 1999 (Voorhees & Tice, 1999). The paradigm established in TREC-8 and continued in the next two TREC Conference QA tracks is simple fact-based, short-answer questions. Initially, answer strings were limited to either 50 or 250 bytes depending on the run type. In TREC-10, the 250 byte condition was eliminated and the list task was added. The list task consisted of 25 questions which specified the number of unique responses to be retrieved, e.g., *"What four countries are the top producers of wheat in the world?"* All other parameters of the main QA task remained the same (Voorhees, 2001).

Discussion at the TREC 2001 Workshop on QA intimated that the QA track in TREC 2002 will accept as correct only fragments which contain the minimal answer to the question. Any explanatory text, even if within the 50 byte limit, will cause the answer to be marked as incorrect. Additionally, the practice introduced in TREC 2001 of a system first determining the most frequent potential answer by searching the web, and then finding a document in the TREC collection which contained that answer fragment will continue to be allowed.

Potential Problems

The need for a more refined evaluation of answer strings was evident from some sample answers shown at the Workshop as they contained text that was non-contributory to the answer and just happened to contain the correct answer that had been provided to the relevance assessors. However, this was not always true. In some instances, the additional text can be argued to have provided useful supportive or confirmatory information. The potential problem I see in the

requirement of a minimal answer is that this evaluation paradigm, which does not permit the inclusion of supporting information that might be useful in some QA scenarios, will foster the development of systems which will be useful in only a subset of the contexts in which QA systems are truly needed.

Furthermore, the decision to allow systems to utilize redundancy on the web to select answers (Brill et al, 2001) will also foster methods that may not be useable in many QA environments. It is highly unlikely that the redundancy approach will transfer to QA systems that are developed for specialized resource environments. While the simple factoid questions for which multiple instances of responses can be found on the web have been the norm in the QA track, this is not typical in other environments for which QA systems provide great utility.

While the existing QA evaluation scenario has utilized very simple questions, has focused on a narrow definition of length of useful answer to the exclusion of other issues, and has permitted the use of a method of determining an answer which will not work in other than the simple query environment, some QA system builders have begun to call for an evaluation paradigm that considers dimensions above and beyond correctness (Breck et al, 2000). We strongly agree with this view and encourage the discussion of a broader evaluation paradigm for the QA Roadmap that will take into account the wide range of environments in which QA is already providing an essential service.

Range of Possible QA Environments

Consider the three following real-life environments for which we have developed QA systems. In each of these environments, the collection, the type of queries, how the system determines answers, and what constitutes an acceptable answer formulation for the user vary dramatically.

1. Scientific Questions from Undergraduate Students

We have developed a QA system (Liddy, 2001) with funding from NASA and AT&T for use within a collaborative learning environment for undergraduate students from two universities majoring in aeronautical engineering who are taking courses that are taught within the AIDE (Advanced Interactive Discovery Environment for Engineering Education). The students are able to ask questions and quickly get answers in the midst of their hands-on collaborations within the AIDE. The collection against which the questions are searched consists of textbooks, technical papers, and websites that have been pre-selected for their relevance and pedagogical value. We are currently working towards the addition of transcripts of class lectures and accompanying power point slides. The students questions are not typically simple factoid questions, but tend more towards '*Why*' and '*How*' questions and require more than bare answers, such as:

? *How do ablating materials minimize energy conducted into a RLV?*
? *What are the changes made to the design of the Shuttle SRM since the Challenger Accident?*
? *How are malfunctions detected for the pitch and yaw gimbal actuators of the space shuttle OMS engines?*

Answers are provided in increasing window sizes, allowing the student to gradually expand the amount of text by mouse-clicking from 'answer-providing passage', to paragraph (s) containing the 'answer-providing passage' to full document(s) containing the 'answer-providing passage'. The system is currently undergoing user testing. The U S Army has funded us to create a similar capability for the students in the Army's intel training programs. They share NASA's vision that

work in the future will consist largely of virtual collaborative situations in which questions that arise will need to be answered electronically from selected sources.

2. Citizens' Search for Statistical Information

Naïve users need to access statistical information, but frequently do not have the sophisticated understanding required in order to translate their information needs into structured database queries using the controlled vocabulary which are currently required. However, these users can articulate quite straightforwardly in their own terms what they are looking for. One approach to satisfying the masses of citizens with needs for statistical information is to automatically map their natural language expressions of their information needs into the metadata structure and terminology that defines and describes the content of statistical tables. To accomplish this goal, under funding from NSF's Digital Government Initiative (http://istweb.syr.edu/~tables/), we undertook an analysis of 1,000 user email queries seeking statistical information from federal agencies which provide internet access to their statistical tables. Our goal was to understand the dimensions of interest in naïve users' typical statistical queries, as well as the linguistic regularities that could be captured in a statistical-query sublanguage grammar. We developed an ontology of query dimensions using this data-up analysis of the queries and extended the ontology where necessary with values from actual tables. We proceeded to develop an NLP statistical-query sublanguage grammar that enabled the system to semantically parse users' queries and produce a template-based internal query representation which was then mapped to the tables' metadata, in order to retrieve relevant tables which were displayed to users with the relevant cell's value highlighted (Liddy & Liddy, 2001). Typical queries were:

? *I am trying to find the percentage of women in the workforce from the years 1900 to 1998*.
? *I want to know how many people worked for small businesses last year.*
? *What was the average amount of time women spent on housework per week in 1900; 1950; 1995?*

This project made it eminently clear that the situation predicts the nature of the questions, the resources searched, and the acceptable answer formulation.

3. Speech-based Inquiries in Travel and Tourism

In an exciting project in the commercial world, we worked with a speech understanding technology company to provide answers to travelers who were planning Caribbean vacations via interaction with a voice-activated system. While the business idea was well-researched, the current status of speech-understanding technology was not, and the corporation failed to pull off the application. However, I mention it here because it introduces a third and very different set of users, answer-providing resources, and answer formulation in which appropriate supporting detail is essential.

? *We're looking for a family resort in the Caribbean with baby sitting, other activities for a family with a one and three year old. Any suggestions?*
? *My fiancee and I were wondering if there was anywhere we could go in October that would not be extremely crowded, yet more secluded?*
? *When is the best time to go on a Caribbean Cruise - and do you recommend bring our 16 year-old so? He is very bright.*

Again this situation points out that evaluation needs to reflect an environment – we do not foresee that all questions will be ones that can be satisfied with short answers which are found redundantly present on the web. Requirements in this particular situation contradict the TREC QA evaluation requirement that evidence supporting the answer should not be provided.

Conclusion

We have found that the collection of documents that will be available for querying, the nature of queries generated by real users, as well as the breadth vs. narrowness of what constitutes a useful answer in each of these instances is not the same. Therefore, it would only seem appropriate that an evaluation should fully specify the user, the purpose for which they are asking their question, and the nature of an acceptable answer. These should be parameters that can be varied in QA evaluations. It is essential that the situational aspects be known so that the criteria provided to the human relevance assessors truly reflect what users in that particular context would require. Evaluations should be designed that simulate as closely as possible the dimensions of the context in which users will be posing their questions. Clearly the use of multiple scenarios would enhance the possibility that evaluation would lead to a range of QA systems, each defined by the parameters of the situation in which they are to be used.

References

Belnap, N. D. (1963). An analysis of questions: Preliminary report. Scientific Report TM-1287. Santa Monica, CA.

Belnap, N. D. & Steel, T. B. (1976). The logic of questions and answers. New Haven, CT., Yale University Press.

Brill, E., Lin, J., Banko, M., Dumais, S. & A. Ng. (2001). Data-Intensive question answering. Notebook Proceedings of the Text Retrieval Conference. Gaithersburg, MD: NIST Special Publications.

Breck, E.J., Burger, J.D., Ferro, L, Hirschman, L., House, D., Light, M. and Mani, I. (2000). How to evaluate your question answering system every day…and still get real work done. Proceedings of Language Resources and Evaluation (LREC).

Liddy, E.D. (2001). Breaking the Metadata Generation Bottleneck. Joint Conference on Digital Libraries. Roanoke, VA., June 25, 2001.

Liddy, E.D. & Liddy, J.H. (2001). An NLP approach for improving access to statistical information for the masses. Proceedings of the Federal Committee on Statistical Methodology Research Conference. Arlington, VA.

Voorhees, E. and Tice, D. (1999). The TREC-8 question answering track evaluation. In Voorhees, E. and Harman, D. Proceedings of the Eighth Text Retrieval Conference. Gaithersburg, MD: NIST Special Publications.

Voorhees, E. (2001). Overview of the TREC 2001 question-answering track. In Voorhees, E. and Harman, D. Notebook Proceedings of the Text Retrieval Conference. Gaithersburg, MD: NIST Special Publications.

A Curriculum-Based Approach to a QA Roadmap

John Prager
IBM T.J. Watson Research Center
Yorktown Heights, N.Y. 10598
Tel (914) 784-6809; Fax (914) 784-6078
jprager@us.ibm.com

Abstract

The QA community is beginning to understand the core problems in the field, and they largely coincide with those of Natural Language Understanding. The difficulty of answering a question by a current QA system is a function of the match or lack of it between the question or its expression and the resources used to answer it, not how difficult it is for a human to answer it. A prominent factor in making a question hard now is not so much in finding an answer but in validating whether a candidate answer is correct. The problem in many ways parallels that of reading comprehension for children, which suggests a graduated approach to developing and evaluating the field. The difficulties faced by QA systems include long-standing issues in computational linguistics, such as anaphora resolution, metonymy etc.; logic-oriented issues such as scope and quantification as introduced by adverbs and articles; structural problems where the answer must be assembled from many sources, as well as reasoning about space, time and numbers. These problem areas are largely orthogonal, and can be introduced progressively with at each step accepted criteria for success.

Introduction

The approach TREC has been taking to Question Answering has been rather like asking fourth-graders to read and understand *Hamlet*, and when they show even some rudimentary success, moving them on to *War and Peace* and then *Finnegan's Wake*. While it is very understandable that members of the community - or indeed several communities: academic, government, military and web-users - wish to push the state of the art as far and as fast as possible, it is inescapable that complete success at QA requires mastering all of the core problems of NLP. This has not been done over the last fifty years and is not going to be achieved anytime soon.

Approximately two years ago, a first QA Roadmap was drafted on behalf of ARDA (ARDA, 2000), based on input from many key researchers in the field (including the present author). That document developed the question taxonomy previously proposed by researchers at SMU (Moldovan et al., 2000). That taxonomy lists a series of increasingly difficult questions, characterizing them by the kind of questioner who would ask them. The taxonomy is very well intentioned but, in hindsight, unfortunately wrong in some of its details or its emphasis and difficult to work with because of two inherent assumptions that appear to have been made, or at least not rejected.

The problematic assumptions are (1) that it is possible to grade the difficulty of questions by semantics independent of the corpus and/or other resources that will be used to answer them, and (2) that what is difficult for a human will also be difficult for a computer. As for the first point, we observe that understanding the question is indeed part of the QA process, but it is only a part. Understanding the corpus (plus ontologies and other kinds of data) is equally important, as is being able to match these resources to the question. Sometimes such a match is trivial, sometimes it requires considerable linguistic processing and/or reasoning: which is the case cannot be determined from the question alone.

For example, consider the question:

> When was Queen Victoria born?.

It is very easy to answer if there is a text passage of the form:

> … Queen Victoria was born in 1819…,

and only a little trickier if the text reads

> … Queen Victoria (1819-1901) ….

However, if the text contains no such statements, but instead just the indirect reference

> … King George III's only granddaughter to survive infancy was born in 1819 …,

along with text (possibly elsewhere) that states

> … Victoria was the only daughter of Edward, Duke of Kent,

along with more text (possibly yet elsewhere) that states

> … George III's fourth son Edward became Duke of Kent …

the question becomes considerably harder to answer.

By contrast, the seemingly difficult question

> Should the Fed raise interest rates?

becomes much simpler to answer in the presence of a news article quoting Alan Greenspan as saying

> All of the current leading economic indicators point in the direction of the Federal Reserve Bank raising interest rates at next week's meeting.

On a lighter level, even the perennial

> What is the meaning of life?

is a cinch to answer if one consults *The Hitchhiker's Guide to the Galaxy* (Adams, 1982)[1].

If one accepts that questions by themselves cannot be arranged in order of difficulty, the very notion of a Roadmap might seem to be called into question. However, it is the thesis of this paper that a systematic approach mirroring somewhat an academic curriculum can achieve the desired goals. A basic method of the classical Western educational system is the incremental dissemination of new information and skills, building on previous knowledge (as opposed to, say, the immersion approach to language learning). Evaluation is performed continually, with testing materials crafted either to examine as closely as possible just the new material, or a combination of new and old, as the teacher sees fit.

Going beyond TREC

The problem with the current TREC-style evaluation using real user's questions and real news articles is that every question can potentially test a different variety and combination of system skills and knowledge, so a system's performance can vary widely from question set to question set. A given system can fare remarkably differently on seemingly isomorphic questions because of idiosyncrasies of the data resources. Granted, using large enough question sets it becomes possible to rank order QA systems, as TREC does (Voorhees & Tice, 2000), but the current setup does not enable one to easily assert exactly what is being tested in a system (except QA in a holistic way), or what, if anything, a system is good at. Amongst other things, this makes it difficult to predict performance when a QA system is to be deployed in a new domain, or how it will behave with different user groups.

Three trends in TREC QA, from the first instance in TREC8 to the proposed TREC2002, have had and are having the benefit of forcing systems to "know" what they are doing. These are: (1) the trend from 250-byte answers to 50-bytes to "exact answer", (2) from 5 submitted answers to a single answer, and (3) the (as yet largely unexploited) possibility of "no answer". These refinements of the track are fine and do a great service in that they greatly reduce the chances that systems get the right answer "by accident", but they represent the end of the line in this particular kind of evaluation development. It should be mentioned, though, that while these improvements are necessary for the evolution of QA systems whose output will in turn be used by other automatic systems, they are not so necessary when the consumers of the output are real users, who can tolerate a set of candidate answers and who will generally be pleased to see the answers in the context of text passages. Having said that, it is true that if a system can do well in the more constrained context it can only benefit its performance in the less constrained one.

The essential difficulty with question answering stems from the fact that textual material is in natural language, and that to consistently answer questions posed against text corpora requires understanding the text. Since these texts were written with human readers in mind, they make copious use of all of the linguistic and stylistic devices that make reading pleasurable and computer understanding difficult: anaphora, definite noun phrases, synonyms, subsumption, metonyms, paraphrases, nonce words, not to mention idioms, figures of speech and poetic or other stylistic variations. For example, in answer to "How did Socrates die", we find from the TREC corpus:

> His chapter on wifely nagging traces nagging back to the late Cretaceous period and notes that one of the all-time nags was Socrates' spouse, Xanthippe. Hemlock was a pleasure by comparison.

and

> We also meet snake root, which is toxic, and poison hemlock, which for over two thousand years has been famous for curing Socrates of life.

In fact, all of the other mentions of Socrates and hemlock together in this corpus happen to be indirect, thus making this simple sounding question particularly difficult. Usually, though, in a large corpus such as TREC uses there are multiple mentions of facts interesting enough to be the subject of questions, and for every obscure reference there are often several plain ones.

Following the train of the argument in this paper, it would seem that by far the easiest way to provide a Roadmap for QA would be to mimic the progression of reading comprehension tests in school, by using texts written for progressively higher grade-levels. These would start with texts employing only short sentences using simple syntax and little imagery, and progress to adult-level texts such as news articles and beyond. The difficulty here, though, is that these elementary texts do not exist in sufficient quantity, especially online, to provide a meaningful-sized corpus (the current TREC QA corpus is 3GB). If we cannot fix the corpora, then at least we can fix the questions. [We should mention here a recent posting by Karen Spark-Jones to the TREC web site (Spark-Jones, 2001). The posting lists a set of questions, and for each one a

[1] The answer is 42.

large number of candidate answer sentences that address some aspect of the questioner's concern, but may or may not answer the question itself. This is in the same spirit as the theme of this paper, as it finesses the issue of finding such sentences, but allows one to concentrate on the problems of question-answer match.]

Impedance match

Using as background the earlier argument that multiple mentions of interesting facts should generally reduce problems of text complexity, we can again advance the suggestion that sets of increasingly difficult questions be developed. The measure of difficulty, though, will be quite different from that espoused in the first QA Roadmap. The notion is to identify components of the QA task that are difficult for a machine to perform, rather than difficult for a human. In some cases, the difficulty will ensue from the absence of a direct answer in the resources used, as discussed above. In other cases, the difficulty will derive from the linguistic and/or logical structure of the question, rather than its semantics (that is, the semantics of the individual content words). Take for example the question "What is the population of France's capital?". Assuming that there is no text that directly restates the question, the task is to first find the capital of France (Paris), and then to find the population of Paris; these two steps may well be performed using different documents or different knowledge bases or databases. The level of difficulty of the question does not stem from the fact that two resources must be searched. Given the problem breakdown, it is straightforward to construct the two necessary queries. The difficulty comes from the question's structure: the system must know that the phrase "France's capital" is a reference to an entity that must itself be found before the outer question can be answered.

The structure of the problem in general ensues from not only the structure of the question but also the availability of *knowledge sources*: both the information resources and the kinds of processing needed to make use of them. The question "What is the largest city in France?" can be answered in a variety of ways: from a direct statement in text; from a table listing French cities and their sizes; from discovering that Lyon is the second largest French city, and that Paris is larger than Lyon; from an enumeration of separately discovered pairs of {city, size} (making assumptions of completeness), and others. The difficulty of the task can be varied by making available or unavailable any of the pertinent knowledge sources. To summarize, the measure of difficulty of the questions mentioned so far in this paper stems from what might be called the *impedance match* (or *mismatch*) between question and knowledge sources. Moving on, we can orthogonally mine the linguistic dimension for incremental difficulty.

The Linguistic Dimension

In what follows we present an unordered and non-exhaustive list of the kinds of linguistic capabilities that a full-fledged QA system should have. These capabilities can be expressed and evaluated by question sets that require that particular competence for successful performance. We have already seen some examples of questions that derive their difficulty from the absence of a direct and straightforward representation of the answer in the available resources. The remaining examples are for the most part easy for humans to address, but illustrate difficulties that computers have with NLP.

Consider the following two questions:

> Name a US state where automobiles are manufactured

and

> Name a US state where automobiles are not manufactured

The vast majority of present-day QA systems will pay no attention to the *not*, although it is critical for correct behaviour. Likewise, other adverbial modifiers such as *just* and *only* can play havoc with the system's performance. Sometimes the presence of a single such modifier can require large amounts of real-world knowledge. Consider

> Name an astronaut who made it to the moon

versus

> Name an astronaut who nearly made it to the moon

One can easily come up with half-a-dozen reasonable interpretations of *nearly* here, each giving different sets of correct answers.

In a similar vein, articles play an important role in question interpretation. The TREC community has been arguing for years whether Atlantic City is a correct answer to "Where is the Taj Mahal?". Making the article indefinite would generate much less of a dispute whether casinos, hotels and restaurants were allowable answers; having a computer understand the difference, though, would be a challenge. Interesting questions arise when articles are absent and the end-user is unknown. Is the question:

> What is mold?

really a hurried form of

> What is a mold?

What if the end user is the native speaker of a language that doesn't use articles? One can imagine an exercise where the system is given a set of questions to be answered in the context of each of a set of user-profiles. These profiles may be no more than simple age/profession/nationality descriptors, but sufficient to elicit different maximum-likelihood interpretations for each question in the set.

An important area where difficulty can be introduced in an orthogonal manner is in that of ungrammatical ques-

tions. Although NIST has tried to make the TREC QA questions immune from this problem, by the author's count about two percent contain one or more misspellings, incorrect capitalizations, incorrect compoundings, or syntax errors. Observing the first such errors in TREC8 has had the unintended beneficial consequence of causing some groups to develop and deploy spell-checkers and other fault-tolerant mechanisms. Raw questions from real users undoubtedly contain a much higher percentage of such errors than in TREC; keyword-based queries, so common on the Web, can be considered to be degenerate cases of ungrammatical sentences.

A common cause of problems, not only in QA but also in basic Information Retrieval, is the lack of lexical match between two equivalent or ontologically-related concepts. Question sets that specifically test subsumption, synonymy, meronymy and other relationships can easily be generated, in the obvious way.

QA systems today don't do well with numbers. "How many"-type questions are easy to answer if the sought figure is discussed in text, but not so if the system has to enumerate instances. Ability to convert between units is largely absent. Ability to evaluate reasonable magnitudes is also missing.

QA systems are currently monolingual. It is clearly desirable to be able to query in one language texts in another, but there is scope for awareness of other languages that falls far short of full CLIR, or maybe that should be CLQA. Even simple questions like "What does ciao mean?", bearing no explicit indication of foreign language presence, can benefit greatly from systems having some notion of what is English.

Summary

Developing a Roadmap for QA entails developing a series of tasks which, when mastered, would result in an extremely capable system. The current TREC approach of requiring QA systems to do everything in the first year, and just be better at it in subsequent years, does not provide the right kind of incremental basis. Instead, rather like in a modular school curriculum, technical areas to be addressed should be identified and codified in question-sets that require the requisite capability to answer. The question-sets may be accompanied by restrictions on resources that may be used. Such "learning modules" can be either orthogonal or incremental, or even some combination. Developing them will not be as easy as generating the TREC question-sets, since, in many cases, knowledge by the question-set compiler of the resources available (text corpora, ontologies, databases) will be necessary to judge how and where a given question is appropriate, just as a textbook author must know the subject matter in order to set appropriate questions for each chapter.

Acknowledgments

The author would like to thank Jennifer Chu-Carroll and David Ferrucci for their helpful comments and suggestions. This work was supported by the Advanced Research and Development Activity (ARDA)'s Advanced Question Answering for Intelligence (AQUAINT) Program under contract number MDA904-01-C-0988.

References

Adams, D. (1982). The Restaurant at the End of the Universe, book 2 of the Hitchhiker's Guide to the Galaxy trilogy, Pocket Books, NY.

ARDA (2000). Issues, Tasks and Program Structuresto Roadmap Research in Question & Answering (Q&A) (http://www-nlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc)

Moldovan, D., Harabagiu, S., Pasca, M., Mihalcea, R. Goodrum, R. Girju, R and Rus, V. (2000). "LASSO: A Tool for Surfing the Answer Net", Proceedings Eighth Text Retrieval Conference, E.M. Voorhees and D.K. Harman Eds., NIST, Gaithersburg, MD.

Spark-Jones, K. (2001) "Question-Answering Data" http://trec.nist.gov/data/qa_no_pword/qa_task.txt.

TREC8 (2000) "The 8th Text Retrieval Conference", E.M. Voorhees and D.K. Harman Eds., NIST, Gaithersburg, MD.

TREC9 (2001) "The 8th Text Retrieval Conference", E.M. Voorhees and D.K. Harman Eds., NIST, Gaithersburg, MD.

TREC2001 (2002) "The 8th Text Retrieval Conference", E.M. Voorhees and D.K. Harman Eds., NIST, Gaithersburg, MD (to appear).

Voorhees, E.M. and Tice, D.M. (2000). "Building a Question Answering Test Collection", *Proceedings of SIGIR 2000*, pp. 184-191, Athens, Greece.

Evaluating QA Systems on Multiple Dimensions

Eric Nyberg & Teruko Mitamura

Language Technologies Institute
Carnegie Mellon University
5000 Forbes Ave., Pittsburgh PA 15213 USA
{ehn,teruko}@cs.cmu.edu

Abstract

Question-answering systems are expanding beyond information retrieval and information extraction, to become full-fledged, complex NLP applications. In this paper we discuss the evaluation of question-answering systems as complex NLP systems, and suggest three different dimensions for evaluation: objective or information-based evaluation; subjective evaluation; and architectural evaluation. We also discuss the role of ambiguity resolution in QA systems, and how ambiguity resolution might be evaluated.

1. Introduction

The recent QA Roadmap (Burger et al., 2001) expanded the scope of question answering along several dimensions, including: multiple question types, multiple answer types, multiple media, multiple languages; interactive dialog with the user to refine/guide the QA process; multiple answer perspectives; and ultimately, answers which provide an evaluation or judgment based on retrieved data. QA systems are expanding beyond information retrieval and information extraction, to become full-fledged, complex NLP applications.

We present three different types of evaluation: a) *information-based evaluation*, which (like the TREC QA track) focuses on the completeness and correctness of the answers given; b) *utility-based evaluation*, which focuses on the usability of the QA system for the end-user; and c) *architectural evaluation*, which focuses on the characteristics of the software architecture used to implement the QA system. For each type of evaluation, we discuss possible ways to define test data and carry out an evaluation.

These three types of evaluation are relevant for next-generation QA systems such as JAVELIN (Nyberg et al., 2001). The ideas presented here draw upon our experience with the evaluation of other complex NLP systems (e.g. Machine Translation (Mitamura et al., 1999), Integrated Information Management (Nyberg & Daume, 2001)) that are directly relevant to advanced QA.

2. Extending Information-based Evaluation: Ambiguity Resolution

At the core of current QA evaluation methods is the objective evaluation used in the TREC QA track. Objective evaluation requires the creation of questions and correct answers for each question, given a corpus and some pre-defined criteria for judging "correctness". The QA Roadmap describes the evolving capabilities of QA systems, which will require new objective measures (i.e. new TREC QA tasks). Although objective evaluation is extremely useful and easy to carry out once the data sets have been created, it is probably not feasible to create a single suite of questions that adequately tests all dimensions of a QA system in an objective manner. For example, different suites might evaluate system's performance on various question types, answer types, document sets, etc. More global capabilities, such as ambiguity resolution, cut across all of the question and answer types and should be evaluated separately. In the remainder of this section we discuss the specific challenges of creating an objective evaluation for ambiguity resolution.

Starting with TREC 2002, the QA evaluation track will include question ambiguities. In general terms, an ambiguous question is one that has more than one meaning or interpretation. In a QA system, question ambiguity is significant when the different meanings imply different answers. If there is a high degree of ambiguity (many different meanings), or the ambiguity implies a much greater degree of information processing (many more texts to be searched), the system should attempt to resolve the ambiguity.

Ambiguity in natural language has been studied in detail in the fields of computational linguistics and machine translation, and all of the classic forms of ambiguity can affect a QA system (lexical ambiguity, syntactic ambiguity, pronominal anaphora, scope ambiguity, etc.). When designing a QA system, it is important to consider a) whether (and how) to detect a particular type of ambiguity; b) whether (and

how) to resolve the ambiguity before searching for an answer; c) whether (and how) to resolve ambiguity as part of composing the answer. The diagram in Figure 1 illustrates the difference between approaches b) and c). In either case, the system can resolve the ambiguity automatically, or interact with the user to resolve the ambiguity.

In the following subsections, we discuss three specific types of automatic disambiguation that can be evaluated in an advanced QA system: context disambiguation, structural attachment disambiguation, and word sense disambiguation.

2.1. Context Disambiguation

Context can be disambiguated automatically by using the analyst profile or past session memory. The context category (e.g. economy, politics, geography, etc.) can be used for disambiguation. Questions might include words that can belong to different domains; for example, the words "line, defense, conference", may indicate an academic context or a sports context.

Questions that refer to attributes of objects may also be ambiguous in different contexts. As noted in the QA Roadmap (Burger et al., 2001), the same attribute name might imply different answer types. For example, the general notion of "dimension" as queried in questions like *How big is New York*? or *How big is the Pacific Ocean?* implies different possible answer types (e.g. a population count, a geographical area in square miles, etc.). In each case, the QA system must select more specific query terms that are appropriate to the particular meaning intended. For example, *How old is Koizumi?* can be answered by searching for a birth date, where a question like *How old is Siemens?* requires searching for events like *incorporated*, *founded*, etc. The strategy depends on knowing whether the question refers to a person or an organization, in this case.

It is a large task to address this type of ambiguity for unrestricted English text, since this presupposes a well-defined semantic model with broad coverage ("world knowledge"). A more feasible method for developing test data and evaluations might be to construct an model of the most relevant contextual ambiguities for intelligence gathering tasks. For the most relevant query object and answer types associated with a particular corpus (e.g. person, organization, location, country), it should be possible to determine the set of salient attributes of each type (e.g. age, location, size), along with the potentially ambiguous question terms that are typically used to refer to those attributes. This type of empirical data gathering presupposes that a set of sample questions are available for analysis. Once all of the attributes and their source language query terms have been identified, a set of questions could be constructed to evaluate a system's ability to search for the correct attribute given an ambiguous query.

Although this discussion has focused on single attributes, realistic questions will also include nested attributes, such as *How big is support for Koizumi?* For this question, it is important to know that a) Koizumi is a person in the political arena, b) *support* in this context implies public opinion concerning job performance, and c) big is a relative measure of public opinion, perhaps based on the results of a public opinion poll. Handling this type of nested ambiguity will require not only the disambiguation of nouns such as *Koizumi* and *support*, but also an understanding of syntactic structure and the relationships represented by prepositions like *for*.

2.2. Structural Disambiguation

One of the challenges for phrase level analysis is the resolution of structural attachment ambiguity (e.g. prepositional phrase attachment). In building the JAVELIN system, we plan to extend the automatic structural attachment heuristics developed for the KANT system (Mitamura et al., 1999) to handle structural disambiguation in question analysis. If the system cannot automatically resolve structural ambiguity, then it will ask the analyst for clarification.

In our work on machine translation, we have developed two fundamental ways to evaluate ambiguity resolution: a) by testing analysis results (meaning interpretations) against a pre-defined "gold standard", and b) by checking the correctness of the translation results (Mitamura et al., 2002). Interestingly, an incorrect ambiguity resolution sometimes has no impact on the quality of the translation result, because the input sentence can be translated correctly in spite of the mistake. The analogy for QA systems is that there will be ambiguous questions that can be answered correctly using simple methods without ambiguity resolution, e.g. simple query term search without reformulation. An adequate test suite for ambiguity is one where the probability of getting the correct answer is significantly increased if some form of ambiguity resolution takes place.

Another type of structural ambiguity is seen in the phrase *domination of China*, which could be interpreted as *someone is dominated by China*, or as *China is dominated by X* If we think of *dominate* as a binary predicate accepting

two organizations or countries as arguments, then the nominal form *domination of X* will be a common way to ask questions about *dominate* events when one party is unknown. The ambiguity arises when the pattern $V_{nominal}$ *of N* can be interpreted such that N is either the subject or the object of *V*.

For both types of ambiguity, designing test suites depends on analyzing a set of representative questions to determine what kinds of structural ambiguity arise in realistic scenarios. Since solving the general problem of ambiguity resolution in English is a large, difficult problem, QA evaluations should narrow their focus initially to the types of structural ambiguity that are relevant for QA systems. Once a set of ambiguous constructions is identified (e.g. the *of* case illustrated above), a variety of test cases should be constructed with respect to the evaluation corpus. Effective test cases will be those where more than one potential answer exists, depending on the interpretation of the question, and getting the right answer involves some form of disambiguation.

We also note that there are structural ambiguities that should always be resolved automatically, because only one structural interpretation is semantically valid.

2.3. Word Sense Disambiguation:

During question analysis, word sense disambiguation may follow from identification of the question context (as mentioned above). When there is more than one word sense for a particular term that is not resolved automatically, the system will ask the analyst to choose a term definition from a given list. Evaluating word sense disambiguation can be broken down into two parts: a) does the system represent all of the possible meanings for ambiguous terms in the corpus, and b) can the system correctly select the appropriate meaning in a given sentence (in the absence of contextual or structural cues). For nouns, this involves assigning all possible object types (person, organization, location); for verbs, it involves assigning all possible event meanings.

Once a set of common ambiguous words are identified, based on an analysis of realistic scenarios, a variety of test cases should be constructed with respect to the evaluation corpus. Effective test cases will be those where more than one potential answer exists, depending on word sense disambiguation, and getting the right answer involves correct choice of word meaning. There may also be cases where only a single answer exists, and all but one sense of a particular word are invalid in the domain context.

2.4. Discussion

For objective evaluation, the question is "*How well does System X resolve ambiguity type Y*?"[1]. Ambiguity resolution is important if resolving the ambiguity significantly enhances the system's probability of getting the right answer. Conversely, when constructing a test suite, it is useful to select questions where the probability of getting the right answer is significantly lower if the system does not resolve the ambiguity. For each of the ambiguity phenomena, an effective test suite will contain questions that have multiple answers. The TREC answer format (regular expressions) can be utilized. The real challenge is in crafting questions that differentiate between systems that disambiguate and those that do not, since the probability of getting the right answer is also influenced by the specific documents in the corpora and the degree of evidence for alternative answers.

Contextual ambiguity has important considerations for question answering systems. When a single, isolated question is asked, the context is unconstrained and the question can be assigned any meaning that is valid in the scope of the entire corpus. When a question is asked in the context of a question answering dialog, the context may be constrained to the particular topic of that session. Note that a continuation question may include ambiguous references (e.g. pronominal anaphora) that refer to concepts originally introduced in either a prior question or an answer. The QA system should automatically resolve ambiguities by referring to the existing context whenever possible.

For information-based evaluation, it is essential to construct test questions and answers that address the purpose of the evaluation. This is true not only for ambiguity resolution, but also the other QA phenomena that can be evaluated objectively (e.g., answer justification, answer completeness, multilingual QA, etc.).

---

[1] Note that objective evaluation does not consider the processing time used by the system. A system that resolves ambiguity during question analysis might in general be faster than a system that resolves ambiguity during answer generation, since it prunes the search space earlier.

3. Utility-Based Evaluation – How Good is the Tool?

As QA systems move beyond the laboratory to real-world applications, objective information-based evaluations must be supplemented by utility-based evaluations that evaluate the effectiveness of the software for real tasks. End-to-end system evaluations must focus on realistic analyst scenarios, and characterize the overall system's performance under different operating conditions. We envision at least three ways to evaluate end-to-end performance, described in the following subsections.

3.1. Percentage of Task Completion

The most important functional metric is whether or not the system can retrieve the desired information. Of course, a comprehensive test suite for task completion should exercise all of the question types and answer types to be covered by the system. But it is also necessary to consider other dimensions, such as the specificity or "vagueness" of the user's question.

If a question is precise and unambiguous (e.g., When was Enron incorporated?), then the system should retrieve the desired information quickly, with no further interaction with the user. On the other hand, if the question is vague (e.g., Where is Enron?), the evaluation could focus on at least two different outcomes: a) the system finds all possible answers (place of business, global markets, etc.), or b) the system refines the question interactively to focus on the "correct" answer (e.g., *Where is Enron's headquarters located?*).

Once a set of reference questions and answers should is created to exercise all of the possible question types and answer types, the test set should be expanded to include various "vague" reformulations of each question, to test task completion under varying levels of initial specificity.

3.2. Efficiency of Task Completion

This efficiency metric will measure how easy it is to get the desired information using the system. This dimension is crucial for a realistic evaluation; since JAVELIN will support interactive planning with the user, it will be necessary to strike a balance between accuracy (task completion) and automaticity (how much burden is placed on the analyst during the resolution of ambiguity, clarification, etc.). We can measure the overall time elapsed (how long the analyst has to wait for the answer), the amount of time spent by the analyst in responding to clarifications, and the total number of clarifications per question.

When evaluating the efficiency of machine translation systems, we often compare the time required for a complete manual translation to the time required for a machine translation plus human post-editing. To make an analogous comparison in QA evaluation, we should compare the time required by an unaided human (using only a search engine) to retrieve an answer with the time required by a human plus QA system. If a given task takes less time when using the QA system (despite the need for user interaction, refinement, etc.), then the QA system is more efficient than a human using a search engine.

3.3. N-Point Subjective Measure

Researchers in human factors have noted that the fastest system is not always the "best" - users may prefer a system that is up to slower than another, if it provides better feedback regarding its progress. In open-domain QA, it will be important to measure the user's perception of various subjective measures, e.g., *How well do you understand what the system is doing?*; *Does the system provide you with adequate feedback?*; *Is the system easy to use?*; *Does the system ask you too many questions?*, etc. Such measures are important in that they help to determine what the user considers a "usable" system - note that a system which performs no clarifications may not inspire confidence in an analyst who expects to spend a certain amount of time guiding the search.

In our work with machine translation systems, we have observed two important phenomena with respect to subjective evaluation: a) there is a definite threshold regarding interactivity – if the system asks too many questions on a particular task, the user will lose patience and select the default response, especially when under time pressure; and b) if the content of or motivation for a clarification question is not apparent to the user, they will lose confidence in the system. The subjective evaluation of QA systems should attempt to determine whether these two phenomena are also relevant for information-seeking tasks.

4. Architectural Evaluation

An objective "black-box" evaluation focuses on only those characteristics that are important to the end user, who cannot "see inside" the actual system as it is working. But it is also important to consider glass-box evaluation, which has two important benefits: a) the ability to evaluate the performance of individual system modules can

help developers to rapidly locate and address problems in functionality, performance, etc.; b) an understanding of how easy it is to tune, extend and maintain the system. Therefore architectural evaluation is primarily for the system developer and the system client, who are concerned with the global characteristics of the QA system as a product of software engineering.

Architectural evaluation can be performed in the context of a design review (Pressman, 2000), which focuses on the architectural design and system documentation rather than an information-based evaluation. Although QA systems are designed and implemented using a variety of paradigms and techniques, a global set of design criteria that can be evaluated in a more or less subjective manner for each QA system. The requirements for an ideal QA architecture are similar to those summarized by the TIPSTER II architecture working group (Grishman, 1996):

? **Standardization**. Does the system specify a standard set of functions and interfaces for information services? Is it possible to mix and match different modules in a straightforward manner? In the IIM system (Nyberg & Daume, 2001) we specified a set of standard interfaces for system components that allow the end-user to perform unlimited customization without recompilation of the main system.

? **Rapid Deployment**. How easy is it to create new applications from existing components? A system with an inherently modular design is easier to reconfigure for new applications.

? **Maintainability**. Is it possible to update one module in the system without affecting the others? One key for rapid progress in QA research is the ability to work on the different aspects of the problem (question analysis, retrieval, answer formulation, etc.) in parallel, with frequent system-level testing.

? **Flexibility**. How easy is it to alter the performance of the system by allowing novel combinations of existing components?

? **Evaluation**. Is it possible to isolate and test specific modules (or versions of modules) side-by-side in the same application? If a system incorporates multiple strategies or "loops" (Harabagiu, et al., 2000), how can we evaluate the contributions made by each strategy or algorithm to the overall utility of the system?

Complex QA systems incorporate several different algorithms, modules, processing loops, etc. Effective glass-box evaluation requires a certain degree of instrumentation inside the software, so that various measurements, logging, etc. may be done before, during and after key processing steps (Nyberg & Daume, 2001). This allows the developers to identify component-specific effects and perform ablation studies that clearly evaluate the contribution of a particular component to the system's overall performance.

If a QA research effort is focused purely on initial discovery of new algorithms, then perhaps architectural evaluation is of secondary importance. However, for longer-term efforts aimed at building a reusable technology base for ongoing development, we argue that architectural evaluation and attention to software engineering are of paramount importance. The JAVELIN project is intended to produce a general, extensible architecture, and we intend to evaluate the JAVELIN system design along dimensions such as reusability (of components, operators, etc.) and external extensibility (e.g., by ARDA's chosen third-party integrator).

5. Conclusion

Ongoing research is expanding the scope of question-answering systems beyond information retrieval and information extraction to include complex NLP techniques. In this paper, we advanced the idea that the evaluation of advanced QA systems can and should be carried out on three different levels: information-based (objective) evaluation, utility-based (subjective) evaluation, and architectural evaluation. As the field moves beyond its focus on information-based (TREC-style) evaluation, we must develop new test suites and test methods to improve the quality of QA systems along all three dimensions.

6. References

Burger, J., C. Cardie, V. Chaudhri, R. Gaizauskas, S. Harabagiu, D. Israel, C. Jacquemin, C. Y. Lin, S. Maiorano, G. Miller, D. Moldovan, B. Ogden, J. Prager, E. Riloff, A. Singhal, R. Shrihari, T. Strzalkowski, E. Voorhees, R. Weishedel (2001). Issues, Tasks and Program Structures to Roadmap Research in Question & Answering (Q&A), http://www-nlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc

Harabagiu, S. D. Moldovan, M. Pasca, R. Mihalcea, M. Surdeanu, R. Bunescu, R. Girju, V. Rus, and P. Morarescu (2000). FALCON: Boosting knowledge for answer engines. In

9th Text REtrieval Conference, Gaithersburg, MD.

Mitamura, T., E. Nyberg, E. Torrejon, and R. Igo (1999). Multiple strategies for automatic disambiguation in technical translation. In *Proceedings of 8th International Conference on Theoretical and Methodological Issues in Machine Translation*.

Mitamura, T., E. Nyberg, E. Torrejon, D. Svoboda, A. Brunner and K. Baker (2002). Pronominal anaphora resolution in the KANTOO multilingual MT system, *Proceedings of 9th International Conference on Theoretical and Methodological Issues in Machine Translation*.

Nyberg, E., J. Callan, J. Carbonell, R. Frederking, J. Lafferty, A. Lavie, T. Mitamura (2002). JAVELIN: Justification-based Answer Valuation through Language Interpretation, proposal submitted to ARDA BAA 01-01. See http://www.lti.cs.cmu.edu/Research/JAVELIN

Nyberg, E. and H. Daume (2001). Integrated information management: An interactive, extensible architecture for information retrieval, *Proceedings of HLT 2001*.

Pressman, R. (2000). *Software Engineering: A Practitioner's Approach*, 5th Edition, New York: McGraw-Hill.

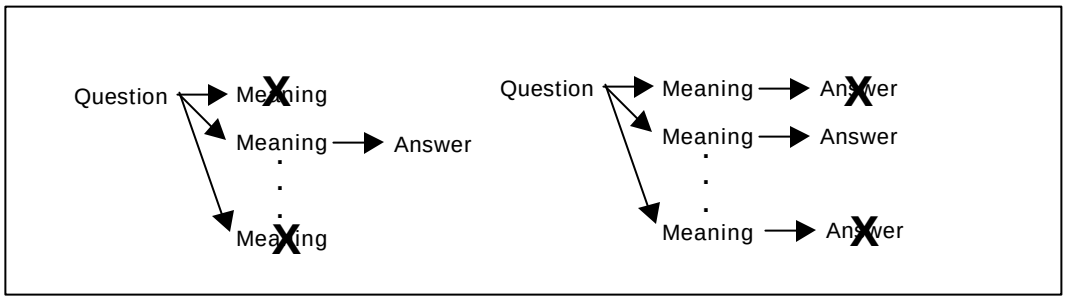


Figure 1: Ambiguity Resolution

Position statement: Inference in Question Answering

Bonnie Webber*, Claire Gardent†, Johan Bos*

*Division of Informatics
University of Edinburgh
Edinburgh EH8 9LW, UK
{bonnie,jbos}@cogsci.ed.ac.uk

†CNRS – LORIA
BP 239 – Campus Scientifique
54506 Vandoeuvre-les-Nancy, FRANCE
claire.gardent@loria.fr

Abstract

One can only exploit inference in Question-Answering (QA) and assess its contribution systematically, if one knows what inference is contributing to. Thus we identify a set of tasks specific to QA and discuss what inference could contribute to their achievement. We conclude with a proposal for *graduated test suites* as a tool for assessing the performance and impact of inference.

1. Introduction

Our point in this position statement is that, to use inference in Question-Answering (QA) in a way that will support what Barr and Klavans (2001) call *component performance evaluation* – assessing the performance of system components and determining their impact on overall system performance – one must identify specific *question-answering tasks* that can potentially gain by exploiting inference. In the first generation of QA systems (i.e., those designed to answer questions in terms of information in structured databases), only a few QA tasks were seen to need inference. In all cases, inference complemented the extensional process of relational (SQL) database querying, through reasoning on the concepts involved:

- Stallard (1986) used terminological reasoning (in a description logic) for the task of mapping from the logical form (LF) representation of a user's query and the concepts it was couched in, into the concepts and relations that formed the *data model* for the database.

- In the context of QA from multiple databases, inference was used in (Hendrix et al., 1978) in the task of developing plans for what databases to access for concept extensions, which would then be combined to produce an answer.

- Kaplan (1982) used inference on the query and its presuppositions for the task of generating a response to a question whose direct answer was not deemed useful.

- Pollack (1986) used inference on the query and an enhanced *data model* for the task of identifying and correcting user misconceptions that underlay otherwise unanswerable (or not usefully answerable) questions.

- In (Mays, 1984; Mays et al., 1982), when a question couldn't be usefully answered at the time it was asked, inference in the form of a temporal tableaux reasoner was used to generate a response to a question whose direct answer was not deemed useful. Specifically, it was used to identify whether the situation described in the question could occur in the future. If so, the QA system could offer to monitor for its occurrance, at which time the question could be answered.

Not all of these QA tasks are relevant to today's (or even tomorrow's) Open-Domain QA systems, which are designed to answer questions on the basis of *unstructured data* (i.e., free text). Nevertheless, it is still the case that there are places where inference can enhance the capabilities of Open-Doman QA systems (Burger et al., 2000; Hirschmann and Gaizauskas, 2001) and/or improve the quality and/or accuracy of their answers. As already noted, our point in this position statement is that, to use inference to these ends, one must identify specific *question-answering tasks* that will drive inference. This will then allow development of the kinds of *graduated test suites* with respect to which *evaluation* can be carried out on both the QA system and the inference engines themselves.

Note that the position we are taking here is very similar to that in (Hobbs et al., 1993), where the authors identify a set of *discourse tasks* that need to be solved in order to explain why the sentences of a text, in combination, would be true. These *discourse tasks* include (but are not limited to): interpreting compound nominals; resolving definite referring expressions; further specifying vague predicates; identifying how predicates apply to their arguments; disambiguating the arguments to predicates; determining coherence relations between adjacent segments of text; and detecting relation of an utterance to the speaker's overall plan. These, in turn, may depend on solving lower-level tasks such as resolving attachment and/or word sense ambiguities, resolving anaphora, and filling in missing (semantic) arguments. But by first specifying the discourse tasks, the authors can show exactly how inference (in their case, *weighted abduction*) can potentially – with efficient search and sufficient background knowledge – be used to solve them. (Note that weighted abduction is not a technique for *forward reasoning*. So any discourse task that requires determining the additional conclusions that can be drawn from a text may require another form of reasoning.)

In the first part of this statement, we identify a set of *question-answering tasks* in which inference could allow enhanced or extended QA services. Our goal is not to comment on what has or has not already been done in using inference in Open-Domain QA systems, but rather to lay out general areas where inference can contribute. We conclude by saying a bit more about *graduated test suites.*

2. QA Tasks

For this short position paper, we restrict the label *QA tasks* to ones that follow from a *functional role* of question or answer, rather than as text *per se*. That is, it is well known that inference can support discourse processing: texts can be parsed using *deduction* – it is what DCGs are all about – and (theoretically) they can be assigned a consistent explanatory interpretation using a combination of *weighted abduction* (Hobbs et al., 1993) and *consistency checking* (Blackburn and Bos, forthcoming). While this kind of interpretation can knit together elements of a text and supply missing (implicit) elements of its fabric, and thereby be critical for deriving answers to particular questions or even particular classes of questions, discussing the role that inference can play in discourse understanding requires its own paper, which we or other people should write.

Similarly, QA interactions are *dialogues*, and work done by Perrault, Cohen, Allen, Litman, Pollack, Walker and others has clearly shown that inference is needed to support dialogue processing – e.g., to decide what a question is really asking for. But this too is a large enough area to require its own paper.

Our focus in this paper then is on the significant set of tasks that remain after both discourse and dialogue understanding are, for the moment, put aside. Among these, we can identify several where inference could provide enhanced or extended QA services.

2.1. Expanding the search criteria for *potential* answers

It is standard procedure in QA to establish search criteria based on the question that has been posed. These search criteria make up the formal *query*, which is used to find *potential* answers in the form of candidate documents that may provide evidence for or contain a *proper* answer.

To increase the yield of potential answers, alternative terms can be added to the query. While this does not intrinsically require inference, what inference can do is expand queries with truth-functionally or defeasibly equivalent *global* reformulations of the original question. These can be used to augment the query with terms that could not have been identified using essentially *local* translation of individual words that ignores their context and functor-arguments dependencies, including implicit (semantic) arguments. For example, abductive reasoning on the question

(1) What do penguins eat?

(solving the implicit argument of *when* the eating event takes place – the same generic "in general" as the generic subject penguins) might produce a defeasibly equivalent version in terms of their *staple diet*. This term would not be added for a question like

(2) What did the characters eat in the seduction scence from the film "Tom Jones"?

which has its (optional) event argument instantiated.

Inference can also expand a query with one-way *entailments* of the original question. For example, being *awarded a degree in Computer Science* (CS) entails being *enrolled for a CS degree*. Given the question

(3) How many students were enrolled in Computer Science at Cambridge last year?

computing its one-way entailments would allow the query to be expanded with *award*$\wedge$*degree*.

Finally, inference can expand queries through subconcepts that form a *partition* (i.e., disjoint cover) of a concept in the original query; a distinct sub-query can be formed for each one. In this way for instance, the query

(4) How many people work for IBM?

could be decomposed into a set of sub-queries such as e.g., *How many men work for IBM? How many women work for IBM* or *How many white collar workers does IBM have? How many blue collar workers does IBM have?*.

Although we have discussed these expansion techniques in terms of constructing a query (either initial or follow-up, in case the initial query does not produce sufficient results), the same techniques could benefit the *ranking* of potential answers with respect to the question, if *recall* on the original query is felt to be sufficient.

2.2. Determining *proper* answers from *potential* answers

A proper answer to a wh-question may be found within a single clause, or it may be distributed through the potential answer (*answer locality*). Moreover, a proper answer may be explicit in the text (i.e., derivable simply by pattern matching), or it may require inference or other method of information fusion (*answer derivability*).

Even where an answer appears to be *explicit* in a text, inference can help determine whether it is a *proper* answer (Bos and Gabsdil, 2000), as with the following potential answers to:

(5) Q: Who invented the electric guitar?
A1: Mr. Fender did not invent the electric guitar.
A2: The electric banjo, cousin of the electric guitar, was invented by Bela Fleck.

A proper answer to this question must entail either (1) that there is someone who invented the electric guitar, or (2) that there is no such person, or (3) that it is true of everyone. All of these are logical relations between a potential answer and a representation of the question in terms of its question domain D (here, persons) and its body B (here, inventing the electric guitar). As such, inference can be used to determine whether any of these relations hold.

Inference can also help when *proper answers* are only implicit in *potential answers*. In (Hobbs et al., 1993), Hobbs et al. show that *weighted abduction* can be used to solve a variety of *discourse tasks*, thereby making explicit information that is implicit in a text. This can be applied to potential answers. For example, a potential answer to the question

(6) Where do condors live?

might contain the compound nominal *the California condor*. As in resolving "the Boston office" (Hobbs et al., 1993), this can be (abductively) resolved to condors whose location is California. That this is a matter of abductive inference rather than simple pattern matching, can be seen by not wanting to draw similar conclusions in determining proper answers to the similar question

(7) Where do terriers live?

Here, compound nominals such as "Yorkshire terrier", "Boston terrier", "West Highland terrier", etc. in potential answers would yield such incorrect proper answers as Yorkshire, Boston, etc.

There is much more to be explored here. Nevertheless, it is clear that inference can be used to support more than one aspect of this task.

2.3. Comparing *proper* answers to wh-questions

The way in which answers are sought in open-domain QA means that one cannot avoid the problem of determining whether proper answers derived from different potential answers (candidate documents) are the same (i.e., mutually entail one another) or different. In the latter case, one may also not be able to avoid the problem of determining whether (i) one answer is more specific than another (i.e., the more specific answer entailing the more general one, but not vice versa); (ii) two answers are mutually consistent but not entailing in either direction; or (iii) two answers are inconsistent. Determining such relations among proper answers becomes a QA task for Open Domain QA, where it was not one for database QA because the underlying relational DB query system was able to recognize and remove all duplicates.

The outcome of such determination depends on whether the original question is taken to have a single answer (a unique individual or property or set) or alternative answers, the set of which is of unknown cardinality. Whatever the reason, these are problems that inference can help solve.

- Answers determined to be equivalent (mutually entailing) can be replaced by a single member of the equivalence class;
- Answers that differ in specificity (one-way entailing) can be replaced by either the most specific one (as with the answer to *When was the Bastille taken?*, where *14 July 1789* is preferred over the less specific *14 July* and *1789*) or by a conjunction of the most specific answers (as with answers to *Who is Noam Chomsky?*, where *MIT linguist*$\wedge$*left-wing activist* is the preferred way to combine the answers in the set *MIT linguist, linguist, MIT academic, political activist* and *left-wing activist*);
- Answers that are mutually consistent but not entailing can be replaced by their conjunction (as with *MIT linguist* and *left-wing activist* above);

- Answers that are inconsistent are the only true alternatives. In the case of questions with unique answers, only one of them can be correct. In the case of questions with alternative answers such as *Where do penguins live?*, all the alternatives may be distinct proper answers.

2.4. Comparing questions

Where efficiency is a goal of QA, it can be supported by determining whether a new question is one that has previously been answered (Harabagiu et al., 2001) or is related in a systematic way to one that has previously been answered. (This is the reason that FAQ-lists exist.) Inference is a valid way of computing both *equivalence* relations between questions and *subsumption* – i.e., whether one question is more specific than another one. The latter allows two different forms of answer re-use. Consider the questions

(8) Where can I go skiing in the Northern Hemisphere in June?

(9) Where can I go for winter sports in the Northern Hemisphere in June?

If one has cached the answer to (8), then one has a partial answer to question (9), which subsumes it. Conversely, if one has already cached the answer to the subsuming question (9), that answer may contain or provide a basis for an answer to question (8). That is, if (9) has been answered by answering the set of questions that follow from each possible way of instantiating the general term "winter sports", then one already has an answer to (8). On the other hand, if question (9) has been answered in general, then (much as with the "linked" questions in TREC-10) sources for that answer might prove a good place to start looking for an answer (8), rather than posing it against a completely open domain.

2.5. Determining *proper* answers to yes/no questions

One may take the set of proper answers to a yes/no question to comprise simply *yes* and *no*, or one may take it more broadly to include temporal and/or modal qualifiers as well – eg. *possibly, sometimes, it depends*, etc. In the first case, determining a proper answer requires identifying what support exists for a positive answer (*yes*); what support exists for a negative answer (*no*); and on which side the support is stronger. Practically, this could involve separate queries – one seeking evidence for the positive assertion, the other, for the negative assertion. These queries could differ because lexical items can have distinct negative-polarity counterparts. For example, given the question

(10) Does Anacin contain any stimulants?

a query seeking evidence for the positive statement might contain the terms ANACIN, CONTAIN and STIMULANT, while the query seeking evidence for the negative statement might contain the terms ANACIN, LACK and STIMULANT. But because *potential answers* retrieved in response to such questions may themselves contain explicit negation (i.e., *no* or *not*), deciding what they support requires determining the scope of negation. Here, inference can determine which of the readings are consistent. Inference can also be used as discussed in Section 2.2. to determine whether two pieces of *evidence* are the same or different, so that instances of the same evidence or instances of stronger and weaker evidence aren't multiply counted.

In general, it is easier to find positive evidence than negative evidence, as what does not hold is most often conveyed implicitly, by the lack of evidence for it (i.e., the *closed-world assumption*). But for certain yes/no questions, evidence for a negative answer may be easier to come by than for a positive one. For example, in a question with a universal quantifier such as

(11) Did Larsson score in every game he played for Celtic?

a single piece of negative evidence (e.g., "Larsson failed to score in Tuesday's game") is needed to justify a negative answer, while a positive answer requires either a potential answer that itself contains a universal quantifier or a set of potential answers that cover the entire set of games. The latter is essentially (extensional) database question-answering, with the *closed-world assumption* that the database covers all positive instances.

2.6. Generating responses in lieu of or support of a direct answer

Unlike in TREC-9, TREC-10 systems were asked to identify when they couldn't answer a question. In database QA, finding no answer to a question was not an uncommon occurence. One reason for this occurring was failure of a presupposition in the question. For example, the question

(12) Have any women been awarded a Pulizer prize for sports journalism?

may have the direct answer *None* because the existential presupposition that there is a Pulizer prize for sports journalism is false. Hence, techniques were developed (Kaplan, 1982) for recognising presupposition failure and for generating responses such as *There is no Pulizer prize for sports journalism.* But as shown

in (Blackburn and Bos, forthcoming), verifying presuppositions involves inference in order to check their consistency and informativity in context.

Another reason for not being able to answer a question is that *positive* information is lacking. Here, a partial response can be formulated if *negative* information can be found that *excludes* something from the set of proper answers. For example, given the question

(13) Which French cities did Reagan like?

information to the effect *Reagan disliked Paris* provides a useful partial response. Inference can be used to recognize that an individual is excluded from the set of proper answers.

A third situation motivating a response is the case of negative answers to extensional yes/no questions, which are rarely very informative – e.g.

(14) Q: Did Hearts played a home game against Celtic in January?
A: No.

In such cases, the answer to a "weaker" question – one that can be computed from the original one by subsumption reasoning, may provide the basis for a useful response – e.g. *Did Hearts play a game against Celtic in January?* or *Did Hearts play a home game against Celtic?* or *Did Hearts play a home game in January?*. More complex questions, such as ones containing quantification and/or negation, may require more complex subsumption reasoning to establish weaker questions that are worth posing.

Note that weakening the question only makes sense for questions answered extensionally, not ones answered through inference or pattern matching such as

(15) Do penguins migrate?[1]

Other situations in which responses are useful in lieu or support of a direct answer, many of which require forms of inference, are described in (Webber, 1986).

3. Graduated Test Suites

While TREC evaluation of QA systems has focussed on the full end-to-end task, some systems have also carried out what Barr and Klavans (Barr and Klavans, 2001) call *component performance evaluation* – assessing the performance of system components and determining their impact on overall system performance. The components of interest here are those that use inference. We see *graduated test suites* as a tool for assessing their performance and impact, allowing: (1) comparison against similar components that do not use inference; (2) comparison of components that differ in what inference tools they use; and (3) assessment of the impact of improvements in inferential ability. We also see graduated test suites as a way of evaluating automated reasoning tools on the inference problems raised by QA.[2]

We now discuss two of the above QA tasks, making explicit what one would expect to see in a distinct test suite for each. As in TREC, developing the test suites would involve carefully crafting a set of examples to the correct level of difficulty, fixing evaluation criteria and delimiting in a more precise way the linguistic task involved.

Expanding the query. Section 2.1. identifies four ways of expanding the query: through equivalence, through entailment, through multiple sub-queries and through abduction. For each of these tasks, inference can be involved as follows.

When expanding the query with semantically equivalent reformulations, inference can be used in at least one of two ways: First, given a subsumption based hierarchy KB encoding relations between word meanings, inference can be used to *find* the set of (structured) concepts which are logically equivalent to the structured concept representing the initial query. Alternatively, for reformulations produced by some other mechanism (e.g, parsing the query and then generating paraphrases from the resulting semantic representation(s)), inference can be used to *check* that they are indeed semantically equivalent.

Similarly, when expanding the query with more specific variants, inference can be used either to *find* within a hierarchy, the set of most specific concepts subsumed by the concept representing the query, or, for potential variants found by other means, simply to *check* that each indeed stands in some kind of entailment relation to the initial query.

Thirdly, when expanding concepts (and/or sets of concepts) in the query into *partitions* (i.e., disjoint covers) of more specific sub-concepts, the task for automated reasoners would be to check that the conjunction of queries $Q_1, \ldots, Q_n$ obtained by replacing a concept in the original query Q by a partition of its immediate sub-concepts is equivalent to the original query.

Finally, queries can be expanded by making implicit information explicit. This requires some kind

[1] Many types of penguin migrate, swimming north each autumn in the Southern Hemisphere and south each spring.

[2] Automated reasoners have been optimised for their performance on problems from mathematics and logic. As this is not necessarily optimal for NL problems, we need to drive their optimisation in this direction. That is the reason for having test suites for both QA components and automated reasoners.

of abduction – e.g, weighted abduction (Hobbs et al., 1993) or model building (Gardent and Konrad, 2000a; Gardent and Konrad, 2000b). With the first, the reasoner is given a semantic representation of the query, along with relevant world, domain and/or lexical knowledge and returns the cheapest explanation (proof) of the query, making explicit the hypotheses (either abduced or assumed) that support it. Similarly, model building will produce a (minimal) model satisfying the formula which encodes the explicit and implicit information expressed by the query.

In all cases, the information (facts in model or logical formulae) resulting from query expansion can be converted to a form appropriate to the query. If queries are Boolean combinations of key words and/or phrases, NL Generation techniques can be applied to each semantic component to produce a parse tree whose leaves constitute a string of lexical *lemmas*, from which key words and phrases can be identified and added to the query.

Determining proper answers. For **wh-questions with a single answer**, the problem of determining a proper answer from a potential answer depends on (i) the *expected answer type* (positive, negative, unknown); (ii) the *answer locality* (whether the answer is contained in a single clause or distributed over the text), and (iii) the *derivability* of the answer (whether it is explicit in the text and derivable simply by pattern matching, or it requires inference or other method of information fusion).

Test-suite examples could therefore be divided into 12 classes, of different complexity, depending on the values of these factors. For example, consider *expected answer type*. Formulated in first-order logic, with ϕ_A representing the meaning of the potential answer A, D the domain of the question and B its body, (1) if the expected answer type is positive, there is at least one object having the properties set by the question. So the inference task is simply: **Prove** $\models \phi_A \rightarrow \exists x(D(x) \wedge B(x))$**.** (2) Alternatively, if the expected answer type is negative, there is no object having the properties set by the question. So the inference task is: **Prove** $\models \phi_A \rightarrow \neg\exists x(D(x) \wedge B(x))$**.** (3) Finally, if the expected answer type is unknown, then *both* the above inference tasks are required.

For **questions with multiple answers**, we can only comment now on the use of inference for questions that can be expanded into a set of more specific subqueries with known cardinality, such as

(16) What is the longest river on each continent?

which can be expanded into *What is the longest river in Europe? What is the longest river in Asia? ....* Once expanded in this way, each sub-query is a simple wh-question with a single answer. This is then the case discussed earlier.

4. Summary

There is no question that QA would not also be enhanced through the use of inference in *discourse tasks* involved in finer-grained examination of the texts retrieved in response to user-queries. It would likewise be enhanced by the use of inference in *dialogue tasks* involved in understanding the user's current utterance with respect to the current QA dialogue. Here we have focussed solely on the use of inference in *QA tasks* – tasks that follow from the *functional role* of a question or an answer – and how it could contribute to achieving these tasks, over and beyond methods that don't use inference.

When considering the development of *graduated test suites* to assess system performance on QA tasks and its impact on overall system performace (and also the performance of automated reasoning tools), it makes sense to consider the use of previous TREC questions and the set of passages (potential answers) that the retrieval components of TREC QA systems have returned in response. The usefulness of doing so is most obvious in the case of two of the tasks discussed here: determining proper answers from potential answers and comparing proper answers to wh-questions. What now requires discussion is what to do next.

5. References

Valerie Barr and Judith Klavans. 2001. Verification and validation of language processing systems: Is it evaluation? In *Proceedings of ACL Workshop on Evaluation Methodologies for Language and Dialogue Systems*, Toulouse, France.

Patrick Blackburn and Johan Bos. forthcoming. *Computational Semantics.* Current draft available from http://www.comsem.org.

Johan Bos and Malte Gabsdil. 2000. First-order inference and the interpretation of questions and answers. In *Proceedings of Gotelog 2000*, pages 43–50, Goteborg, Sweden.

John Burger, Claire Cardie, Vinay Chaudhri, Robert Gaizauskas, and *et al.* 2000. Issues, tasks and program structures to roadmap research in question & answering. Technical report, National Institute of Standards and Technology. Available on-line at http://www-nlpir.nist.gov/projects/duc/papers/QA.roadmap-paper_v2.pdf.

Claire Gardent and Karsten Konrad. 2000a. Interpreting definites using model generation. *Journal of Logic, Language and Information*, 1(2):193–209.

Claire Gardent and Karsten Konrad. 2000b. Understanding each other. In *Proceedings, 1^{st} Annual Meeting of the North American Chapter of the ACL*, Seattle WA.

Sanda Harabagiu, Dan Moldovan, and et al. 2001. Falcon: Boosting knowledge for answer engines. In *Proceedings of the 9^{th} Text Retrieval Conference (TREC 9)*, pages 479–488, National Institute of Standards and Technology. Available on-line at http://trec.nist.gov/pubs/trec9/papers/smu.pdf.

Gary Hendrix, Earl Sacerdoti, Daniel Sagalowicz, and Jonathan Slocum. 1978. Developing a natural language interface to complex data. *ACM Transactions on Database Systems*, 3(2):105–147.

Lynette Hirschmann and Rob Gaizauskas. 2001. Natural language question answering: The view from here. *Natural Language Engineering*, 4.

Jerry Hobbs, Mark Stickel, Paul Martin, and Douglas Edwards. 1993. Interpretation as abduction. *Artificial Intelligence*, 63(1-2):69–142.

Jerrold Kaplan. 1982. Cooperative responses from a portable natural language database query system. In Michael Brady and Robert Berwick, editors, *Computational Models of Discourse*, pages 167–208. MIT Press, Cambridge MA.

Eric Mays, Aravind Joshi, and Bonnie Webber. 1982. Taking the initiative in natural language data base interactions: Monitoring as response. In *Proceedings of the European Conference on Artificial Intelligence*, pages 255–256, Orsay, France.

Eric Mays. 1984. *A Modal Temporal Logic for for Reasoning about Changing Data Bases with Applications to Natural Language Question Answering.* Ph.D. thesis, Dept of Computer and Information Science, University of Pennsylvania, Philadelphia PA.

Martha Pollack. 1986. *Inferring Domain Plans in Question-Answering.* Ph.D. thesis, Department of Computer & Information Science, University of Pennsylvania.

David Stallard. 1986. A terminological simplification transformation for natural language question answering systems. In *Proceedings of the 24^{th} Annual Meeting, Association for Computational Linguistics*, pages 241–246, Columbia University.

Bonnie Webber. 1986. Questions, answers and responses. In Michael Brodie and John Mylopoulos, editors, *On Knowledge Base Systems*, pages 365–401. Springer-Verlag, New York.

The Challenge of Technical Text

Michael Hess, James Dowdall, Fabio Rinaldi

University of Zürich, Institute of Computational Linguistics
Winterthurerstrasse 190, CH-8057 Zürich, Switzerland
{hess,dowdall,rinaldi}@ifi.unizh.ch

Abstract

When evaluating and comparing Answer Extraction and Question Answering systems one can distinguish between scenarios for different information needs such as the "Fact Finding", the "Problem Solving", and the "Generic Information" scenarios. For each scenario, specific types of questions and specific types of texts have to be taken into account, each one causing specific problems. We argue that comparative evaluations of such systems should not be limited to a single type of information need and one specific text type. We use the example of technical manuals and a working Answer Extraction system, "ExtrAns", to show that other, and important, problems will be encountered in the other cases. We also argue that the quality of the individual answers could be determined automatically through the parameters of correctness and succinctness, i.e. measures for recall and precision on the level of unifying predicates, against a (hand-crafted) gold standard of "ideal answers".

1. Introduction

The classical type of *information need* satisfied by existing IR systems can be described with the scenario of "Essay Writing": If you have to write an essay on a given topic you need to locate as much backup material dealing with this topic as possible, i.e. preferably whole documents[1].

Increasingly, more specific types of information needs become important. *First*, one need not catered for by the "Essay Writing" scenario is a determination to locate factual knowledge about individually identifiable entities, concerning their location in time or space, their properties, or their identity with other entities. This could be called the "Fact Finding" scenario, and it is the situation assumed by the QA Track of TREC. The questions are factual questions ("where is/who is XYZ"). One source of such information is, of course, news items but also includes encyclopedias, text books, and fact sheets.

A *second*, equally important, information need beyond the "Essay Writing" scenario arises in situations where concrete problems require explicit solution(s) from a collection of documents. This could be called a "Problem Solving" scenario, and the questions asked are procedural ("how do I do XYZ"). A typical, real world, example is that of an airplane maintenance technician who needs to repair a defective component. He must locate in the massive maintenance manual of the aircraft the exact description of the specific repair procedure. Other text types that contain procedural information are "case data bases" used for trouble shooting purposes, operational handbooks, and some types of scientific articles (e.g. diagnostic and therapeutic reports in medicine).

Third is the situation where you need to find information about principles and regulations, i.e. what one might call the "Generic Information" scenario. The typical questions are definitional ("what is"), and the typical texts consulted in this situation are on-line encyclopedias, but also technical standards publications. Many technical manuals also contain numerous definitions of concepts or devices. It can also be argued that deontic texts (laws etc.) also fall under this heading, and they are extremely important in society.

What users need in the "Fact Finding", "Problem Solving", and "Generic Information" scenarios are systems capable of finding those exact (parts of) sentences in document collections that constitute the answer to their question. Depending on the type of question ("where is/who is", "how do I", "what is") different problems will be prominent to different degrees. Thus, named entities are important for answering factual questions but less so for problem solving and definitional questions. There is also evidence that for the latter two types of questions a deeper (syntactic and semantic) analysis of questions is needed than for the factual ones. In order to define standards for comparative evaluations that are not biased towards one particular type of information need, examples of queries and texts of different types should be used from the very beginning.

In the present position statement we will briefly describe ongoing research in the related fields of Question Answering (QA) and Answer Extraction (AE), primarily in the dual context of the TREC QA track (section 2.) and of our own work on the first text type mentioned above, i.e. technical manuals (section 3.). Later we will present some of the problems that are specific to different text types (section 4.), briefly consider the difficulties of evaluating AE systems (section 5.), and finally mention the resources used in our work (section 6.). As relative 'outsiders' we explicitly aim at providing a critical and, in some respects, dissenting voice, giving the view of somebody approaching Question Answering from a perspective different from that defined (and circumscribed) by the TREC QA track.

2. Results from TREC

Results from the two first TREC Question Answering Tracks (Voorhees, 2000; Voorhees and Harman, 2001) seemed to show that standard, keyword based, IR techniques are not sufficient for satisfactory Answer Extraction. When the answer is restricted to a very small window of text (50 bytes), systems that relied only on those techniques fared significantly worse for the kind of questions used in

[1] It has been often observed that Information Retrieval should rather be called "Document Retrieval".

the QA track than systems that employed some kind of language processing.

More successful approaches employ special treatment for some terms (Ferrett et al., 2001) and named entity recognition (Humphreys et al., 2001), or a taxonomy of questions (Hovy et al., 2001). Interestingly, some sort of convergence appears to be emerging towards a common base architecture which is centered around four core components (Abney et al., 2000; Pasca and Harabagiu, 2001). Passage Retrieval (Clarke et al., 2001) is used to identify paragraphs (or text windows) that show some general similarity to the question (according to some system specific metric), a Question Classification module is used to detect possible answer types (Hermjakob, 2001), an Entity Extraction module analyzes the passages and extracts all the entities that are potential answers, and finally a Scoring module (Breck et al., 2001) ranks these entities against the question type, thus leading to the selection of the answer(s).

The results of this general design are promising for the kind of factual questions that make sense in the context of news messages. Since such questions ask mostly about properties of individually identifiable entities, good named entity recognition can go a long way towards finding informative text passages. However, for other types of questions (procedural and definitional) we need to be able to analyze other types of constructions, and pinpoint answers more precisely. This means that the choice of a single type of text for the purpose of comparative evaluation creates the risk of "over-fitting" in that all competitors converge on the techniques used by the most successful system for this particular type of text. This effect tends to stifle innovation rather than foster it, and we think that a wider range of texts should be used in comparative evaluation from the beginning to counteract this danger.

It appears that, partly, the problem has already begun to emerge in the latest TREC QA track (TREC10). On one hand, many systems are converging towards the 'generic AE system design' described above, on the other hand, the system that did best (Soubbotin and Soubbotin, 2001) made massive use of heuristics and patterns, that might have limited portability to other domains and other types of applications.

3. ExtrAns

Over the past few years our research group has developed an Answer Extraction system (ExtrAns) (Rinaldi et al., 2002; Mollá et al., 2000) that is mainly geared towards procedural and definitional questions over technical texts.

Two real world applications have so far been implemented with the same underlying technology. The original ExtrAns system is used to extract answers to arbitrary user queries over the Unix documentation files ("man pages"). A set of 500+ unedited man pages has been used for this application. An on-line demo of ExtrAns can be found at the project web page.[2]

More recently we tackled a different domain, the Airplane Maintenance Manuals (AMM) of the Airbus A320. The combined challenges of an SGML-based format and the more technical nature of the text and a larger size (120MB)[3] have been met using the original basic architecture (Fig.1), plus a specialized XML based tokenizer and a new CSS-based display utility.

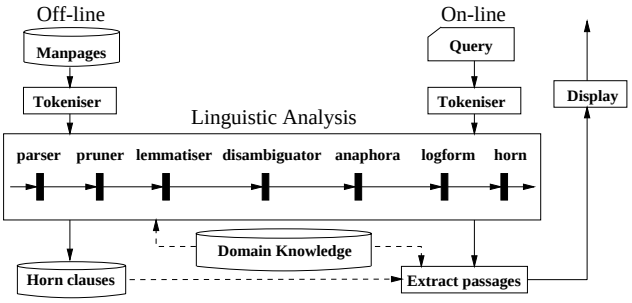


Figure 1: Architecture of the ExtrAns system

Essentially, ExtrAns extracts answers from documents by semantically comparing queries against document sentences. This is achieved by deriving, from documents and queries, the basic semantic relationships of each sentence and representing them as Minimal Logical Forms (MLF). These are representations that use selected reification and underspecification to keep them open to dynamic, incremental and non-destructive extension, depending on requirements. Answers are derived from these logical forms by deductive proof. This representation is both expressive enough to allow non-trivial comparison and computationally "light" enough for real world applications. True, this approach requires expensive deep linguistic analysis of questions and documents, involving syntax, semantics and consideration of lexical alternations (synonyms and hyponyms) but it returns, in exchange, the exact answer sentences (ideally) and often manages to even determine the individual parts of sentences constituting the exact answer(s) to user questions.

The general design of the system is fairly standard. A (very powerful) tokenizer identifies word and sentence boundaries as well as domain specific multi-word terms. Once tokenized, sentences are parsed using Link Grammar (LG) (Sleator and Temperley, 1993). Link Grammar's ability to predict the syntactic requirements of unknown words ensures that an analysis of all sentences is returned. So ExtrAns always produces MLFs, possibly extended with special predicates that mark any unprocessed tokens as "keywords". Multi-word terms (to be extracted independently and beforehand) are parsed as single syntactic units. Relieving LG of the need to compute the internal structure of such terms reduces the time and space involved for parsing technical text by almost 50%.

A corpus-based approach (Brill and Resnik, 1994) then disambiguates prepositional phrase attachments as well as gerund and infinitive constructions. An anaphora resolution algorithm (Lappin and Leass, 1994) resolves sentence-internal pronouns. The same algorithm can also be applied

[2] http://www.ifi.unizh.ch/cl/ExtrAns/

[3] Still considerably smaller than the size of the document collections used for TREC.

to sentence-external pronouns but this is not (yet) done in ExtrAns.

From the resulting disambiguated linkage, semantic relations between verbs and arguments as well as modifiers and adjuncts are expressed as a MLF. Strict underspecification ensures this only involves objects, eventualities and properties. These predicates are conjoined, and all variables are existentially bound with maximal scope. By way of an example, (1) represents the sentence, *"A coax cable connects the external antenna to the ANT connection"*:

```
(1) holds(o1),
    object(coax_cable,o2,[v3]),
    object(external_antenna,o3,[v4]),
    object(ANT_connection,o4,[v5]),
    evt(connect,o1,[v3,v4]),
    prop(to,p1,[o1,v5]).
```

ExtrAns identifies three multi-word terms, translated into (1) as the objects: `v3`, a coax_cable, `v4` an external_antenna and `v5` an ANT_connection. The entity `o1` represents the fact of a 'connect' eventuality involving two objects, the coax_cable and the external_antenna. This reified argument, `o1`, is used again in the final clause to assert the eventuality happens '*to*' v5 (the ANT_connection).

The utility of reification, yielding the additional arguments `o1`, `o2`, `o3` and `o4` as hooks to the abstract entities they denote is that the expression (1) can now be modified by monotonically adding constraints over these entities without destructively rewriting the original expression (Schneider et al., 1999). So the sentence *"A coax cable **securely** connects the external antenna to the ANT connection"* changes nothing in the original MLF, but additionally asserts (2) that `o1` (i.e. the fact that the coax cable and the external antenna are connected) is *secure*:

```
(2) prop(secure,p8,o1).
```

This MLF only needs to refer to the reification of an **eventuality** for further modification but other, more complex, sentences will need to refer to the reifications of **objects** (e.g. for non-intersective adjectives) or of **properties** (e.g. for adjective modifying adverbs).

ExtrAns extracts the answers to questions by forming the MLF of the question and running Prolog's theorem prover to find the MLFs from which the question can be derived. So,

"How is the external antenna connected ?'

becomes:

```
(3) holds(V1),
    object(external_antenna,O2,[V5]),
    evt(connect,V1,[V4,V5]),
    object(anonymous_object,V3,[V4]).
```

If a sentence in the text used as a knowledge base asserts that the *external antenna* is connected to or by *something*, the query will succeed. This *something* is the anonymous object of the query. If there are no answers (or too few) ExtrAns relaxes the proof criteria by introducing hyponymy related tokens as part of the MLF. Additionally, a sentence identifier indicates from which tokens the predicate is derived (not shown in the example above). This information is used to highlight the (relevant parts of the) answer in the context of the document (see Fig. 2).

This kind of very parsimonious representation could appear too "semantically weak" for general QA. This may be true but it is optimized for the task at hand (AE) and can be extended, at will, for more demanding tasks (such as full QA). The MLFs can also be used to ensure that sentences are retrieved that are, in strictly logical terms, not correct answers, but they are useful nevertheless. Thus (4i-ii) are useful (albeit not logically correct) answers, in addition to the correct answers (4iii-iv).

(4)
i. The external antenna must not be directly connected to the control panel.
ii. Do not connect the external antenna before it is grounded.
iii. The external antenna is connected, with a coax cable, to the ANT connection on the ELT transmitter.
iv. To connect the external antenna use a coax cable.

4. Text Types, Question Types, and Problem Types

At present, discussions in the TREC community around the further development of Answer Extraction and Question Answering (e.g. in the "Roadmap" document (Burger et al., 2001)) address a very large number of problem and question types, many of them very thorny. However, they do so almost exclusively against the background of *one specific document type*, viz. newspaper texts.

We feel, on the basis of six years' of development and experimentation with Answer Extraction systems, that this exclusive focus on a single, very specific, type of document is not ideal, and that other document types should be considered from the beginning. There are three reasons for this:

1. Processing Strategies developed for newspaper texts become less relevant to users accessing increasing volumes of technical data.
2. Some important problems of AE/QA hardly occur in newspaper texts.
3. Some of the problems that are quite fundamental to any kind of AE/QA can be found in a more isolated, "pure", form in other types of text.

Concerning the *first* point, it is our experience that better access to archived newspaper texts and similar documents is low on the list of priorities for most potential users of QA/AE-Systems in industry, administration, and academia. One exception may be intelligence agencies with interests in monitoring news streams. However, systems that allow high-precision access to the information stored in texts covering narrower, more technical, domains would be welcomed by many organisations in business, administration, and research. Cases in point are (among others):

- Technical manuals of complex systems (any large technical system comes with massive manuals, most often in machine-readable form)
- On-line help systems (for software or other complicated products, such as some financial products)
- Customer queries (systems that process and answer e-mails and/or Web inquiries)
- Access to abstracts and full texts of scientific articles (such as Medline).

Concerning the *second* point, there are some important problems *not* given sufficient weight in the Roadmap document, due to the fairly specific characteristics of newspaper texts:

- **Domain specific terminology:** It is generally recognized that the compilation and use of terminologies is a top priority for the automatic processing of texts in technical applications. The *use* of a (reliable) terminology for a given domain makes the processing of texts vastly simpler, faster, and more useful than without (the quality of Machine Translation systems, for instance, remains dismal without terminology). However, the automatic *compilation* of terminologies ("term extraction") is basically an unsolved problem (none of the available methods produce really useful results). More work is needed in this field but the problem is very peripheral in the Roadmap document.
- **Procedural Questions:** In many of the applications mentioned above (apart from natural language interfaces to technical manuals also on-line help systems and customer e-mail processing systems) the procedural questions of the type *"How do I do X?"* ("How do I convert Apple files to UNIX text format?", "How can I move funds from checking to savings?") are of paramount importance. However, this type of question makes little sense in the framework of newspaper texts, and is therefore given too little attention in the Roadmap document.
- **Generic Questions:** In the documents used for the above-mentioned types of applications (but also in on-line encyclopedias etc.) many sentences are *generic* (timeless rules). Typical questions directed at such texts are *"How do you stop a Diesel engine?"* or *"What is a typhoon?"*. These, too, are relatively rare in newspaper texts (which normally describe individual, time-bound facts), and they are consequently not mentioned in the Roadmap document [4]. Although generic sentences are admittedly a thorny problem they must not be ignored, due to their general importance.

[4] A small number of definitional questions were included in TREC9. In TREC10 their number was significantly higher, due to the different source of the questions. It has however been observed that a corpus of newspaper articles is not the best place to search for answers to that type of questions (Voorhees, 2001).

Concerning the *third* point, there is a number of problems that are fundamental to any kind of AE/QA system, and that do occur in newspapers texts, but which are "drowned" by the numerous other difficulties resulting from the characteristics of newspaper texts. Among them are:

- **Intensional constructions**: Contrary to (almost) common belief, intensional constructions are fairly common in perfectly normal language, and not treating them properly results in wrong answers. Cases in point are "higher order verbs" (as in "pack **attempts** to store the specified files in a packed form" - it may not succeed) and intensional uses of adjectives (as in "Only the super-user can allocate **new** files" - they don't exist yet).
- **Anaphoric references:** Although it has been argued that anaphoric reference (by means of pronouns or definite noun phrases) is irrelevant for document retrieval purposes (or even damaging) the situation is definitely different for AE/QA. Crucial information is often contained in sentences that refer to entities *only* by anaphoric references. Moreover, information is often given in technical manuals just once, so even one missed pronominal reference may seriously impair retrieval performance. Even for the relatively simple task of named entity recognition we must often have recourse to some of the techniques needed for reference resolution ("Bill Gates of Microsoft" &... "Gates" ... "the Gates company" etc.).
- **Pluralities**: Reference to groups of objects (be it through plurals ["dogs"] or through conjunctions ["Fido and Rover"]) is a well-known headache, in particular due to the different possible readings of plural noun phrases (collective/distributive/cumulative: "Fido and Rover fought/barked/ate up the food"). While in many cases it is possible to leave underspecified the exact number of objects introduced by pluralities this is no option when we want to get exact numbers from textual documents (e.g. via "how many"-questions).

The specific characteristics of newspaper texts that somehow overshadow these problems are:

1. **Range of topics:** Due to the vast range of topics covered by newspapers the topic of *sense ambiguity* becomes a top priority problem (cf. "Where is the Taj Mahal?"). In more restricted domains we can usually get away with little or no sense disambiguation (and if we have to perform it, it is much simpler than in open domains). Since sense disambiguation is a very thorny problem, domains where it is not of primary importance would be most useful.

   The wide range of topics also creates the rather ill-understood problem of the type "original vs. copy" ("What is the height of the Statue of Liberty?" - only the original, no models thereof).

2. **Time-dependence of information:** The things described by newspapers are mostly time-dependent

("When was Yemen reunified?" or *"Who is the president of Ghana?")*. Keeping track of stages (i.e. the changes that the world is undergoing) is difficult (not least as we can, of course, refer to past states of affairs, and would therefore be able to process the various ways in which natural language encodes such information [the whole tense system!]).

3. **Volume of information:** The sheer volume of information in newspapers archives puts such a heavy burden on processing systems that a strong bias towards shallow analysis is created. One case in point is SRI's TACITUS which was replaced by FASTUS for the MUC competitions, for reasons of speed alone, although TACITUS is a much more powerful system.

Naturally, all these problems will have to be solved sooner or later but, in our opinion, the far more fundamental problems mentioned above could be approached best when kept somewhat sheltered from these minefields.

We certainly do not argue against the use of very large, TREC-like, collections of newspaper texts in the development and evaluation of AE/QA systems but argue for the early inclusion of more moderate volumes of technical texts representative of other, very important, types of documents.

5. Evaluation of AE/QA Systems

As experience gained in the past QA tracks has shown the question of how AE and QA systems shold be evaluated consists of at least two components:

1. What should the answer sets look like?
2. How should the quality of an answer be determined?

The *first* question concerns, among other things, the question of the size of the answer string and, connected with it, that of answer justifications. There is agreement that a fixed-length string that happens to contain the correct answer but in a wrong document context should not be counted as correct (e.g. the answer string "Bush" taken from a document written when George Bush was president but dealing exclusively with shrubs). However, this requirement forces assessors to consult the original document and determine whether the answer string is justified. Clearly a considerable element of uncertainty is entered into the evaluation that way (Is the justification allowed to be implicit in, and/or distributed over, the document? When is an answer justified?)[5].

For a pure AE system, i.e. one *retrieving* explicit answers rather than *computing* answers from possibly distributed, possibly implicit, information (as done by true QA systems) this problem can be contained somewhat by requiring systems to retrieve not fixed-length strings but (not necessarily contiguous) fragments of sentences of potentially unlimited length that, when concatenated, constitute the complete answer, ideally as a well-formed sentence, as seen in Fig. 2. That this is a sensible requirement becomes particularly obvious in technical domains. Consider, for instance, the question:

> **Do I need write permissions to remove a symbolic link?**

A 50-byte answer window may retrieve from the Unix manual, among others, the string:

```
" need write permission to remove a
symbolic link, "
```

Checking the document sentence will reveal that this string is a completely wrong answer as the sentence from which it was taken is:

> **Users do not need write permission to remove a symbolic link, provided they have write permissions in the directory.**

The arbitrary limit of 50 bytes just happened to cut off the crucial negation. However, requiring the AE system to return a complete, ideally well-formed, sentence will result in the justification to be part of the answer itself (in this case, the entire document sentence should be returned).

Another aspect of the first question concerns the *test queries*. Clearly, it is always better to use real world queries than queries that were artificially constructed to match a portion of text. By using, as we suggest, manuals of real world systems, it is possible to tap the interaction of real users with this system as a source of real questions (we do this by logging the questions submitted to our system over the Web). Another way of finding queries is to consult the FAQ lists concerning a given system available on the Web. By combining those two sources we compiled a list of 524 questions about the Unix domain. However, a large proportion of them is problematic as they have no answers in the document collection or are clearly beyond the scope of an automatic system (for example, if the inferences needed to answer a query are too complex even for a human judge). Nevertheless they are a useful starting point for a set of test queries in this domain.

Concerning the *second* issue, that of answer quality, the standard measures of Precision and Recall are not ideal for an Answer Extraction system, when applied to individual answer sentences. It can, in particular, be argued that Recall is significantly less important than Precision, as the aim of such a system is to provide (at least) one correct answer, rather than all the possible answers in a given collection. The user needs to find one good answer to a question and they are not interested in repeatedly finding the same answer.

In the Question Answering track of TREC a measure of precision is therefore used that takes this into account, viz. the Mean Reciprocal Rank (MRR). The Rank of a given result is the position in which the first correct answer is found in the output list of the system. Over a given set of answers the MRR is computed as the mean of the reciprocals of the ranks for all the answers.

The problem with this approach is that the underlying assumption, that an answer returned by an AE system is either completely correct or completely wrong, is not entirely realistic. Quite often we get a series of answers

[5] for the latter see:
http://www.isi.edu/natural-language/projects/webclopedia/controv-trec10-eval.html

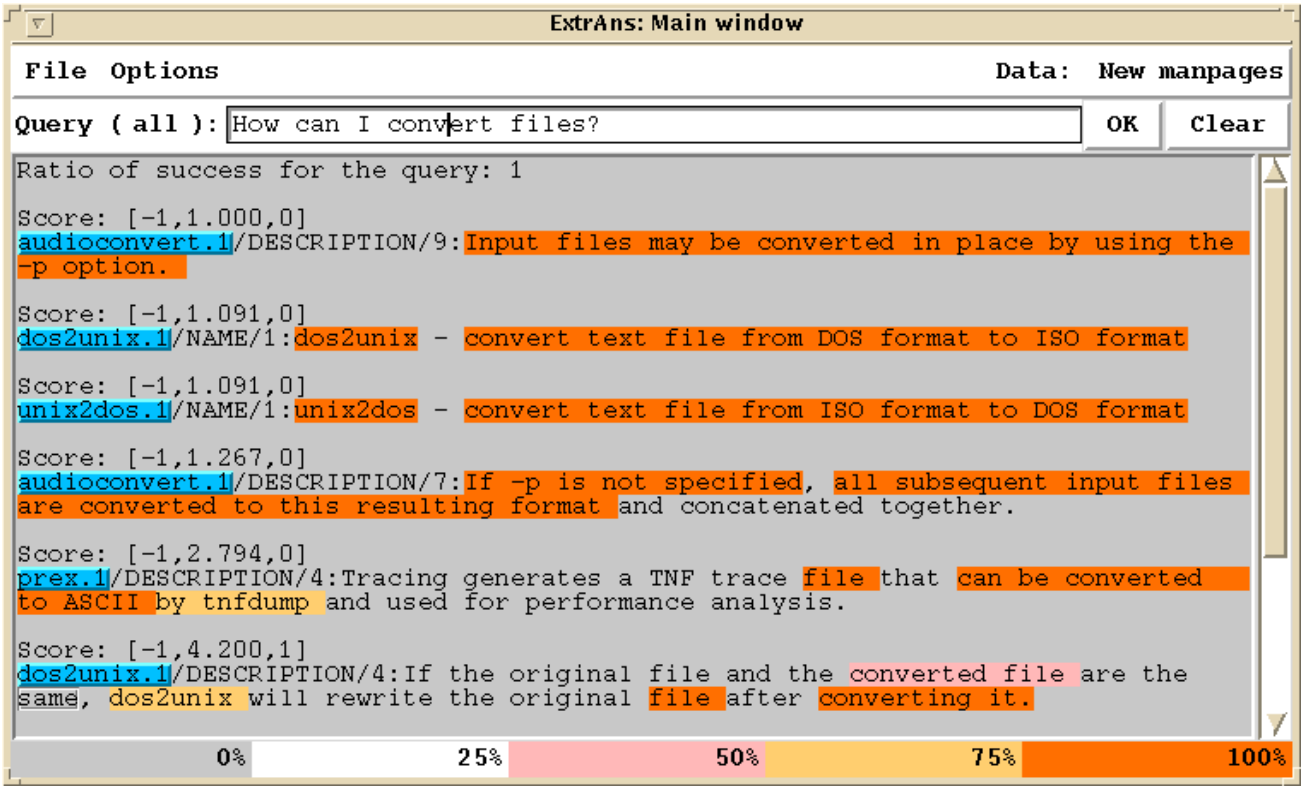


Figure 2: Identifying Relevant Parts of Sentences.

which are all correct to some degree but not entirely correct. We need some kind of weighting, exactly as in document retrieval, but again on the sentence level. The way this weighting should be performed is, however, less clear. One approach might be to find a representative set of correct answers by making a person write the ideal answers to a number of questions (labour-intensive but feasible), and then to find the sentences in the documents that are "semantically close" to these ideal answers automatically.

Semantic closeness between a sentence and the ideal answer, i.e. the weight of an answer sentence, could be computed by combining the two measures that one might call *"succinctness"* and *"correctness"*. Both measures compare a potential answer sentence with the ideal answer. Succinctness and correctness are the counterparts of precision and recall, respectively, but now on the sub-sentential level. These measures can be computed by checking the overlap of words between the sentence and the ideal answer (Hirschman et al., 1999), but we suggest a more content-based approach. Our proposal is to compare not words in a sentence, but their logical forms. Of course, this comparison can be done only if it is possible to agree on how logical forms should look like, to compute them, and to perform comparisons between them. The second and third conditions can be fulfilled if the logical forms are simple conjunctions of predicates that contain some minimal semantic information. In this paper we will use a simplification of the minimal logical forms used by ExtrAns (Schwitter et al., 1999). Below are two sentences with their logical forms:

(5) *rm removes one or more files.*
`remove(x,y), rm(x), file(y)`

(6) *csplit prints the character counts for each file created, and removes any files it creates if an error occurs.*
`print(x,y), csplit(x), character-count(y),` **`remove(x,z), file(z)`**`, create(x,z), occur(e), error(e)`

As an example of how to compute succinctness and correctness, take the following question:

Which command removes files?

The ideal answer is a full sentence that contains the information given by the question and the information requested. Since *rm* is the command used to remove files, the ideal answer is:

(7) *rm removes files.*
`remove(x,y), rm(x), file(y)`

Instead of computing the overlap of *words*, succinctness and correctness of a sentence could now be determined by computing the overlap of *unifying predicates*. The overlap of the unifying predicates ("overlap" henceforth) of two sentences is the maximum set of predicates that can be used as part of the logical form in both sentences. The predicates in boldface in the two examples above indicate the overlap with the ideal answer: 3 for (5), and 2 for (6).

Correctness of a sentence with respect to an ideal answer (recall on the predicate level) is the ratio between the

overlap and the number of predicates in the ideal answer. In the examples above, correctness is 3/3=1 for (5) and 2/3=0.66 for (6). This means that (5) is completely correct in that it returns all the relevant predicates while (6) is only partially correct in that it describes the removal of files by a command but that this command is not the "ideal command" (the removal is, in fact, merely a side-effect of a command whose primary purpose has nothing to do with file removal).

Succinctness of a sentence with respect to an ideal answer (precision on the predicate level) is the ratio between the overlap and the total number of predicates in the sentence. Succinctness is, therefore, 3/3=1 for (5), and 2/8=0.25 for (6). This means that (5) returns only relevant predicates while (6) contains some extraneous material.

Finally, a combined measure of succinctness and correctness could be used to determine the semantic closeness of the sentences to the ideal answer. By establishing a threshold to the semantic closeness, one can find the sentences in the documents that are listed as answers to the user's query.

The advantage of using overlap of unifying predicates against overlap of words is that the (semantically highly relevant) *relations between the words* also affect the measure for succinctness and correctness. We can see this in the following artificial example. Let us suppose that the ideal answer to a query is:

(8) *Madrid defeated Barcelona.*
`defeat(x,y), madrid(x), barcelona(y)`

The following candidate sentence produces the same predicates:

(9) *Barcelona defeated Madrid.*
`defeat(x,y),` **`madrid(y),`**
`barcelona(x)`

However, at most two predicates can be chosen at the same time (in boldface), because of the restrictions of the arguments. In the ideal answer, the first argument of "defeat" is Madrid and the second argument is Barcelona. In the candidate sentence, however, the arguments are reversed. The overlap is, therefore, 2. Succinctness and correctness are 2/3=0.66 and 2/3=0.66, respectively.

While these ideas have not been implemented yet they may be useful as a contribution to the question of how answers in AE systems should be weighted according to their quality. While the "gold standard" (the ideal answers) would have to be compiled by hand, comparisons against this standard could be done in a wholly automatic fashion.

6. Resources

Some of the resources that we used in our work are:

a The Aircraft Maintenance Manual (AMM) for the Airbus A320. The original SGML markup has been converted into XML for simpler processing (in English, 120 MB total, 45 MB excluding markup).

b The Aircraft Troubleshooting Manual (ATM) for the Airbus A320. Original SGML converted into XML (in English, 62 MB total).

c The on-line manual of Unix (Solaris) in English.

d A list of 524 real user questions about Unix.

e A terminology database (semi-automatically extracted) for the aircraft manuals (approx. 3000 terms).

f Terminology Visualization Tools.
Additional XML markup that denotes the extracted terms is automatically inserted into the manual. The new markup tags can be tied to presentational information (given e.g. by CSS stylesheets), so that when the manual is browsed the terms are highlighted and differentiated from the rest of the text. Most modern web browsers are capable of handling such specification of the information.

Of these resources all the manuals are copyrighted but the lists (questions, terms) are not.

7. References

Steven Abney, Michael Collins, and Amit Singhal. 2000. Answer Extraction. In Sergei Nirenburg, editor, *6th Applied Natural Language Processing Conference*, pages 296–301, Seattle, WA.

Eric Breck, John Burger, Lisa Ferro, Warren Greiff, Marc Light, Inderjeet Mani, and Jason Rennie. 2001. Another system called qanda. In *(Voorhees and Harman, 2001)*.

Eric Brill and Philip Resnik. 1994. A rule-based approach to prepositional phrase attachment disambiguation. In *Proceedings of COLING*, volume 2, pages 998–1004, Kyoto, Japan.

John Burger, Claire Cardie, Vinay Chaudhri, Robert Gaizauskas, Sanda Harabagiu, David Israel, Christian Jacquemin, Chin-Yew Lin, Steve Maiorano, George Miller, Dan Moldovan, Bill Ogden, John Prager, Ellen Rilo, Amit Singhal, Rohini Shrihari, Tomek Strzalkowski, Ellen Voorhees, and Ralph Weischedel. 2001. Issues, Tasks and Program Structures to Roadmap Research in Question & Answering (Q&A). http://www-nlpir.nist.gov/projects/pub/roadmapping.html.

C.L.A. Clarke, G.V. Cormack, D.I.E. Kisman, and T.R. Lynam. 2001. Question Answering by Passage Selection (MultiText experiments for TREC-9). In *(Voorhees and Harman, 2001)*.

Olivier Ferrett, Brigitte Grau, Martine Hurault-Plantet, and Gabriel Illouz. 2001. Qualc - the question-answering system of limsi-cnrs. In *(Voorhees and Harman, 2001)*.

Ulf Hermjakob. 2001. Parsing and Question Classification for Question Answering. In *ACL'01 workshop "Open-Domain Question Answering"*, pages 17–22.

Lynette Hirschman, Marc Light, Eric Breck, and John D. Burger. 1999. Deep Red: A reading comprehension system. In *Proceedings of ACL'99*, University of Maryland.

Eduard Hovy, Laurie Gerber, Ulf Hermjakob, Michael Junk, and Chin-Yew Lin. 2001. Question answering in webclopedia. In *(Voorhees and Harman, 2001)*.

Kevin Humphreys, Robert Gaizauskas, Mark Hepple, and Mark Sanderson. 2001. University of Sheffield TREC-8 Q&A System. In *(Voorhees and Harman, 2000)*.

Shalom Lappin and Herbert J. Leass. 1994. An algorithm for pronominal anaphora resolution. *Computational Linguistics*, 20(4):535–561.

Diego Mollá, Gerold Schneider, Rolf Schwitter, and Michael Hess. 2000. Answer extraction using a dependency grammar. *Traitement Automatique de Langues (T.A.L.), Special Issue on Dependency Grammar*, 41(1):127–156.

Marius Pasca and Sanda Harabagiu. 2001. Answer mining from on-line documents. In *ACL'01 workshop "Open-Domain Question Answering"*, pages 38–45.

Fabio Rinaldi, Michael Hess, Diego Mollá, Rolf Schwitter, James Dowdall, Gerold Schneider, and Rachel Fournier. 2002. Answer extraction in technical domains. In *Proceedings of the CICLING02 Conference*, pages 360–369, February.

Gerold Schneider, Diego Mollá Aliod, and Michael Hess. 1999. Inkrementelle minimale logische formen für die antwortextraktion. In *Proceedings of 4th Linguistic Colloquium*, University of Mainz, September 7-10. FASK.

Rolf Schwitter, Diego Mollá, and Michael Hess. 1999. Extrans - Answer Extraction from Technical Documents by Minimal Logical Forms and Selective Highlighting. In *Proceedings of the Third International Tbilisi Symposium on Language, Logic and Computation*, Batumi, Georgia.

Daniel D. Sleator and Davy Temperley. 1993. Parsing english with a link grammar. In *Proceedings of the Third International Workshop on Parsing Technologies*, pages 227–292.

M. M. Soubbotin and S. M. Soubbotin. 2001. Patterns of Potential Answer Expressions as Clues to the Right Answers. To appear in Proceedings of TREC-10.

Ellen M. Voorhees and Donna Harman, editors. 2000. *Eighth Text REtrieval Conference (TREC-8)*. NIST.

Ellen M. Voorhees and Donna Harman, editors. 2001. *Ninth Text REtrieval Conference (TREC-9)*, Gaithersburg, Maryland, November 13-16.

Ellen M. Voorhees. 2000. The TREC-8 Question Answering Track Report. In *(Voorhees and Harman, 2000)*.

Ellen M. Voorhees. 2001. Overview of the TREC 2001 Question Answering Track. To appear in Proceedings of TREC-10.

Question Answering in the Infosphere: Semantic Interoperability and Lexicon Development

Steven Lulich*, Paul Thompson†

* Program in Linguistics
& Cognitive Science
Dartmouth College
Hanover, NH 03755
steven.m.lulich@dartmouth.edu

† Institute for Security Technology Studies
Dartmouth College
45 Lyme Road, Suite 200
Hanover, NH 03755, U.S.A
Paul.Thompson@dartmouth.edu

Abstract

Much recent question answering research has focussed on supporting the textual retrieval needs of intelligence analysts. Question answering may also play a role in other less textual domains, such as sensor networks, or the Joint Battlespace Infosphere (JBI). We propose a connectivistic database to serve as the core of a lexicon which may be used to improve current methods of question answering, as well as other natural language and ontology processing application

1. Introduction

The question answering vision (Carbonell et al., 2000) and roadmap (Burger et al., 2000) documents describe a five year program for research and development for question answering systems with a focus on how such systems could support the needs of an intelligence analyst. DARPA's Office of Information Exploitation (IXO) program has the mission to ". . . develop sensor and information systems with application to battle space awareness, targeting, command and control, and the supporting infrastructure required to address land-based threats in a dynamic, closed-loop process." IXO is developing 1-, 5-, and 20-year vision statements to meet the challenges of these systems. These dynamic information environments require intelligent middleware to broker services to connect information users and sources. For example, users pose natural language questions, which must be translated into the query languages and ontologies of the heterogeneous systems making up the JBI (United States Air Force Scientific Advisory Board, 1999, 2000; Infospherics, 2001). While technologies in this area will build on current DARPA programs providing tools for efficient human creation of ontologies (DARPA Agent Markup Language, 2002; DARPA Rapid Knowledge Formation, 2002), because of the dynamic, rapidly changing environment represented by the JBI, it is necessary that more automated approaches to semantic interoperability be developed, as well.

We suggest the desirability of a connectivistic database to serve as the core of a lexicon which may be used to improve current methods of question answering, as well as other natural language and ontology processing applications. Specifically, we illustrate the use of such a lexicon in the Joint Battlespace Infosphere (JBI). Related work has been done on statistical tools that automate the process of mapping from one ontology or grammar to another (Thompson, 2001). We are interested in building on this work, as well as using mixed-initiative approaches (Haller et al., 1999) to provide human input, where needed.

2. Lexicon Development: Application of Linguistic Knowledge to Natural Language Processing

2.1. Properties of natural language which may be mimicked computationally

Three aspects of natural language are submitted for consideration:

- Grammars consist of categories which may be cognitively manipulated synchronically or altered diachronically (Heine, 1997), such as phones, morphs, words, and grammatical classes. The categories within grammars are defined with respect to each other, much as the words of a dictionary are defined with respect to other words in the dictionary, and do not therefore line up evenly across languages (Whaley, 1997). For instance, the study of languages as diverse as English, Tagalog, Manchu, and !Xhosa has resulted in the understanding that lexical classes in different languages do not all conform to the same mould. Some languages employ lexical classes which are not employed in English, and vice versa. Furthermore, the same class in different languages may not be easily reconciled with each other, and the distinctions between classes, even noun and verb, can sometimes become blurred. Morphologists and psycholinguists such as Joan Bybee (1988) and Ardi Roelofs (1992),

to name only two, have explored the idea of a connectivistic lexicon with some success, both conceptually and experimentally.

? Grammars do not consist only of minimal units and rules for combining them. It has been found that the human brain stores a far more redundant amount of linguistic information than had previously been thought. Work with aphasic patients shows that the use of rules in combining morphemes may be thought of as a back-up method for producing morphologically complex words when access to the lexicon fails (Badecker & Caramazza, 1998). Psycholinguistic experiments have shown that the timing of lexical access for morphologically simple words is not significantly different from the timing of lexical access for morphologically complex words, and phonetic and psycholinguistic studies indicate that some prosodic structures are stored as whole units alongside of the individual segments of which they are comprised (Levelt, 1999; Grzegorz Dogil, personal communication).

? Grammars are learned best by immature brains – brains with degraded short term memory – which may learn only general principles of grammar before narrowing down to specific principles (Deacon, 1997). Deacon outlines work done by others in cognitive and computer science which involved training of neural networks to learn a grammar to a relatively large degree of accuracy when the "short-term memory" of the network was disturbed. Studies by MacWhinney (1978) and Peters (1983) indicate that generalizations (rules) gradually emerge from stored rote forms, which are initially processed and stored as unanalyzed wholes, cf. (Bybee, 1988). These studies corroborate both the work done by Deacon, and the evidence that linguistic data stored in the lexicon is often redundant.

2.2. Proposal for the design of a lexicon which mimics these properties

A lexicon with five main components may serve to mimic these properties of natural language: a Pattern Finding Engine (PFE), Short Term Memory (STM), Long-Term Memory (LTM), Connectivistic Database (CD), and an Anchor Set (AS),

2.2.1. Pattern Finding Engine and Memory

The Pattern Finding Engine (PFE) searches a text for patterns, and, during the training phase of the lexicon, stores those patterns in the Short-Term Memory (STM), while the strings predictable from those patterns are stored in the Connectivistic Database. For instance, starting from scratch, the PFE recognizes a sentence such as "Johnny ate the apple" as a single unit. This imitates the theory derived from the work of Deacon, MacWhinney, and Peters above. This single unit is stored as a whole in the CD as an object of class "lexical unit." Exposure to more sentences, such as "Johnny ran away" and "The apple is red", enables the PFE to recognize "Johnny" and "the apple" as units, and to store them in the CD, along with "ran away" and "is red". Further exposure to sentences such as "Apples taste good" and "Jack and Jill ran up the hill" allows the PFE to recognize "ran" as separate from "away again", and "s" as a morpheme attached to "apple".

Initially, PFE is not better than chance at finding correct patterns. Therefore, potential patterns are stored in STM. As more and more occurrences of patterns in STM are found by PFE, the patterns in STM are stored in Long-Term Memory (LTM). Because some units larger than the segment or the word may occur with great frequency, the work of PFE together with STM and LTM allows an imitation of the theory that the lexicon is not redundancy free. This also allows us to capture idioms as whole chunks (Nunberg et al, 1994).

2.2.2. Connectivistic Database

An object of class "lexical unit" represents all of the information concerning a single unit. Within the object of class "lexical unit" is a set of objects of class "link". Each object of class "link" contains two variables: a pointer, pointing to one other object of class "lexical unit"; and a value corresponding to the strength of that connection. Each "lexical unit" also contains an activation value, which records and keeps track of the activation of that unit at all times. Activation is a measure of the probability that a certain unit will be the next one chosen out of the lexicon, and is determined by the amount of activation flowing to it through its connections with other activated units. Each "lexical unit" also has an abstract position variable, represented by an n-dimensional vector, which identifies a location for the "lexical unit" in an abstract n-dimensional Minkowsky space.

Throughout the training phase, with the help of PFE, STM, and LTM, the CD automatically organizes itself into an n-dimensional Minkowsky space. Categories are automatically approximated by defining opposing categories with respect to each other along a similar dimension. Sets of categories which are not defined with respect to each other are defined along different dimensions. Such definitions may be approximated without prior human or machine coding (Klein, 1998; Levine et al., 2001).

2.2.3. Anchor Set

Initial training of the lexicon is supervised by a human assigning certain "lexical units" to corresponding absolute concepts. Such "anchor points" provide the basis for translation from one grammar or ontology to another via the lexicon. English "chair" and German "Stuhl", for instance, refer to roughly the same concept. Therefore, the word "chair" in an English trained lexicon, and the word "Stuhl" in a German trained lexicon will both be anchored to the concept of "CHAIR". The Anchor Set (AS) can be used then to manipulate and align the abstract n-dimensional vector spaces of the two lexicons such that, by extrapolation, lexical units with nearly identical position vectors should theoretically be nearly identical in meaning or use, depending on the dimension. The more anchor points that are explicitly taught to the AS, the more accurate this alignment will be.

2.3. Discussion

To the best of our knowledge, though the ideas and evidence outlined in this paper in favor of a connectivistic view of the lexicon have been explored by linguists already, there has been no attempt to apply such a model to challenges in natural language processing. Certainly this may partially be attributed to the fact that the computing power necessary to undertake such a task has not long been available.

We believe that development of such a lexicon is relevant to Question Answering technology in several ways. First, the lexicon, whatever shape it may take, is an important and central part of any natural language processing application. Without it, language is simply noise. We believe therefore that the form of the lexicon has a direct effect on the overall performance of the application. Second, in answering a single question, it is often necessary to extract information from multiple sources of varying media and ontologies. The information coming from these disparate sources must somehow be fused together and outputted into yet another ontology or medium. Because this conception of a lexicon is easily trained, it is easily transportable across multiple domains and ontologies or grammars. As discussed in section 2.2.3, the Anchor Set allows translation from one ontology to another via the lexicon, thus enabling this kind of fusion of information. Finally, though certainly not exhaustively, the automatic categorization of words along different dimensions, and the connections between words may be helpful as a tool for word sense disambiguation.

3. Questions in the Infosphere

3.1. Background

Question answering in heterogeneous sensor networks involves some of the same issues as question answering in more textual domains, but also introduces other aspects. The answer to the question may not exist in the network at the time the question is asked. Sensors may need to be tasked to provide the answer. A mapping must be made between the language of the user and the descriptions of the functionalities of various sensors. There is high transaction volume in the Joint Battlespace Infosphere (JBI) and questions may overlap in various ways. Efficient question answering calls for query planning and optimization along the lines of work done in relational databases (Jarke & Koch, 1984) and knowledge bases, but with additional factors introduced by the distributed, mobile, highly dynamic nature of sensor networks. Also, because much of the data in these networks will be structured, question answering in this environment can also build on research on natural language interfaces to relational databases (Adroutsopoulos et al., 1995; Urro & Winiwarter, 2001).

The JBI consists of client users, databases, sensors, and filtering or fusion operations. These filtering or fusion operations are carried out by fuselets, lightweight data fusion elements. Fuselets use simple logical rules to take inputs from other elements of the JBI, such as sensors, or other fuselets, to derive fused information. The functionality of each fuselet is described using a Fuselet Markup Language (FML). The JBI is implemented as a publish and subscribe architecture, where each fuselet publishes its services and subscribes to the outputs of other elements of the JBI. Questions in the JBI are answered by breaking the question into components and efficiently routing the components through the JBI network of fuselets, databases, and sensors.

Although ontologies may be provided for various sub-domains, it may be necessary to rapidly create and map among ontologies on the fly. For example, a fuel truck may be represented in separate ontologies for target tracking and for logistics. It must be possible to: a) determine that the two representations are of the same type of entity, b) reason within the joint probability space represented by the two ontologies, and c) answer questions by fusing information from the two domains. We will investigate a variety of tools to achieve semantic interoperability. In addition to the linguistic approaches to lexicon development discussed in section 2, we plan to explore statistical, text-based mapping and subsumption tools (Woods, 1997; Buckland et al., 1999; Gey et al., 2001; Schatz, 2002).

3.2. A JBI Fuselet Example

As a simplified example of question answering in the Infosphere, consider the following. In a battlefield situation when an enemy target is to be fired upon, it is first necessary to ascertain that no friendly assets are in the vicinity that might be adversely affected. A subset of the JBI involving a network of sensors, radio transmitters operated by groups of soldiers, advanced Land Warrior personal GPS systems, current roster information, other sources of information, and fuselets would be needed to make this determination. The current location, velocity, and vector of all friendly assets would need to be determined. If processing this information takes too much time, the target opportunity might be missed. If the enemy target is fired upon without the information being processed accurately, friendly assets may become casualties. Personnel in the tactical operation center would submit a natural language query, "Are any friendly assets in danger of being hit, if the target at UTM grid coordinate XY123456 is fired upon?" This query would then be interpreted by the question answering system. Fuselet 1 would aggregate the outputs from the soldiers' radio transmitters. Fuselet 2 would aggregate the output of the GPS systems. Fuselet 3, with situational tracking software, would fuse the outputs of Fuselets 1 and 2. Fuselet 4 in the personnel services center would fuse outputs from databases with current roster information, as well as with outputs from other databases making adjustments to the current roster, e.g., lists of soldiers on medical leave. Fuselet 5 would fuse the outputs of Fuselets 3 and 4 and produce as output a report for the tactical operations center, answering the query.

4. Conclusions

We intend to address question answering issues in the JBI, in particular those concerning closed-loop sensor networks. Our domain has some overlap with that of the intelligence analyst described in the question answering vision and roadmap documents, but has significant differences, as well. We intend to build a sensor network integrated with textual messages. We will make use of

ontologies, such as a sensor markup language, but we will also explore connectivistic lexicon, corpora linguistic, and other techniques to learn about our domains in a more dynamic manner, as necessary.

5. References

Androutsopoulos, I.; Ritchie, G.D.; & Thanisch, P. (1995). Natural language interfaces to databases – An introduction. Journal of Natural Language Engineering, 1(1), p.29-81

Badecker, W. & Caramazza, A. (1998). Morphology and Aphasia. In A. Spencer, & A.M. Zwicky (Eds.), The Handbook of Morphology (pp. 390-405). Oxford: Blackwell Publishers Ltd.

Buckland, M., Chen, A., Chen, H., Gey, F., Kim, Y., Lam, B., Larson, R., Norgard, B., & Purat, Y. (1999). Mapping Entry Vocabulary to Unfamiliar Metadata Vocabularies. D-Lib Magazine, 5(1).

Burger, J.; Cardie, C.; Chaudhri, V.; Gaizauskas, R.; Harabagiu, S.; Israel, D.; Jacquemin, C.; Lin, C..; Maiorano, S.; Miller, G.; Moldovan, D.; Ogden, B.; Prager, J.; Riloff, E.; Singhal, A.; Shrihari, R.; Strzalkowski, T.; Voorhees, E.; Weischedel, R. (2000). Issues, tasks and program structures to roadmap research in question & answering (q&a). Gaithersburg: National Institute of Standards and Technology.

Bybee, J. (1988). Morphology and Lexical Organization. In M. Hammond & M. Noonan (Eds.), Theoretical Morphology: Approaches in Modern Linguistics (pp. 119-142). San Diego, CA: Academic Press.

Carbonell, J.; Harman, D.; Hovy, E.; Maiorano, S.; Prange, J.; & Sparck Jones, K. (2000). Vision statement to guide research in question & answering (Q&A) and text summarization. Final version 1. Gaithersburg : National Institute of Standards and Technology.

DARPA Agent Markup Language (DAML) (2002). http://dtsn.darpa.mil/ixo/daml%2Easp.

DARPA Rapid Knowledge Formation (RKF). (2002). http://dtsn.darpa.mil/ixo/rkf%2Easp.

Deacon, T. (1997). The Symbolic Species: The Co-evolution of Language and the Brain. New York: W.W. Norton.

Gey, F.; Buckland, M.; Chen, A.; & Larson, R. (2001). Entry vocabulary – a technology to enhance digital search. In Proceedings of HLT 2001: First International Conference on Human Language Technology Research (pp. 91-95). San Francisco: Morgan Kaufmann.

Haller, S.; McRoy, S.; and Kobsa, A. (Eds.). (1999). Computational Models of Mixed-Initiative Interaction Boston: Kluwer.

Heine, B. (1997). Cognitive Foundations of Grammar. New York: Oxford University Press.

Infospherics: Science for Building Large-scale Global Information Systems. (2001). http://actcomm.dartmouth.edu/infospherics/

Jarke, M. & Koch, J. (1984). Query optimization in database systems. Computing Surveys, 16(2), 111--152.

Klein, A. (1998). Textual Analysis Without Coding: It Can be Done. Dissertation, Mathematical Social Sciences, Dartmouth College.

Levelt, W.J.M. (1999). Producing spoken language: a blueprint of the speaker. In P. Hagoort & C.M. Brown (Eds.) The neurocognition of language (pp. 94-122), Oxford: Oxford University Press.

Levine, J.H.; Klein, A.; & Mathews, J. (2001). Data Without Variables. Journal of Mathematical Sociology, 23(3), 225--273.

MacWhinney, B. (1978). The Acquisition of Morphophonology. Child Development Publication, Chicago: University of Chicago Press.

Nunberg, G; Sag, I.; & Wasow, T. (1994). Idioms. Language, 70, 491--538.

Peters, A.M. (1983). The Units of Language Acquisition. Cambridge, U.K.: Cambridge University Press.

Roelofs, A. (1992). A spreading-activation theory of lemma retrieval in speaking. Cognition, 42, 107--142.

Schatz, B.R. (2002). The Interspace: Concept navigation across distributed communities. IEEE Computer. 35(1), 54--62.

Thompson, P. (2001). Classification Crosswalks: From Interchange to Interoperability. Classification Crosswalks: Bringing Communities Together The 4th NKOS Workshop at ACM-IEEE Joint Conference on Digital Libraries (JCDL).

United States Air Force Scientific Advisory Board. (2000). Report on Building the Joint Battlespace Infosphere, vol. 1 Summary SAB-TR-99-02.

United States Air Force Scientific Advisory Board. (1999). Report on Building the Joint Battlespace Infosphere, vol. 2 Interactive Information Technologies SAB-TR-99-02.

Urro, R. & Winiwarter, W. (2001). Specifying Ontologies – Linguistic Aspects in Problem-Driven Knowledge Engineering. In Proceedings of the 2nd International Conference on Web Information Systems Engineering, Los Alamitos, IEEE Computer Society Press.

Whaley, L.J. (1997). Introduction to Typology: The Unity and Diversity of Language. Thousand Oaks, CA: Sage Publications.

Woods, W.A. (1997). Conceptual indexing: A better way to organize knowledge. Sun Microsystems Research Technical Report TR-97-61.

Multiple-perspective and Temporal Question Answering

James Pustejovsky
Dept. of Computer Science
415 South Street
Brandeis University
Waltham, MA. 02254
jamesp@cs.brandeis.edu
www.cs.brandeis.edu/~jamesp

Janyce Wiebe
Dept. of Computer Science
University of Pittsburgh
Pittsburgh, PA 15260
wiebe@cs.pitt.edu
www.cs.pitt.edu/~wiebe

Mark Maybury
Information Technology Division
The MITRE Corporation
202 Burlington Road
Bedford, MA 01730
maybury@mitre.org
nrrc.mitre.org

Keywords: question answering systems, multiperspectives, temporal expressions, events

1.0 Introduction

The question answering vision (Carbonell et al. 2000) and roadmap (Burger et al.,2002) articulate a research and development direction for the next five years. Although a range of question and answer types are described, the ability to interpret a question and provide an answer with respect to different perspectives and the ability to answer questions involving temporal dimensions are largely unaddressed. This position paper argues for the importance of multiple perspective and temporal question answering and attempts to outline some aspects of the problem that would be important to capture on the Q&A roadmap. We address these problems in the context of two ARDA Northeast Regional Research Center (NRRC) Workshops, held in the summer of 2002, focused on time and multiple perspectives (nrrc.mitre.org).

2.0 Multiple-Perspective Question Answering

2.1 Multiple-Perspective Questions

A question may explicitly request multiple perspectives, for example ``What are the positions of German political parties on UN resolution 53?" or "What opinions are being expressed in the world press about US plans to invade Afghanistan?" In addition, questions asking for speculations or opinions might most appropriately be interpreted as asking for answers from multiple perspectives. Examples are ``How should the US response be to the terrorist incident?'' and "Will the US economy improve in the next six months?" Even for other questions, a multiple-perspective treatment may be very useful to an analyst or consumer. The user could be given the option to ask for multiple perspectives, whatever the specific form of the question. Finally, questions themselves can signal the perspective of the source or speaker (who could hold distinct views) while at the same time eliciting a multiperspective response as in "What do the Europeans think about the short-sighted US policy in the Middle East?".

2.2 Multiple-Perspective Answers

Perhaps the most obvious situation in which a question may be answered differently from multiple perspectives is when people or groups hold different beliefs about what is factually true. However, answers from different perspectives also include ideological beliefs, religious beliefs, evaluations, judgments, and speculations. They might reflect personally held beliefs, or official positions in legal, political, religious, or ideological platforms. In addition, the source of the belief might be a specific person, a group, a political or economic sector, or even the general culture at large. Recognizing the type of perspective reflected in an answer is essential for knowing how to interpret the information and what we can learn about the source.

We can envision a system that does not provide a single answer but rather presents the various positions on a topic currently being expressed in the world press, to help the user answer the question for himself or herself.

For the results to be useful, they should be characterized and clustered for presentation to the user. Storing the results in a knowledge base would support reasoning about multiple

perspectives on a topic, and detecting changes in perspective and trends over time.

Thus, five main aspects of the problem are the following:

- ? Retrieval of text segments containing candidate answers from multiple perspectives (Wiebe 1994, Wiebe et al. 1999).

- ? Characterization of the type of perspective of each answer. The answer may be presented as factual in the original source, or as a belief or opinion. It might reflect personally held beliefs, or official positions in legal, political, religious, or ideological platforms. It might be positive or negative evaluative, or speculative.

- ? Characterization of the source of the perspective. The source of the perspective may be an individual, a group, a political or economic sector, etc. Because beliefs about beliefs about beliefs, etc., may be presented, a structured representation of sources is needed.

- ? Comparison and clustering of the answers into similar perspectives, for presentation to the user.

- ? Representation of the answers in a knowledge base. As questions are answered from multiple perspectives over time, storing the results in a knowledge base would support queries such as which sources have expressed negative evaluations toward various topics, or which perspectives have changed over time.

Following are examples of multiple perspectives expressed in text. First, here are different views expressed about the same topic in editorials.

"General Musharraf has wisely chosen to throw in his lot with the US." (from *The India Times*).

"Looking at the event from the beginning most people including myself were convinced that President Musharraf's decision to support the USA was ill-thought, ill advised and was only taken for financial reward in a hurry." (from *The Frontier Post, Pakistan*)

In the following passage, which describes a factual dispute, the sources of the perspectives are people mentioned in the text:

"Agha [Tayab Agha, spokesman for Taliban leader Mohammad Omar] claimed the Taliban continued to rule in Kandahar, Oruzgan, Zabol, Ghazni and Helmand provinces. Afghan and Western sources, along with travelers who arrived today in Spin Boldak, disputed his claim, saying the Taliban only control parts of most of these provinces and had no influence over Ghazni at all (from *The Washington Post Foreign Service*).

A rich representation is needed to capture the characteristics of perspectives, their sources, and their objects, which may themselves be perspectives.

In addition to involving answers from multiple perspectives, questions often refer explicitly to time sensitive information, the area of question answering which we consider next.

3.0 Temporal Question-Answering: When time makes a difference

Humans live in a dynamic world, where actions bring about consequences, and the facts and properties associated with entities change over time. For this reason, temporally grounded events are the very foundation from which we reason about how the world changes. To be sure, named entity recognition is crucial to analyst reporting, information extraction, and question-answering systems; but without a robust ability to identify and extract events and time-stamps from a text, the real "aboutness" of the article can be missed. Moreover, entities and their entities change over time as well; hence a database of assertions about entities will be incomplete or incorrect if it doesn't reflect such time-stamps (e.g., the status of the World Trade Center Buildings before and after Sept. 11, 2001). To this end, event recognition drives basic inferences from text.

The focus of the Time and Event Recognition for Question Answering (TERQAS) workshop (time2002.org) is to address the problem of how to answer temporally-based questions about the events and entities in news articles. Currently,

questions such as those shown below are not generally supported by Q&A systems:

1. Is Gates currently CEO of Microsoft? (*time-stamp* question)
2. When does the seminar take place? (*punctual event* question)
3. How long did the hostage situation in Berlin last? (*Duration of event* question)
4. On what days were there bombings in the Middle East? (*Quantified event* question)
5. What airplane crashes occurred shortly after assassinations? (*Quantified event* question with *relative event ordering*)
6. What terrorist actions occurred within a week of political speeches by extremist governments? (*Quantified event* question with *relative event ordering*)
7. What bombings have occurred during the occupation of the West Bank? (*Quantified event* question with *durative event overlapping*)

What characterizes these questions as beyond the scope of current systems is the following: they refer, respectively to the temporal properties of the entities being questioned, the relative ordering of events in the world, and events that are mentioned in news articles, but which have not occurred at all.

3.1 Temporal Question-Answering Challenges

There has recently been a renewed interest in temporal and event-based reasoning in language and text, particularly as applied to information extraction and reasoning tasks (cf. Pustejovsky and Busa 1995; Mani and Wilson 2000; 2001 ACL Workshop on Spatial and Temporal Reasoning). Several papers from the workshop point to promising directions for time representation and identification (cf. Setzer and Gaisauskas, 2001, Filatova and Hovy, 2001, Schilder and Habel, 2001). Many issues relating to temporal and event identification remain unresolved. In our efforts we aim to (a) to examine how to formally distinguish events and their temporal anchoring in text (news articles); and (b) to evaluate and develop algorithms for identifying and extracting events and temporal expressions from texts.

Relative to the first goal above, we are addressing four basic research problems:

1. Time stamping events (identifying an event and anchoring it in time)
2. Ordering events with respect to each other (relating more than one event in terms of precedence, overlap, and inclusion)
3. Reasoning about the ramifications of an event (what is changed by virtue of an event)
4. Reasoning about the persistence of an event (how long does an event or the outcome of an event persist)

3.2 TimeML and TIMEBANK

To answer these problems, we are presently working to define a specification language and an annotated Gold Standard corpus. A specification language, TimeML, will be defined and developed. This XML-compliant language should formally model most of the following properties of time and events:

1. How to represent the interval values of events (time-stamping);
2. How to represent aspectual properties of an event (what phase of an event is being time-stamped);
3. How to represent all possible temporal ordering relations between two events;
4. How to model shallow (entailed) ramifications of an event (what related events are triggered by an event's occurrence);
5. How to model when a state persists and when it does not (what states follow from an event)

Once the initial definition and specification of TimeML is complete, it will be necessary to begin annotation on a large number of news articles, in order to create a Temporal Gold Standard (TIMEBANK). This entails the annotation of at least 400 articles, taken from four separate sources: 100 DUC articles; 100 ACE articles; 100 AP News articles; and around 100 PropBank annotated articles. We are presently in the process of the construction of TIMEBANK, the annotated corpus that we will provide as a community resource when completed, subject to appropriate copyright restrictions.

The specification language TimeML will suggest but not determine the nature of how answers to temporal questions are best presented to the user. This remains largely an issue of habitability and

usability of the application. Nevertheless, answers to temporal questions may take one of several forms:

1. Selections from database entries, populated from the appropriate information extraction algorithms;
2. Textual fragments from news articles, indicating total or partial answers to the question;
3. Answers may be abstracted and represented visually in terms of a timeline or a hyperbolic visualization algorithm.

The second goal mentioned above involves the evaluation of existing, and development of new temporal extraction algorithms. The four research problems given above correspond roughly to extraction algorithms of increasing degrees of sophistication and complexity. Time stamping events is not too dissimilar from named entity recognition; event ordering identification is somewhat similar to relational parsing; and capturing persistence and ramification properties of events is similar to identifying dependencies in a dependency grammar.

The algorithms will be applied and tested against the development corpus of the gold standard, TIMEBANK. Evaluation against a blind test set will measure for accuracy of answers for a range of questions, as defined by the participants, paying particular attention to target the specific temporal properties of the text with different questions.

Significantly, the results of our workshop will enable the community to begin addressing an entirely new type of question-answering capability, and one that is necessary for answering questions pertaining to the deeper content of news articles.

4.0 Implications for Q&A Road Map

The above observations point to the importance of research into multi-perspective and temporal Q&A. Some of the key milestones on the roadmap include:

- Characterize the types and nature of multiple perspectives and temporal aspects
- Establish and iteratively refine an ontology of multiple perspectives both for question analysis and answer generation. Do the same for temporal questions.
- Create corpora that include both multiple perspective and temporal phenomena
- Create annotation standards for multiple perspective and temporal markup

The two NRRC workshops described in this article will contribute in the next three months to advancing the state of the art by creating:

- An ontology of perspective
- An annotated corpus of multiple perspective questions and answers
- A repository of linguistic clues indicative of perspective
- A baseline of experimental results (segmentation, property annotation, clustering)
- A standard markup language for temporal and event expressions, TimeML
- A gold standard corpus for temporal expressions, TIMEBANK

5.0 Conclusion

This paper describes two important aspects of question answering that have gone largely unaddressed: time and multiple perspectives. These are important elements that should be reflected in the Q&A roadmap.

6.0 References

Allen, J. "Maintaining Knowledge about Temporal Intervals", Communications of the ACM, 26(1):"832-843.

Burger, John; Cardie, Claire; Chaudhri, Vinay; Gaizauskas, Robert; Harabagiu, Sanda; Israel, David; Jacquemin, Christian; Lin, Chin-Yew; Maiorano, Steve; Miller, George; Moldovan, Dan; Ogden, Bill; Prager, John; Riloff, Ellen; Singhal, Amit; Shrihari, Rohini; Strzalkowski, Tomek; Voorhees, Ellen; Weischedel, Ralph. (2002) “Issues, Tasks and Program Structures to Roadmap Research in Question & Answering (Q&A). www-nlpir.nist.gov/projects/duc/papers/qa.Roadmap-paper_v2.doc

Carbonell, Jaime; Harman, Donna; Hovy, Eduard; Maiorano, Steve; Prange, John; and Sparck Jones, Karen. 2000. “Vision Statement to Guide Research in Question & Answering (Q&A) and Text Summarization”. Final version 1.www-nlpir.nist.gov/projects/duc/papers/Final-Vision-Paper-v1a.pdf

Filatova, E. and E. Hovy (2001) "Assigning Time-Stamps To Event-Clauses", in Proceedings of ACL Workshop on Temporal and Spatial Information Processing, Toulouse, France, July, 2001.

Mani, I., Wilson, G., Ferro, L., and Sundheim, B. 2001. Guidelines for annotating temporal information. Proceedings of Human Language Technology Conference. hlt2001.org/papers/hlt2001-31.pdf

Mani, I. and G. Wilson (2000) Robust Temporal Processing of News", in Proceedings of the 38th Annual Meeting of the ACL, Hong Kong.

Northeast Regional Research Center (nrrc.mitre.org)

2001 ACL Workshop on Spatial and Temporal Reasoning.

Pustejovsky, J. and F. Busa (1995) A Revised Template Description for Time in MUC-6 (v3), http://www.cs.nyu.edu/cs/faculty/grishman/muc6.html.

Schilder, F. and C. Habel (2001) "From Temporal Expressions to Temporal Information: Semantic Tagging of News Messages", in Proceedings of ACL Workshop on Temporal and Spatial Information Processing, Toulouse, France, July, 2001.

Setzer, A. and R. Gaizauskas (2001) “A Pilot Study On Annotating Temporal Relations In Text ", in Proceedings of ACL Workshop on Temporal and Spatial Information Processing, Toulouse, France, July, 2001.

Wiebe, J., R. Bruce, and T. O'Hara (1999) “Development and Use of a Gold Standard Data Set for Subjectivity Classifications”, in *Proc. 37th Annual Meeting of the Assoc. for Computational Linguistics (ACL-99),*

Wiebe, J. (1994) “Tracking Point of View in Narrative”, *Computational Linguistics*, 20.2: 233-287.

Summarization-Based Japanese Question and Answering System from Newspaper Articles

Yohei Seki* and Ken'ichi Harada*

*Department of Computer Science, Keio University
Kanagawa, Japan 223-8522
yohei.seki@dream.com

Abstract

Recently, many researchers are focusing on the application of Natural Language Processing (NLP) techniques such as summarization, information extraction, and text mining. One of the challenges with these technologies is developing an accurate Question and Answering System (Burger et al., 2001). In this paper, we will discuss Japanese Q&A problematic issues that have appeared in our experimental system. Our system is implemented with multi-document summarization (MDS) techniques.

Keywords: Japanese Q&A System, multi-document summarization technique, information fusion from multiple newspaper articles, and QAC (Question and Answering Challenge)

1. Introduction

There is a year long workshop being held by the National Institute of Informatics in Japan called NTCIR-3. We participated in the 'Question and Answering Challenge' (QAC) dryrun (Fukumoto and Kato, 2001) in the winter of 2001: Japanese Q&A tasks. We created an experimental system for the Japanese Q&A to detect problems specific to the Japanese language. Our input data was Mainichi Newspaper articles from 1998 and 1999 Year. This included about 230,000 articles. In this paper, we propose a multi document summarization based approach for Q&A. We also discuss some Japanese related problematic issues.

This paper consists of seven sections. We explain the tasks of QAC in Section 2, and discuss details of our system design and approach in Section 3. Section 4 provides an overview of our system user interface. Section 5 contains a brief evaluation of our system with QAC problems. In Section 6, some problematic issues are discussed. Finally, we present our conclusions in Section 7.

2. Question and Answering Tasks in QAC

The Question and Answering Challenge workshop (QAC) (Fukumoto and Kato, 2001) consisted of three tasks. The first and second task contained the same 50 questions. A list of five accurate answers was the goal in the first task; The goal of the second task was to extract the correct answer set. The third task had 10 problems and each problem had one follow-up question. The dryrun with these three tasks was held on five consecutive days in December, 2001.

The Answers were to be noun phrases which indicated a person's name, organization names, money, size, date and so on. The source documents were a two-year-period of Japanese newspaper articles.

3. Our Multi-Document Summarization Based Approach for the Q&A System

Our approach for the Q&A System consisted of three procedures: question analysis, summarization of questions from various articles, and answer formation.

3.1. Question Analysis

The Question analysis process is basically divided in two parts. One is the detection of question type, and the other is the extraction of keywords with a numeric score that summarizes documents. We use the Japanese part-of-speech tagger, 'Chasen'[1] in order to break the question sentences into morphemes. Question types are categorized with keywords as follows:

- Interrogative pronoun
  - Nan(-i) (What)
    - modifying suffix
      - Nen (Year)
      - Gatsu (Month)
      - Nichi (Day)
      - Nin (How many people)
      - Kai (How much times)
      - Ken (How many units)
  - Dare (Who)
  - Doko (Where)
    - Kuni (Which country)
    - Kaisha (Which company)
  - Itsu (When)
  - Ikura (How much)
  - Dono, Dore (Which)
    - Kikan (How long)
    - Ryou (The amount)

Figure 1: Japanese Question Taxonomy

[1] http://chasen.aist-nara.ac.jp/

The question taxonomy above shows that Japanese question types are determined by a combination of an interrogative pronoun and a modifying suffix.

Another process is keyword detecting and scoring. We score keywords in each question as follows:

1. Each matching noun morpheme receives 1 point.
2. The proper noun or phrase containing the proper noun receives 3 points.
3. A time related adverb/noun receives 0.5 points.
4. Each verb or adjective morpheme (except some basic elements) receives 1 point.

3.2. Sentence Extraction with Multi-Document Summarization Technique

Next, we extracted sentences related to each question keywords from a two year supply of newspaper articles. The question keyword scores determine these individual sentence scores.

If a sentence contains a keyword, the keyword score is added to the sentence score, then the sentence score is divided by the sum of all the keyword scores in that question. Therefore, a maximum score of a sentence is 1. If a score of any sentence is more than 0.4, the sentence is extracted and stored into the answer file for that question. This is a kind of cut and paste summarization technique (Jing and McKeown, 2000) from a wide source of newspaper articles (McKeown and Radev, 1995). In order to accelerate our system's performance, some multi-document summarization techniques (Mani, 2001) with text segmenting and clustering (Stein et al., 1999) were also needed. When this MDS approach is adopted, the Q&A accuracy performance must be kept in mind. MDS has some information fusion or aggregation processes to avoid overlapping information. If this process was applied wrongly, the correct answer would be removed from summary. We did not implement this process at this stage but implemented a similar process at the answer formation stage.

3.3. Answer Formation from Summary Sentences

Answer Formation is the process of extracting answers from summary sentences using question types. We implemented this step as pattern matching according to question type information with Perl. We use question type information like Nan-Nen Nan-Gatsu (In what year and month did the event happen?), and encode that information in regular expressions like $/(0-9)\{1,2\}gatsu(0-9)\{1,2\}nichi/$ in order to detect answer candidates.

Some question types were needed to extract distance patterns or make answers with a parsing technique. We implemented noun formation functions according to question types with a recursive function about part-of-speech information (concerned with noun morpheme type). The noun phrase formation process was different according to question types and was localized with Perl functions. Some examples are as follows:

1. Who (Dare) Questions
   'Chasen' tagged personal names as 'noun-proper noun-personal name'. When 'Chasen' tagged a personal name correctly, the personal name is extracted based on the noun formation. In addition, an abbreviated name like 'J.F.K.' or some hard to place place noun needs to be extracted with an answer formation process. This type answer was not tagged correctly with the morpheme tagger. Therefore, we need some parsing technique to look before and after the part-of-speech information.
2. When (Itsu) Questions
   'When' questions' difficulties mainly stemmed from unknown details: What year, month, day, or time? We extracted answers from 'when' questions with time-related number extraction and formation. When some time-related suffixes were matched, this pattern was formed following Japanese conventional time-expressing order; year, month, and day. When time information was expressed with 'of' or other modifying terms, there might be gaps between some time expressions. For example, 'In Keicho 5 (1600), the war of Sekigahara started on the 15th September.' The year and the date are separated in the sentence but both are necessary in an answer. If that information together was expressed in one sentence, our system would have no problem extracting the correct answer to form one time expression.
3. Where (Doko) Questions
   'Where' questions also varied in their answers according to the details. To find a specific location of an event such as a war in East Timor in Indonesia, the initial input question might not be able to place 'Daerah Istimewa Aceh' province without wider geographic information. The morpheme tagger tagged a place noun as 'noun-...-place' and a country name noun as 'noun-...-place-country'. In our system, this distinction is judged mainly based on question keyword information. When the question was judged to be concerned with country name, the corresponding function was called.
4. Amount Questions
   In the Japanese language, amount information is characterized with a modifying suffix like 'liter' or 'cubic meter'. Therefore, this suffix information is key in extracting an answer. Number information was tagged correctly as 'noun-number' or 'prefix-auxiliary-number'. Our system formed these elements to make quantity noun phrases.

Extracted answers were scored with their source sentence score and their occurring frequencies. Some answer candidates with same meanings were merged to a single answer with information fusion or aggregation techniques to avoid overlapping answers.

3.4. Detecting Answers for Follow-up Questions

In Task 3, we employed a different approach because follow-up questions often contain pronouns instead of nouns and don't contain specific keywords. To extract an answer in a follow-up question, we use a summary from the first question and the question type pattern in the follow-up question. Some follow-up question examples are shown as follows:

1. (a) What are the titles of Mr. Natsume Soseki's most famous work?
   (b) What was his eldest son's occupation? (his = Mr. Natsume Soseki)
2. (a) When did the 'Aerosmith' make their debut?
   (b) What was their first hit at that time? (at that time = their debut time)
3. (a) What are the three biggest festivals in Japan?
   (b) Where are those festivals held? (those = the three biggest)

4. System User Interface

The Q&A system produced summaries including sentence weights and source article ID numbers. They were tagged in XML-style formats. When the answer formation process was executed, answers were provided with their occurring articles by using summary information. This system is shown in Figure 2.

5. Evaluation

QAC results were evaluated with MMR (Maximal Marginal Relevance) scoring (Mani, 2001) and F-score (or F-measure) (Stein et al., 2000) metrics. Some bugs in our system were removed after the dryrun was finished. The results of our present system are shown as follows.

1. Task 1 (Top five Q&A)
   Task 1 had 50 questions. We scored the top 5 answers as follows: if the best answer was in fact correct, 1 point was added to the score; if second best answer was correct, 0.5 points was added to the score; ...; if the fifth best answer was correct, 0.2 points was added to the score. The total score ranges are shown in Table 1.

Score	Rates
$1 \leq \quad - \quad$	$\frac{13}{50}$
$0.5 \leq \quad - \quad < 1$	$\frac{7}{50}$
$0 < \quad - \quad < 0.5$	$\frac{8}{50}$
0	$\frac{22}{50}$

Table 1: Scoring in Task 1

   Answer scores with over 1 point contained four time-related questions, two questions about organization and personal names, one question about great literary and artistic works, money, people, units, and countries.

2. Task 2 (Answer Set)
   Task 2 had the same questions as Task 1. The goal of Task 2 was to extract the correct answer set. Our system answered this task as the best 10 answers. F-score ($\frac{2 \times Precision \times Recall}{Precision + Recall}$) ranges are shown in Table 2.

F-score	Rates
$0.6 < \quad - \quad \leq 1$	$\frac{2}{50}$
$0.4 < \quad - \quad \leq 0.6$	$\frac{3}{50}$
$0.2 < \quad - \quad \leq 0.4$	$\frac{9}{50}$
$0 < \quad - \quad \leq 0.2$	$\frac{19}{50}$
$\quad - \quad$	$\frac{17}{50}$

Table 2: F-score in Task 2

   Questions with the best two scores were concerned with literary and artistic works and countries. Both questions contained multiple answers.

3. Task 3 (Follow-up Q&A)
   Task 3 had 10 follow-up questions to each of the original questions. Out of the 10 questions, two questions contained correct answers in the top rank: they were a time-related question and a question about debut work. Another three questions contained correct answers in the top five ranks. Another two questions contained correct answers. The remaining three questions did not come up with a correct answer: questions concerning occupations, ranks, and personal names.

6. Some Problematic Issues

In this research, we only used surface information and didn't use deeper semantic information like a thesaurus would provide. Our result set contained erroneous elements, but in Task 2, $\frac{2}{3}$ of the correct answers were found. There are two reasons why correct answers were not found: there was too much erroneous information extracted and the correct answers were not extracted and put in the initial summary.

The source input data of QAC contained a very large (about 230,000) amount of articles. Our system caused some time-consuming problems because our system extracted summaries with common weighing values for every question type. Some questions extracted too many summary and others didn't extract enough summaries. In fact, the assigned threshold 0.4 was very sensitive according to question types. When this threshold was set as '> 0.4' (not equal), some questions contained more accurate answers in the best 10 answer candidates, but other questions' answers were missed. Although our threshold , of course, can be changed easily according to question type, some explicit criteria between threshold values and question types were hard to establish. In addition, when commonly used and polysemous question keywords were detected, many sentences with erroneous elements were extracted.

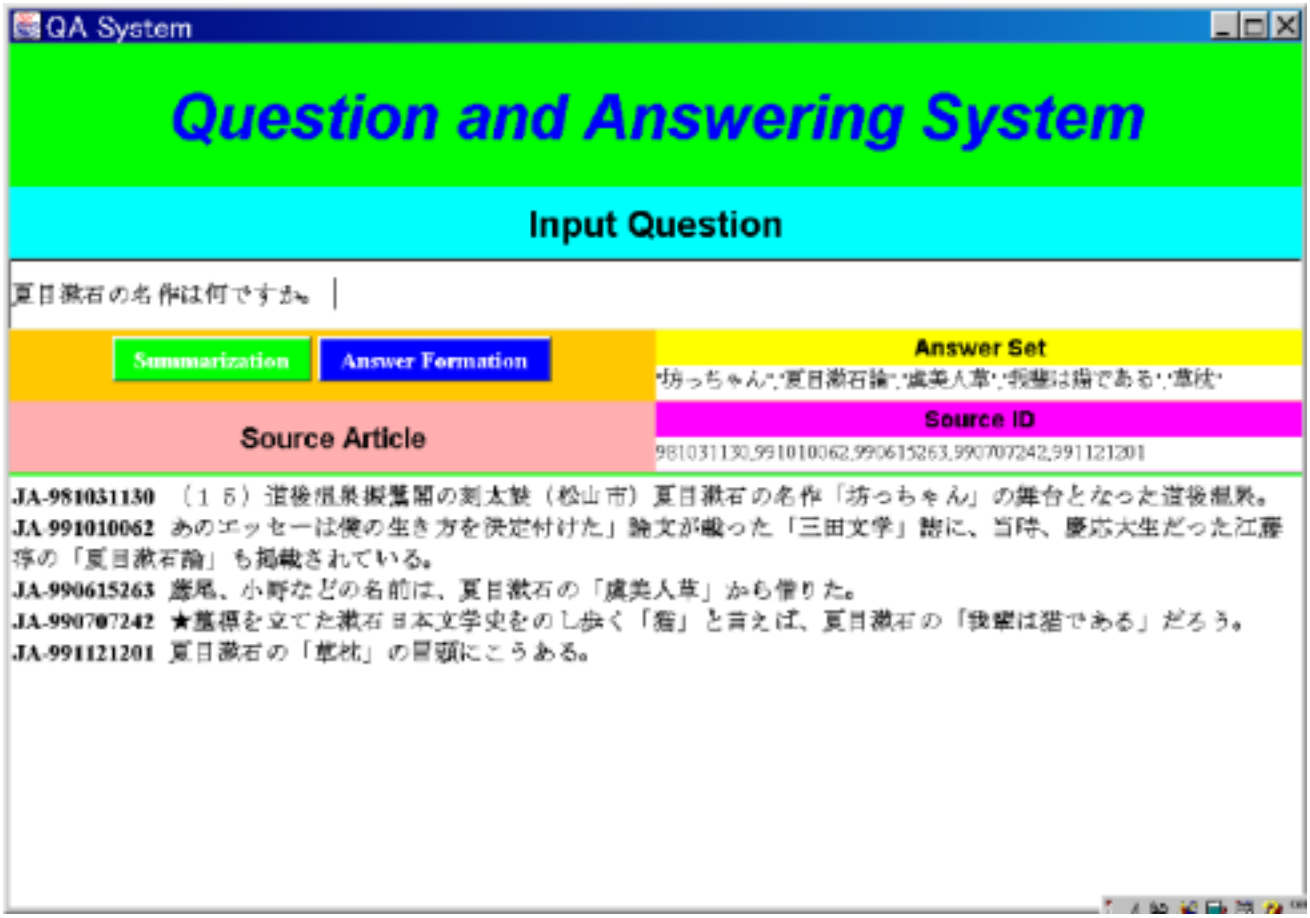


Figure 2: Q&A System

On the other hand, answer quality problems mainly stemmed from the question analysis quality. Questions which extracted too much erroneous information were mainly concerned with unique personal names or too specific place names. Other questions which did not contain correct answers were relatively unique-patterned questions. In order to increase the accuracy, we need to use a more semantic sensitive program.

We explained our improvement strategy for the Japanese Q&A problematic issues. In Japanese, there are two ways to say 'in the second place': "Dai-ni-i" and "ni-i". In the latter, the prefix "Dai" is omitted. We implemented a noun phrase formation to detect an answer with a parsing technique, but the two Japanese examples above came up with two different answers. A technique in detecting same meanings to make a single answer is also needed. This technique is a kind of multi-document summarization technique (Mani, 2001), especially for information fusion from multiple sources.

7. Conclusions and Future Direction

We tested our experimental Q&A System mainly using morpheme type information and the multi-document summarization based technique. Our results contained $\frac{2}{3}$ of the correct answers and each answer was provided with its occurring article ID number. Therefore, our system is useful for checking results with people.

In Japanese, question analysis process is a little more complex than English because question type is determined with the combination of interrogative pronoun and modifying suffix. A parsing and information fusion techniques regarding Japanese morphemes are needed in implementing the answer formation process.

In order to improve our results, some semantic information for the question category or taxonomy of inquiries (Burger et al., 2001) may be needed to reduce the amount of incorrect answers from a large summary source. In addition, if the assigned threshold for summarization is changed according to question type information, better results will follow. In order to determine precise thresholds according to question types, we will try more Q& A tasks and adjust our system.

Acknowledgements

We thank the National Institute of Informatics and members held for NTCIR-3 QAC, and also thank Robert B. Whitehead for re-reading the English.

8. References

J. Burger, C. Cardie, V. Chaudhri, R. Gaizauskas, and S. Harabagiu et al. 2001. Issues, tasks and program structures to roadmap researh in

question & answering (q & a). http://www-nlpir.nist.gov/projects/duc/roadmapping.html.

J. Fukumoto and T. Kato. 2001. An overview of question and answering challenge (qac) of the next ntcir workshop. http://www.nlp.cs.ritsumei.ac.jp/qac/qac-ntcirWS2.pdf.

H. Jing and K. McKeown. 2000. Cut and past based text summarization. In ANLP-NAACL 2000, Seattle, WA USA, May.

I. Mani. 2001. Automatic Summarization, volume 3 of Natural Language Processing. John Benjamins, Amsterdam, Philadelphia, first edition.

K. McKeown and D. R. Radev. 1995. Generating summaries of multiple news articles. In the 18th Int. ACM SIGIR Conf. on Research and Development in Information Retrieval, pages 74–82, Seattle, WA USA, July.

G. C. Stein, T. Strzalkowski, and G. B. Wise. 1999. Summarizing multiple documents using text extraction and interactive clustering. In Pacific Association for Computational Linguistics (PACLING-1999).

G. C. Stein, T. Strzalkowski, G. B. Wise, and A. Bagga. 2000. Evaluating summaries for multiple documents in an interactive environment. In 2nd Int. Conf. on Language Resources & Evaluation (LREC2000).

Question Answering system for POLISH (POLINT) and its language resources

Zygmunt Vetulani

Adam Mickiewicz University
Dept. of Computer Linguistics and Artificial Intelligence
ul. Umultowska 87, PL-61614 Poznan, Poland
http://main.amu.edu.pl/~vetulani
vetulani@amu.edu.pl

Abstract

In this paper we would like to present several issues related to our long-term research on question answering in Polish. Experiment-generated corpus of question-answer pairs, as well grammatical resources for developing Q&A systems for Polish language are presented.

1. Introduction

The research reported in this paper is a part of a long-term project aiming at the software platform with emulated linguistic competence to study man-machine interaction (Vetulani, 2000b; Vetulani & Marciniak, 2000). A question answering system constitutes an essential part of this project. The name POLINT stands for successive versions of systems derived from the Polish module to the ORBIS system (Colmerauer, Kittredge). What makes an essential difference with respect to its predecessors is that POLINT may be used as an interface in real-time systems because of substantial efficiency improvement. The recent version[1] of the system enables the user to ask questions concerning an episode of a football match (cf. Appendix 1). Two further systems are now being developed: the ACALA virtual robot controlled by the natural language interface (Vetulani & Marciniak, 2000) and a virtual "interactive glass case" for Archeological Muzeum in Poznan (MUZARP, by Vetulani and Gribko).

Our research, the substantial part of which is presented in this paper, focussed first of all on the following issues among those mentioned in the Q&A Roadmap Paper (Burger et al., 2002):

1) question taxonomies (and formal models),
2) question processing (syntax, semantics, parsing, understanding),
3) real time question answering (efficient processing),
4) interactive Q&A (dialogue structure),
5) user profiling for Q&A.

2. Empirical background: reference corpus for system design and evaluation

The development of POLINT was preceded by empirical studies on question answering in Polish. This preparatory work consisted in collection of a small but highly annotated corpus of information-acquisition-oriented question-answering dialogues. This corpus contains of 582 question-answer pairs collected during 30 sessions with human subjects. The questions were collected at sessions involving two participants: the information seeker and the information provider. The information seeker was supposed to formulate written questions to the information provider about the content of a picture (with regard to an intentionally banal subject: a scene with St. Claus, children, gifts, etc.). The information seekers were given a partial knowledge of the scene: the same picture with several blank areas. This very special setting and a particular mode of communication amounted with a number of observations, which, despite obvious limitations, are of interest especially at the early stage of QA system design. Examples of the observed syntactic phenomena of general interest are:

- short questions (average between 6 and 7 words in a question and between 2 and 3 words in a nominal group),
- rare ellipsis of whole constituents,
- low complexity of questions (small number of polypredicative questions: 35/582),
- rare use of relative clauses in questions,
- practical absence of questions with negated predicate,
- ...

Besides these purely syntactic observations, the corpus permitted preliminary studies on various discourse related phenomena such as: anaphorical links between answers and questions, long distance anaphora in dialogues, focus structure, dialogue structure and internal linking devices (anaphora, ellipsis, common-pattern-links, linking words).

Another practically useful result (used when designing POLINT) was the typology of observed syntactic structures (of course very much biased by the experiment setting, domain, mode etc.). The corpus attests mainly questions which require relatively little inferences. Most of them belong to the following categories (according to Arthur Graesser's taxonomy, cf. Burger et. al., 2002):

- verification,
- disjunctive,
- concept completion,
- feature specification,

[1] An early version of the system was tested as a front-end to the EXPÆRT system to store information retrieved from text documents about arts (Martinek&Vetulani, 1991).

- quantification,
- request/directive.

Within the typology of questions proposed in (Vetulani, 1989) these are mostly *basic questions*, in opposition to *non-basic questions* (cf. *compound questions* discussed in Belnap & Steel (1976)) rare in the kind of question-answering discourse oriented to the acquisition of factual, fine granulated information.

The St. Claus Corpus is supplied with rich annotations (only for questions). What follows is an annotated question-answer pair from this corpus.

Question: Co trzyma Mikolaj w prawej rece? /What is St. Claus holding in his right hand?
(1) $X_{subst,a}$;$V_{f,p(3)}$;N_n; $_{<w>}N_l$
(2) (?)[Arg_1: Mikolaj; Predicate: trzyma; Arg_2: ?; Arg_3: w prawej rece]
(3) [Arg_1: $_3$(N_n); Predicate: $_2$($V_{f,p(3)}$); Arg_2: $_1$($X_{subst,a}$);Arg_3: $_4$($_{<w>}N_1$)]
(4) Predicate=TRZYMAC-CZYMS(Arg_1,Arg_2,Arg_3)
Answer: Nic/Nothing

The structure (1) shows the surface linear ordering of different parameterised categories (X - interrogative phrase, N_g - noun phrase in genitive etc.); it is *called formal linear model* of the sentence. The lines (2) and (4) form the so called *predicate-argument structure* of the sentence (the line (4) describes the type of semantic requirements of the predicate). Somehow more abstract representation of the sentence is formed by (3) together with (4) (abstraction is made of surface forms, but relative position of the surface string (beginning of) is noted using the left-low numerical index, cf. for example the value 3 in $_3(N_n)$). Theoretical models for corpus annotation were introduced as an application of the unification-oriented concept of question-answer relationship (Vetulani, 1989) being inferred from the classical works by Ajdukiewicz (1965), Belnap and Steel (1976) and others.
What has appeared to be particularly useful are formal models (1) because they may be used as a skeleton of a formal grammar. Although the initial corpus is relatively small (due to time consuming and complex hand annotation procedure) it may be extended in a coherent way at any moment because the documentation of corpus generating experiment is very detailed and the collection procedure is simple (cf. Appendix 2). The corpus (now called St. Claus Corpus) and its methodology has been thoroughly described in paper publications (Vetulani,1989, 1990) and has been recently included as a basic resource in the data part of the Polish national project aiming to create NL evaluation tools for Polish (as announced at LREC1998 by Bien (1998)). Now, the St. Claus Corpus[2] is being prepared for free distribution for non-commercial purposes and will soon be available through the Internet. (This is a good reason for its presentation at the present QA Workshop.)

[2] Its substantial enlargement is being planned for the nearest future.

An important part of the above mentioned MUZARP project is based on empirical studies as well. In this project (now under development) the human user will be allowed to ask questions to virtual individuals represented in the "virtual interactive showcase". Questions will be about the "virtual showcase" world. In order to define the profile of a hypothetical user we have begun corpus collection where the (potential) human users were asked to imagine questions they *would like* to ask to the virtual scene participants *if* they were apt to do so. (The scene represents ancient country people at work.). The corpus collection is in progress and no systematic processing has started yet. It is already clear, however, that the MUZARP corpus will be substantially different from the StClaus Corpus (which is not surprising at all).

3. Generic grammatical resources of POLINT

The strength of any NL parsing (understanding, processing, etc.) system is measured by the power of its grammar and dictionary. These two modules contain the essential part of linguistic information about the language being processed but the respective role of each of them varies from case to case. In the POLINT system grammatical information is spread between rules and dictionary items, forming a lexicon-grammar. This solution will enable application of linguistically motivated heuristics to limit (at linear cost) the search during the rules-driven parsing.

3.1. Grammar rules

The POLINT grammar is composed of DCG-like rules. (It is implemented in PROLOG, but the parsing technique is much more sophisticated then the standard parsing algorithm inherent to PROLOG). That means that they have context free shape and allow arbitrary terms, including variables, as parameters. As POLINT was conceived as NL understanding system, the main goal of syntactic rules is to result in sentence segmentation useful for further (or parallel) semantic evaluation. The chosen theoretical model is the predicate-argument model, as described in (Vetulani, 1989). As a linguistic grammatical background we use the traditional phrasal approach, with some simplifications when compared to the traditional syntactic categorisations. For example we have removed some classical, very common but for us superfluous categories, as e.g. *subject phrase* or *direct/ indirect complement.* Instead, in both cases, we are using the category *noun phrase* to denote the sentence phrase which function will be specified by values of morpho-syntactic parameters (as, e.g., case: genitive). In principle, we have assumed that a sentence is composed of one or more noun phrases (arguments) and just one verbal phrase (predicate) in an order which is highly free in Polish. In practice, because of over-generation of rules, the initial (generic DCG-like) grammar has been transformed into more effective one, based on a "new" category of *sentence_segment* (*sentence_segment* is composed *of noun_phrase/verbal_phrase + sentence_segment*). This solution involving recursion will permit to control

effectively parsing by involving special control parameters (cf. Vetulani 1997) which function as heuristics calculated at the pre-analysis stage. (A "normal" grammar, i.e. grammar not involving rules engaging the category *sentence_segment* may easily be obtained from the POLINT grammar.)

At present, the POLINT system is based on ca 150 grammar rules encoded in PROLOG (ca. 60KB of the source ASCII code). These rules may be grouped as follows:

? sentence level rules: ca 35
? argument level rules: ca 45
? predicate level rules: ca 20
? other and auxiliary rules: rest

What follows is an example of a relatively simple rule encoded in PROLOG. These rules recognise the kernel of the verb group, based on a non-transitive finite verb, possibly reflexive or/and negated and optionally complemented by an adverb.

```
gv0(A,gv0_1(M),[[Ro,Li,Os,Cz],[R,L,mian,T],Rel],
[czas_0,0],[Tz,Neg,[Wcz,[Ro,Li,Os,Cz]]],
[[W,N]|X0],X4) :-
 neg_pred(A,Neg,[[W,N]|X0],X1),
 slo9(0,czas_0,[[Ro,Li,Os,Cz],Rel0,[R,L,mian,T]],
 X1,X2),
 eqw(X1,[[Wcz,_]|_]),eqw(Ro,R),eqw(Li,L),
 pron_refl_1(A,Rel0,X2,X3),
 eqw(X3,[[_,K]|_]),
 case([adv_poss(1,K) ->
gr_adv(A,1,M,S,X3,X4),
adv_poss(2,K) ->
gr_adv(A,2,M,S,X3,X4)|
gr_adv(A,3,M,S,X3,X4)]),
 sem(A,[gv_0(Neg),Tz,Rel0,S,Rel]).
```

3.2. Dictionary

The POLINT grammar requires a dictionary of the kind of lexicon-grammar, i.e. a lexicon where predicative words are supplied with syntactic information. At the pre-analysis stage the sentence is being scanned word by word for all predicative words, the syntactic requirements are read out from the dictionary and compared to properties of surrounding words. This observation usually permits formulation of a plausible hypothesis about syntax of the considered sentence in form of an expected configuration of sentence arguments. Such configurations are used as input parameter to the parsing module in order to make parsing more deterministic. This method proved particularly efficient while analysing sentences of medium size and medium complexity. The POLINT grammar has been tested with a dictionary containing ca 3000 dictionary entries (one word form per entry). Now, work is in progress to generate automatically (or semi-automatically) the system's dictionaries. The following resources are being tested as possible support of automatic dictionary generation: the morphological analyser LEM by Vetulani and Obrebski, cf. (Hajnicz, Kupsc, 2001), the resources of POLEX, GRAMLEX and CEGLEX projects (reported at LREC 2000 (Vetulani, 2000a)).

At present, the grammar is being translated into our new formalism FROG based on DCG-like rules well suited for free order languages with frequent discontinuity phenomena (Vetulani, 2002). This is a preparative step for further enhancement of the system's grammatical coverage to fully include discontinuous constructions. In this form, free distribution for non-commercial purposes is planned.

4. Coverage

In order to characterise the grammatical (and functional) coverage of the system we have listed a number of problems covered by POLINT:

- confirmation questions ("Czy" + affirmative sentence?)
- questions about arguments ("Kto/Who...?", "Co/What...?", "Z kim/With whom...?", etc.)
- questions concerning place ("Gdzie znajduje sie...?" / "Where is...?")
- questions concerning time ("Kiedy...?" / "When...?")
- questions concerning existence (Kogo nie ma...?/Who is absent...?)
- questions concerning name ("Jak nazywa sie...?" / "What is the name of...?")
- concerning type, position in a hierarchy ("Kim jest...?" / "Who is...?")
- about complement ("Czyim bratem jest...?" / "Whose brother is ....?")

At the predicate-argument level the word order is arbitrary (the system ignores differences in the degree of pragmatic markedness depending on the order of arguments).

The system recognises correctly a large class of nominal constructions. The following are the main types of noun phrases the system understands: proper names, complex proper names, complex noun group, common names, pronouns, genitive (possessive-like) constructions, complement nominal constructions. The nominal groups may be also completed with relative clauses (with possible iteration or embedding), adjectives etc. The predicates may take one, two or tree referential or locative arguments (Vetulani 1989). The predicate group may be, e.g.: personal forms of verbs, constructions with the auxiliary "byc" ("to be"), construction with noun in the instrumental case or with an adjectival group, constructions based on a supporting verb, construction with negation. The POLINT grammar was tested against the StClaus Corpus with satisfactory result (80% syntactic coverage for non-polypredicative, non-elliptical questions).

5. Efficiency

Contrarily to the most of NL systems written in PROLOG, the parser of POLINT is a real time system. This effect is difficult to reach for languages with flexible word order, like Polish, because of intensive and costly backtracking if grammar rules observe traditional grammar encoding procedures (various rules for various surface orderings, each rule reflecting the surface ordering of words). The main idea applied in the POLINT system to improve efficiency was to precede application of the grammar rules by a pre-analysis module. The pre-analysis was based on the concept of "lexical witness" for syntactic phenomena and on systematic usage made of lexicon

grammar dictionary. A lexical witness (as, e.g., relative pronoun for relative clause) may help to select appropriate grammar rule in a deterministic way. Exploration of syntactic or/and semantic requirements may help to limit the grammatical search space up to making the search deterministic in many cases. This additional information may be obtained from the dictionary when reading-in the sentence (in linear time).

6. Acknowledgements

The Author wishes to thank Alain Colmerauer for having hosted him as a young research fellow at GIA in Marseille, many years ago. This exercise permitted him to discover the charm of the Computer Question Answering Challenge.

7. References

Ajdukiewicz, K., 1965. *Logika Pragmatyczna (Pragmatic Logic)*. Warszawa: PWN.

Belnap, N.D.Jr., Steel, T.B.Jr., 1984. *The Logic of Questions and Answers.* New Haven: Yale Univ. Press.

Bien, J., 1998. Evaluating Analysers of Polish. In A. Rubio et al. (eds.), *First International Conference on Language Resources and Evaluation, Granada, Spain, 28.05.-30.05.1998, (Proceedings).* Paris: ELRA. 951-955.

Burger, J. et al., 2002. Issues, Tasks and Program structures to Roadmap Research in Question & Answering (Q&A). (www-nlpir.nist.gov/projects'duc/papers/qa.Roadmap-paper_v2.doc)

Hajnicz, E. and A. Kupsc, 2001. A survey of morphological analysers for the Polish language (in Polish). Prace IPI PAN (ICS PAS REPORTS), No 937. Warszawa: IPI PAN.

Martinek, J. and Z. Vetulani, 1991. An Expert system for Art History Data and Documents. In J. Banczerowski (ed.). *The Application of Microcomputers in Humanities, Adam Mickiewicz* University Press, Poznan, 63-74.

Vetulani, Z., 1989. *Linguistic problems in the theory of man-machine communication in natural language. A study of consultative question answering dialogues. Empirical approach.* Bochum: Brockmeyer.

Vetulani, Z., 1990. *Corpus of consultative dialogues. Experimentally collected source data for AI applications.* Wyd. Nauk. UAM (Adam Mickiewicz University Press), Poznan.

Vetulani, Z., 1997. A system for Computer Understanding of Texts. In R. Murawski and J.Pogonowski (eds), *Euphony and Logos* (Poznan Studies in the Philosophy of the Sciences and the Humanities, vol. 57). Amsterdam-Atlanta: Rodopi. 387-416.

Vetulani, Z., 2000a. Electronic Language Resources for POLISH: POLEX, CEGLEX and GRAMLEX. In: M. Gavrilidou et al. (eds.), *Second International Conference on Language Resources and Evaluation*, Athens, Greece, 30.05.-2.06.2000, (Proceedings), ELRA, 367-374.

Vetulani, Z., 2000b. Understanding Human Language by Computers: Projects in Artificial Intelligence and Language Technology. In Yosiho Hamamatsu et al. (eds). Formal Methods and Intelligent Techniques in Control, Decision Making, Multimedia and Robotics. Proceedings of the 2nd International Conference, Polish-Japanese Institute of Information Technology, Warsaw, October 2000, 218-229.

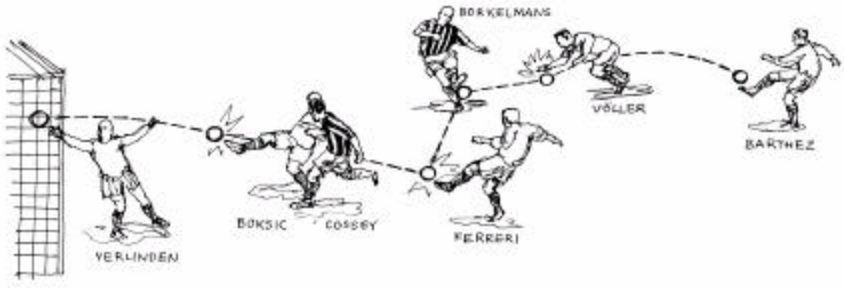
Vetulani, Z., 2002. A reinterpretation of the Definite Clause Grammar: Free Order DCG (FROG) (typescript, to appear).

Vetulani, Z. and Marciniak, J., 2000. Corpus Based Methodology in the Study and Design of Systems with Emulated Linguistic Competence. In Dimitris N. Christodoulakis (ed.). *Natural Language Processing - NLP 2000, Lecture Notes in Artificial Intelligence, no 1835*. Springer. 346-357.

Appendices

Appendix 1. Example of soccer game scene askable in POLINT

Information represented picturally in Figure 1 is encoded in the form of PROLOG predicates and accessible through POLINT.



User: - Jak nazywa sie pilkarz, który strzelil bramke? /What is the name of the player who scored?/
System: - Boksic.

Figure 1. Episode represented in the data-base[3] and a question-answer exchange.

Appendix 2. Experiment design

We are presenting here a detailed description of the St. Claus experiment setting.

1. Participants: A and B.

2. The scene (S) is represented by a complete picture (P) and an incomplete picture (P') (see below).

3. The participant A has the picture P'.

4. The participant B has the picture P.

5. Goal for A: to complete his knowledge about S.

6. Scenario for A: to ask questions to B (in writing).

7. Scenario for B: to answer the question (in writing).

8. Both A and B control (see) all previous questions and answers.

9. Restrictions:

- a single answer follows a single question (but no restrictions on the form of questions and answers),
- A and B are not permitted any form of communication (oral, gesture),
- dialogues are limited to 20 question-answer cycles (which corresponds to 30 min.-1h. sessions),
- a human supervisor is present during the session.

10. It is implicitly suggested to the participants that the experiment is a part of psychological research.

11. Instructions are read by the supervisor at the beginning of the dialogue session and no other explanations are allowed; the instructions are, however, available to participants (in writing) during the session.

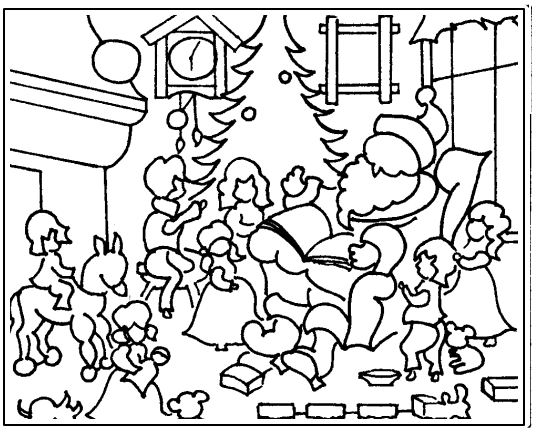


Figure 2. Complete picture (P)

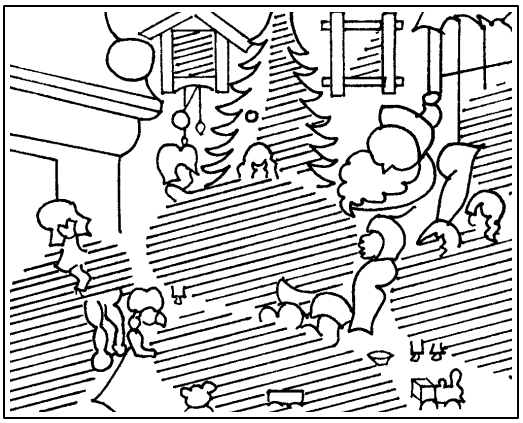


Figure 3. Incomplete picture (P')

LREC Workshop #8

Language Resources for Translation Work and Research

Programme

09:00 - 09:30	***Opening*** by Elia YUSTE, **Workshop Chair** (agenda - speakers introduction)
09:30 - 10:00	Silvia HANSEN & Elke TEICH, Computational Linguistics and Translation and Interpreting Departments (respectively), Saarland University, Saarbrücken, Germany *The creation and exploitation of a translation reference corpus*
10:00 - 10:30	Keynote Speaker - Maeve OLOHAN, CTIS, UMIST, Manchester, UK *Comparable Corpora in Translation Research: Overview of Recent Analyses Using the Translational English Corpus*
10:30 - 11:00	**Keynote Speaker** - Federico ZANETTIN, Università per Stranieri di Perugia, Italy *Corpora in Translation Practice*
11:00 - 11:20	Morning coffee break
11:20 - 12:00	Toni BADIA, Gemma BOLEDA, Carme COLOMINAS, Agnès GONZÁLEZ, Mireia GARMENDIA, and Martí QUIXAL, Universitat Pompèu Fabra, Barcelona, Spain *BancTrad: a web interface for integrated access to parallel annotated corpora*
12:00 - 12:30	Michael BARLOW, Department of Linguistics, Rice University, USA *ParaConc: Concordance software for multilingual parallel corpora*
12:30 - 13:00	Belinda MAIA, Faculdade de Letras, Universidade do Porto, Portugal *Corpora for terminology extraction - the differing perspectives and objectives of researchers, teachers and language service providers*
13:00 - 13:30	Lynne BOWKER, School of Translation and Interpretation, University of Ottawa, Canada *Working Together: A Collaborative Approach to DIY Corpora*
13:30 - 15:00	Lunch break
15:00 - 15:30	Elia YUSTE, Centre for Computational Linguistics, University of Zurich, Switzerland *Language Resources and the Language Professional*
15:30 - 16:00	Marita KRISTIANSEN & Magnar BREKKE, Norwegian School of Economics and Business Administration, Bergen, Norway *Textual and terminological bridgeheads for traversing the language gap*
16:00 - 16:40	Natalie KÜBLER, Intercultural Centre for Studies in Lexicology, University Paris 7, France *Creating a Term Base to Customize an MT System: Reusability of Resources and Tools from the Translator's Point of View*

16:40 - 17:00 Afternoon coffee break

17:00 - 17:30 Angelika ZERFASS, Language Technology Consultant, Germany
Evaluating Translation Memory Systems

17:30 - 18:00 Marie-Josée DE SAINT ROBERT, Chief, Terminology and Technical Documentation Section, Languages Service, United Nations Office at Geneva, Switzerland
Language resources at the Languages Service of the United Nations Office at Geneva

18:00 - 18:30 **Keynote Speaker** - Gerhard BUDIN, Department of Translation and Interpretation, University of Vienna, Austria
Global Content Management - challenges and opportunities for creating and using digital translation resources

18:30 - 19:00 ***Round-up Session***

Workshop Organisers and Programme Committee

[in alphabetical order]

Ms Elia YUSTE ***(Workshop Chair)***
Computerlinguistik, Institut für Informatik der Universität Zürich
Winterthurerstrasse 190
CH – 8057 ZÜRICH
Switzerland
yuste@ifi.unizh.ch

Dr Frank AUSTERMÜHL ***(Programme Committee Member and Main Adviser)***
Johannes Gutenberg-Universität Mainz
Fachbereich 23: Angewandte Sprach- und Kulturwissenschaft
Institut für Anglistik und Amerikanistik
An der Hochschule
D-76726 GERMERSHEIM
Germany
frank@austermuehl.de

Dr Gerhard BUDIN ***(Programme Committee Member and Keynote Speaker)***
Department of Translation and Interpreting Studies
University of Vienna
Gymnasiumstrasse 50
A-1090 VIENNA
Austria
gerhard.budin@univie.ac.at

Dr Maeve OLOHAN ***(Programme Committee Member and Keynote Speaker)***
Centre for Translation and Intercultural Studies
UMIST
PO Box 88
MANCHESTER M60 1QD
UK
maeve.olohan@umist.ac.uk

Dott. Federico ZANETTIN ***(Programme Committee Member and Keynote Speaker)***
Università per Stranieri di Perugia
Palazzo Gallenga - Piazza Fortebraccio, 4
I - 06122 PERUGIA
Italy
fz@federicozanettin.net
zanettin@unistrapg.it

Table of Contents

N.B. Papers are displayed here in the same order as they were presented on the Workshop day (please, refer to the **Programme** *above).*

Author Index

The creation and exploitation of a translation reference corpus

Silvia Hansen*, Elke Teich†

*Computational Linguistics, Saarland University
Postfach 151150, 66041 Saarbrücken, Germany
hansen@coli.uni-sb.de

† Applied Linguistics, Translation and Interpreting, Saarland University
Postfach 151150, 66041 Saarbrücken, Germany
e.teich@mx.uni-saarland.de

Abstract

While in many branches of linguistics monolingual reference corpora are widely used, in translation research as well as translation practice the concept of a translation reference corpus has not yet assumed a similarly important role. In this paper, we present the design of a German-English and French-English translation corpus and explore its use as a reference corpus for translatologists as well as translators. First, we introduce the basic computational techniques needed to build such a translation reference corpus, covering the preparation of the corpus as well as its linguistic annotation. Second, discussing some typical translation problems that occur in English-German and English-French translations, we show how the corpus can be queried making use of the linguistic annotation.

1. Introduction

In the last decade or so natural language corpora have assumed an increasingly important role in descriptive linguistics. Not only are they employed to inform lexicologists, lexicographers and grammarians in the construction of dictionaries and grammars, but also they gain importance as works of reference for linguists more generally. There are many corpora—especially for English (e.g., BNC[1], ICE[2], Bank of English[3])—that have been made accessible via the Internet with special user interfaces which allow one to query a corpus by means of KWIC concordances.

Also in translation research, corpora have started to become acknowledged as an important source of information in the investigation of theoretical issues in translatology, such as the question about the status of translations as a special kind of text with specific, possibly universal, properties. Here, the typical corpus is a parallel corpus consisting of two subcorpora, one containing source language (SL) original texts and the other containing translations of those texts into a target language (TL), where SL and TL texts are aligned (e.g., the Chemnitz corpora[4]). Some researchers advocate a three-way corpus design, where original texts in the TL are included as well (e.g., the Oslo corpora[5] as well as the work carried out at Saarbrücken (Teich & Hansen, 2001; Teich, 2001)), the latter being called a comparable corpus (cf. Baker, 1995; 1996). Also in translation practice, parallel corpora are increasingly being used in the form of translation memories. The compilation of such translation memories is supported by translation corpus workbenches. Thus, parallel corpora assume an increasingly important role both in theory and practice.

In this paper we explore the role of translation corpora as works of reference for translatologists as well as translators. It seems to us that there is a lacking interaction between the developers of corpus tools and researchers and practitioners in the field of translation. The goal of the present paper is to initiate such an exchange. We proceed in the following way. First, we discuss the basic computational techniques needed to make a corpus usable as a translation reference corpus (Section 2). We show how a corpus needs to be prepared (alignment, encoding) and how it should be enriched with linguistic information, so that it becomes possible to pose queries to it that are interesting and relevant from a translation point of view. Second, we show how a translation corpus can be queried with a parallel concordancing tool. We illustrate the use of an English-German-French translation reference corpus for solving some typical translation problems that occur in translating from English into German and from English into French (Section 3). Section 4 concludes the paper with a summary and some issues for future work.

2. Computational techniques

Corpus preparation. For the creation of a translation reference corpus, a parallel corpus needs to be aligned. For this purpose, an alignment program must be applied. One such program is Déjà Vu (Atril, 2000). Figure 1 shows a German SL and an English TL text aligned with this tool.

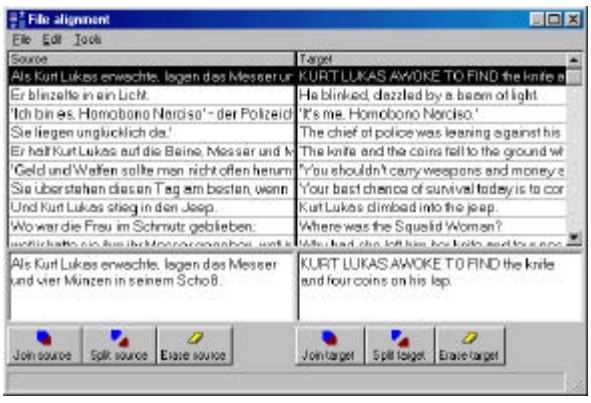


Figure 1: Multilingual corpus alignment

[1] http://sara.natcorp.ox.ac.uk/lookup.html
[2] http://www.ucl.ac.uk/english-usage/ice-gb/sampler/download.htm
[3] http:// titania.cobuild.collins.co.uk/form.html
[4] http://www.tu-chemnitz.de/phil/InternetGrammar/
[5] http://www.hf.uio.no/german/sprik/english/index.shtml

Déjà Vu aligns a text and its translation on sentence basis, storing the aligned texts in one file or in two separate files depending on the requirements of the query tool used in later stages of analysis. Files can be exported to translation workbenches and to Microsoft Excel and Access. Figure 2 shows a Déjà Vu output in a TSV (tab separated vector) format.

"Als Kurt Lukas erwachte, lagen das Messer und vier Münzen in seinem Schoß." "Kurt Lukas awoke to find the knife and four coins on his lap."
"Er blinzelte in ein Licht." "He blinked, dazzled by a beam of light."
"'Ich bin es, Homobono Narciso' - der Polizeichef stand an seinen Jeep gelehnt -, 'fast hätte ich Sie überfahren. Sie liegen unglücklich da.'" "'It's me, Homobono Narciso.' The chief of police was leaning against his jeep."
"Er half Kurt Lukas auf die Beine, Messer und Münzen fielen herunter, Narciso hob sie auf." "The knife and the coins fell to the ground when he helped Kurt Lukas up."

Figure 2: Déjà Vu alignment format

Also, we encode each text of the corpus in terms of a header that provides meta-information such as title, author, publication, translator, etc as well as text type/register information (domain, tenor and mode of discourse). This is important to enable corpus queries according to register or other independent variables.

Text files are encoded in XML using a modified version of the Text Encoding Initiative (TEI) standard[6] (a short header including meta-information is illustrated in Figure 3) and employing a standard XML editor (here: XML Spy[7]). The text body is annotated for headings, sentences, paragraphs, etc.

```
<tei.2>
  <teiHeader>
    <fileDesc>
      <filename>infanta_tl_e.txt</filename>
      <subcorpus>fiction (trans_en)</subcorpus>
      <language>English</language>
      <titleStmt>
        <title>Infanta</title>
        <author>
          <name>J. M. Brownjohn</name>
        </author>
      </titleStmt>
      <translation>
        <direction>German-English</direction>
      </translation>
      <sourceText>
        <title>Infanta</title>
        <language>German</language>
        <author>
          <name>Bodo Kirchhoff</name>
        </author>
      </sourceText>
    </fileDesc>
    <encodingDesc>Modified TEI</encodingDesc>
  </teiHeader>
  <text>
    <body> </body>
  </text>
</tei.2>
```

Figure 3: XML corpus encoding

Corpus annotation. A translation reference corpus should at least be annotated with part-of-speech and syntactic information. Part-of-speech tagging is carried out fully automatically, either using a rule-based or a statistical approach, where recently, statistical approaches prevail. For multilingual applications, it is important that the tagger can be used for more than one language. Analyzing a corpus in terms of syntactic structure is still a challenging task and cannot be carried out automatically with satisfactory accuracy yet. Recently researchers in computational linguistics who are interested in the accurate parsing of large amounts of text promote what has been called interactive parsing, where a parser carries out a shallow parse and a human may correct or add information to the proposed parse. For example, the parser assigns syntactic labels to the elements of a clause, but does not resolve syntactic ambiguities of particular kinds, such as PP-attachment, leaving this to the human to deal with.

One system which combines part-of-speech tagging and shallow parsing is the ANNOTATE system (Plaehn & Brants, 2000) under development in the TIGER[8] and NEGRA[9] projects. ANNOTATE uses the TnT tagger (Brants, 2000) that can be applied multilingually and has been trained on a number of languages, including English and German. The tag set used for English is the Susanne tag set (Sampson, 1995); the one for German is based on the Stuttgart-Tübingen tag set (Hinrichs et al., 1995). ANNOTATE carries out an analysis of phrase categories as well as grammatical functions using a program based on Cascaded Markov Models (CMM (Brants 1999a, 1999b)). During the interactive annotation with ANNOTATE (see Figure 4), terminal nodes are labeled for parts-of-speech and morphology, non-terminal nodes are labeled for phrase categories and edges are labeled for grammatical functions.

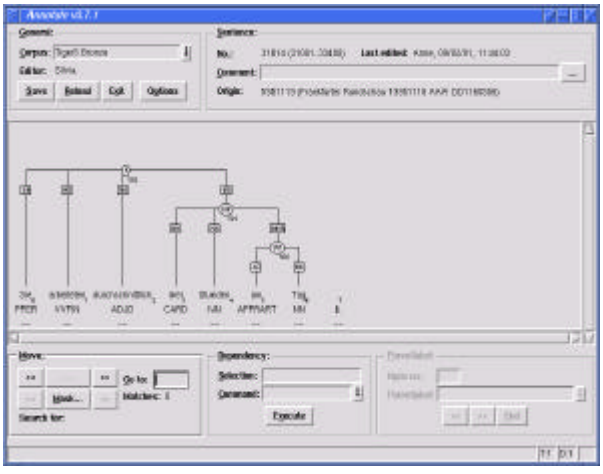


Figure 4: Interactive annotation with ANNOTATE

The tagged and parsed corpus data are stored in the form of a relational database, but can be exported to text format.

Corpus querying. For parallel concordancing, query tools such as the IMS Corpus Workbench (Christ, 1994)

[6] http://www.tei-c.org/index.html
[7] http://www.xml-spy.com
[8] http://www.ims.uni-stuttgart.de/projekte/TIGER/
[9] http://www.coli.uni-sb.de/sfb378/projects/NEGRA-en.html

can be employed. Its query processor (CQP) allows queries for words and/or annotation tags on the basis of regular expressions. For an example of a query executed on a parallel English-German corpus see Figure 5.

```
# Query: DE_EN; passives-de = [pos="VB.*"] [] {0,1} [pos="VVN.*"];
#------------------------------------------------------------------------------
   729: newspaper . A ferry had <been sunk> just off the island . ' I
  -->de_de: In den Gewässern vor der Insel war eine Fähre gesunken .
   850: country ' s future will <be decided> today . Yours too , perha
  -->de_de: Zukunft des Landes entscheidet sich heute .
   927: nced , because shots had <been fired> at a remote polling stati
  -->de_de: Der Schriftsteller und er müßten aufbrechen , in einem
```

Figure 5: Sample query with CQP

3. Solving translation problems with a translation reference corpus

With a corpus annotated in the way described in the preceding section, we now have available a translation resource that is searchable in a meaningful way. While with a raw text corpus we can only formulate string searches, we can now make use of the annotations in querying the corpus. In the following, we discuss some examples of translation problems between English, German and French. The examples are taken from two genres, narrative and factual writing. For querying the corpora selected, we use CQP (cf. Section 2).

English present and past perfect. While both English and German have present and past perfect tenses, their usage conditions differ cross-linguistically and it is sometimes hard to tell whether a one-to-one translation is the appropriate choice. The French tense system also has present and past perfect, but there are other options as well. Figure 6 shows two parallel concordances for English present and past perfect in narrative texts.

```
# Query: DE_EN; [pos="VH.*"] [pos="RR.*"] {0,1} [pos="VVN"];
#------------------------------------------------------------------------------
 509: night , he said . Adaza <had run> them off and was selling
-->de_de: Der Fotograf Adaza habe sie angefertigt und verkaufe sie für
 1120: igure and the blood that <had discoloured> a whole patch of grass
-->de_de: Die Fahrt endete vor einer Zwergschule , in der das Wahllokal
war , vor einer Blutlache , die ein ganzes Rasenstück färbte , vor einer
 2779: footsteps . Their guest <had appeared> on the terrace . Kurt Luk
-->de_de: Der Gast hatte auf die Terrasse gefunden .
 2953: ' Very few of our guests <have ever found> their way to this
-->de_de: ' Nur wenige unserer Gäste haben bisher auf diese Terrasse
gefunden

# Query: FR_EN; [pos="VH.*"] [pos="RR.*"] {0,1} [pos="VVN"];
#------------------------------------------------------------------------------
 1239: ver ventured there ; she <had even built> a low wall with her own
-->fr_fr: l ' épouse du pasteur avait même construit de ses mains un
 1395: sk , until the last rose <had dropped> into his open handkerchie
-->fr_fr: Il continua sa besogne , jusqu ' à ce que la dernière tête de rose
fût tombée dans son mouchoir ouvert .
 1478: , ' Do you realize what <has happened> to you ? When you
-->fr_fr: - Te rends-tu compte de ce qui vient de se produire en toi ?
 1499: ted Sheikh , and now you <have turned> into a thief ! I have
-->fr_fr: En arrivant ici ce matin , tu étais un cheikh respecté , et
maintenant tu es devenu un voleur !
```

Figure 6: Parallel concordances for English perfect

What can be seen here is that in translations into German, the translational choice is in fact often one-to-one, but also, past tense or present subjunctive is used. In the French parallel texts, we find direct translations, but also *passé anterieur* and "venir de".

English reduced relative clauses. Reduced relative clauses are a typical feature of English and French, but not so much of German. We can thus expect translational problems from English into German. A concordance query to a parallel corpus shows the translational options available (cf. Figure 7).

```
# Query: DE_EN; [pos="N.*"] [pos="VVN"];
#------------------------------------------------------------------------------
 197: g away under tin roofs . <Carcasses suspended> from chains
-->de_de: An Ketten hängend , bluteten zuckende Rinder aus . Schweine
 2180: ed behind on his own . A <crucifix reposed> on his lap in place of
-->de_de: An Stelle des Buchs lag ein Kreuz in seinem Schoß .
 2833: And the mountains wore <cloud-caps frayed> at the edges by
-->de_de: Und die Berge trugen Wolkenhüte , die zur Sonne hin
ausfransten .

# Query: FR_EN; [pos="N.*"] [pos="VVN"];
#------------------------------------------------------------------------------
1864: of him . This time , the <instrument provided> by Providence was
-->fr_fr: L ' instrument de la Providence fut cette fois un passe-temps
 2812: the presence of all the <people gathered> on the Blata , and in his
-->fr_fr: ' Le cheikh Francis et le patriarche se donnèrent l ' accolade
devant le peuple réuni sur la Blata , et dans son sermon , sayyedna parla
```

Figure 7: Parallel concordances for English reduced relative clauses

We see that English reduced relative clauses are indeed translated into French one-to-one (or zero-equivalent), whereas in German translations we find the present participle or full relative clauses (or zero-equivalent).

English cleft sentences. Cleft (and pseudo-cleft) constructions are a typical feature of the English grammatical system (cf. Erdmann, 1990). While they do exist in German as well, German has other options of realizing information distribution patterns, e.g., by word order variation. Because here, the search space for a translational choice is rather wide, finding a translational equivalent for an English cleft construction is therefore a notorious problem in translating from English into German. Again, a parallel concordance can provide help (cf. Figure 8).

```
# Query: DE_EN; [word="it|It"] [pos="VB.*"] [pos!="JJ.*"] {1,2}
[pos="DDQ.*|PNQ.*|CST"];
#------------------------------------------------------------------------------
 8620: simply as N , because <it is N that> makes this one-way function
-->de_de: Es ist dieses N, das die Einwegfunktion umkehrbar macht,
 8967: cells growing . <It is these properties that> make them attractive
-->de_de: Gerade diese Eigenschaften lassen sie als Wirkstoffe gegen
Krebs vielversprechend erscheinen.
 9112: is in control . <It is they alone that> persist from one generation to
-->de_de: Nur die Gene bleiben in der Generationenabfolge erhalten.
 9523: History records that <it was Galileo who> was foremost in
-->de_de: Die Geschichte belegt, daß vor allem Galilei die Zeit als eine
fundamentale Größe im gesetzesgleichen Wirken des Kosmos etablierte.
```

Figure 8: Parallel concordance for English clefts

The concordance shows that for compensation a focus particle or adverb (e.g., `gerade´) can be used to signal the syntactic focus.

4. Summary and conclusions

In this paper, we have suggested that translation corpora can assume the role of works of reference for translators and translatologists. In order for translation corpora to serve this purpose, they need to be enriched with linguistic information (Section 2). We have shown that some minimal linguistic annotation (part-of-speech, shallow phrase structure) can already make a translation corpus a valuable resource for dealing with some typical translation problems (Section 3).

While parallel concordancing tools operating on the basis of syntactic annotations already offer useful information, there are a number of further developments that can increase the value of a translation corpus. First, in corpus searches, it may be useful to be able to express constraints on the target language expression as well. Only few parallel concordance programs allow for this. Second, it could be very useful to be able to refer to a comparable TL corpus as well for a comparison of the translations with original TL texts. Third, for dealing with more complex kinds of translation problems, a translation corpus should be annotated with more abstract kinds of linguistic information, e.g., semantic and discourse information. This requires more comprehensive annotation methods and more sophisticated query facilities – both of which are current research issues in computational linguistics (cf. Teich et al., 2001).

Finally, from the perspective of the developers of corpus tools, translation corpora are an invaluable source for testing the applicability of such tools in multilingual contexts.

5. References

Atril, Development SL, 2000. *Déjà Vu. Productivity system for translators*. Software Manual. (http://www.atril.com/).

Baker, M., 1995. Corpora in translation studies: An overview and some suggestions for future research. In *Target* 7(2): 223-245.

Baker, M., 1996. Corpus-based translation studies: the challenges that lie ahead. In H. Somers (ed.), *Terminology, LSP and Translation: Studies in Language Engineering in Honour of Juan C. Sager*. Amsterdam: Benjamins: 175-186.

Brants, T., 1999a. *Tagging and Parsing with Cascaded Markov Models - Automation of Corpus Annotation*. Saarbrücken Dissertations in Computational Linguistics and Language Technology, Volume 6, German Research Center for Artificial Intelligence and Saarland University.

Brants, T., 1999b. Cascaded Markov Models. In *Proceedings of 9th Conference of the European Chapter of the Association for Computational Linguistics (EACL-99)*. Bergen.

Brants, T., 2000. TnT - A Statistical Part-of-Speech Tagger. In *Proceedings of the Sixth Applied Natural Language Processing Conference ANLP-2000*. Seattle.

Christ, O., 1994. A modular and flexible architecture for an integrated corpus query system. In *Proceedings of COMPLEX 94, 3rd Conference on Computational Lexicography and Text research*. Budapest: 23-32.

Erdmann, P., 1990. *Discourse and grammar. Focussing and defocussing in English*. Tübingen: Niemeyer.

Hinrichs, E., H. Feldweg, M. Boyle-Hinrichs, and R. Hauser, 1995. *Abschlußbericht ELWIS. Korpusunterstützte Entwicklung lexikalischer Wissensbasen für die Computerlinguistik*. Technical report, University of Tübingen.

Plaehn, O., and T. Brants, 2000. Annotate - An Efficient Interactive Annotation Tool. In *Proceedings of the Sixth Conference on Applied Natural Language Processing (ANLP-2000)*. Seattle.

Sampson, G., 1995. *English for the Computer*. Oxford: Oxford University Press.

Teich, E., S. Hansen, and P. Fankhauser, 2001. Representing and querying multi-layer corpora. In *Proceedings of IRCS Workshop on Linguistic Databases*. Philadelphia.

Teich, E., and S. Hansen, 2001. Methods and techniques for a multi-level analysis of multilingual corpora. In *Proceedings of Corpus Linguistics 2001*. Lancaster.

Teich, E., 2001. *Contrast and commonality in English and German system and text. A methodology for the investigation of the contrastive-linguistic properties of translations and multilingually comparable texts*. Habilitationsschrift (submitted for publication), Saarland University.

Comparable Corpora in Translation Research: Overview of Recent Analyses Using the Translational English Corpus

Maeve Olohan

Centre for Translation and Intercultural Studies
UMIST
PO Box 88
Manchester
M60 1QD
maeve.olohan@umist.ac.uk

Abstract

This paper discusses the use of a comparable corpus in translation research, where a comparable corpus comprises, on the one hand, a corpus of translations and on the other hand a corpus of non-translated texts, both corpora being similar in composition, size and other attributes. The Translational English Corpus, housed at the Centre for Translation and Intercultural Studies in Manchester, is presented as an example of a comparable corpus used in researching translation. The rationale for using a corpus of this kind to research translation is addressed. Results of a number of empirical analyses are then summarised, and the potential development and future exploitation of this corpus resource are outlined.

1. Corpora and Translation Studies

According to Michael Stubbs (2001: 151), corpus linguistics is concerned with "what frequently and typically occurs", as opposed to isolated, unique instances of language: "Corpus linguistics […] investigates relations between frequency and typicality, and instance and norm. It aims at a theory of the typical, on the grounds that this has to be the basis of interpreting what is attested but unusual". The corpus-based approach to studying translation has rapidly gained in popularity over the past eight to ten years, with a wealth of data now emerging from studies using parallel corpora, multilingual corpora and comparable corpora. In addition, corpora, whether of the ad-hoc or the reference kind, are proving a useful tool in the translator training classroom. Furthermore, most specialised translators would now be lost without their translation memory system, i.e. essentially an aligned parallel corpus of source texts and their translations.

This paper focuses on the first of these applications of corpora, namely corpora in translation research. The special issue of *Meta* on this topic published in 1998 is useful for an overview of work in this area, as is Chapter 3 of Kenny, 2001). Olohan (forthcoming b) highlights some of the strengths and limitations of corpus-based translation studies, based primarily on views put forward by Maria Tymoczko (1998) and Ian Mason (2001). This paper therefore does not present an overview of the literature nor does it address the criticisms levelled at corpus-based translation studies. Instead it assumes an understanding of corpus-based translation studies as the application of corpus analysis techniques, both quantitative and qualitative, to the study of aspects of the product and process of translation. Built into this is the recognition that there are differing opinions as to what aspects of translation we can apply these techniques to, and that the methodology requires refinement through application, discussion of findings and critical assessment. This process is now being undertaken by an ever-growing number of scholars in translation studies and it will ultimately lead to a better understanding of the scope, significance, usefulness and appropriateness (or not) of corpora to study translation processes and products.

2. Translation as Process and Product

The empirical study of the translation process emerged almost twenty years ago in translation studies, following on the heels of developments in second language research. It has since involved the identification, description and analysis of what happens during translation, i.e. of the mental steps taken by translators between, and including, reception of the source text and production of the target text. Introspection (in particular the think-aloud method) has been the principal methodological tool used in investigations of the translation process, and the introspective studies carried out to date have been largely data-based and descriptive, often focusing on specific aspects of the translation process (e.g. use of reference material, decision-making criteria). While a number of researchers have carried out descriptive empirical research in this area using the think-aloud method, there are methodological difficulties with research of this nature and, as a result, these attempts to investigate the cognitive processes at work during translation have met with scepticism from some quarters. Criticism has focused in particular on the methodology for data elicitation and collection, including its inability to provide access to thought processes which are subconscious or automated, but also on issues of scale and object of investigation.

While translation process researchers have readily acknowledged the potential shortcomings of this data elicitation method, it has been welcomed as a means of gaining some insight into something which is otherwise not accessible to the researcher. However, an alternative approach to translation process research is suggested by Bell (1991), who proposes that a model can and should be developed through a combination of induction (i.e. inferring processes from the product) and deduction (i.e. using introspective data such as diaries) (ibid.: 29). He suggests describing "translation competence in terms of generalizations based on inferences drawn from the

observation of translator performance" (ibid.: 39). He proposes to observe translator performance by analysing the translation product: "by finding features in the data of the product which suggest the existence of particular elements and systematic relations in the process" (ibid.). This approach lends support for the suggestion that the compilation and use of corpora of translations would allow us to analyse features of translation products which can provide evidence of translation processes, both conscious and subconscious, particularly if we can investigate "relations between frequency and typicality, and instance and norm", as advocated by Stubbs (2001: 151).

3. TEC – Translational English Corpus

TEC (Translational English Corpus) is a corpus of translated English held at the Centre for Translation Studies in Manchester. It consists of contemporary written translations into English of texts from a range of source languages and it was designed specifically for the purpose of studying translated texts. There are currently just under 7 million words in the corpus, made up of full running texts falling into four text types – fiction, biography, newspaper articles and in-flight magazines – with fiction representing more than 80% of the total. The translations are by native speakers of English, both male and female, and mostly date from 1983 onwards. In addition to the texts themselves, information is held on the translator and translation process, compiled via questionnaires to translators and publishers, and stored in header files.

One of the fundamental concepts in corpus-based translation studies has been the notion of comparable corpus, defined by Mona Baker (1995: 234) as "two separate collections of texts in the same language: one corpus consists of original texts in the language in question and the other consists of translations in that language from a given source language or languages…both corpora should cover a similar domain, variety of language and time span, and be of comparable length". Baker's initial groundbreaking work posited a number of features of translation which could be investigated using comparable corpora (Baker, 1996), for example, that translations tend to be more explicit on a number of levels than original texts, and that they simplify and normalise or standardise in a number of ways.

Much of the empirical analyses carried out thus far have focused on the literary component of TEC, namely fiction only, or fiction and biography. Thus, the corpus of original English put together for use as a comparable corpus is a set of texts selected from the imaginative writing section of the British National Corpus (BNC). It has been constructed specifically to match TEC in terms of both composition and date of publication (from 1981 onwards). As in the case of TEC, the BNC texts are produced by both male and female authors, all native speakers of English. Unlike TEC, however, some of the texts in the BNC subcorpus are extracts – albeit as long as 40,000 words. This was not deemed a significant difference in the current studies as they investigate intrasentential patterns. The Translational English Corpus is being added to all the time, which means that successive studies present data from TEC at different stages in its growth and the composition of the BNC subcorpus is modified accordingly.

Given that TEC and the BNC subcorpus are comparable in terms of parameters such as size and composition, features of the language of translation identified in the corpus of translation may thus be compared with features of non-translated language as found in the BNC subcorpus. Much of the work with TEC carried out to date has focused on syntactic or lexical features of translated and original texts which may provide evidence of the processes of explicitation, simplification or normalisation mentioned previously. It is possible to catch glimpses of these processes in think-aloud protocols where the translators are conscious of them and are employing them as part of controlled cognitive processes. However, corpus data may provide evidence which may constitute the result of such processes operating on a more subconscious level too.

4. Examples of Comparable Corpus Analyses

It is beyond the scope of this paper to present in detail the studies which has been carried out thus far using TEC and a BNC subcorpus. However, the results of some recent studies are summarised here, followed by an outline of some future directions for translation research using comparable corpora.

4.1. Optional Reporting *that*

The first large-scale empirical study using TEC and the BNC subcorpus indicated a substantially heavier use of the reporting *that* with verbs SAY and TELL in constructions such as examples [1] to [4] in TEC than in the BNC subcorpus, and it was suggested that this may be evidence for a tendency towards explicitation in translated English (Olohan and Baker, 2000).

[1] *He **says that** the ship is now forty-eight hours overdue and he wants explanations* (BNC)

[2] *He **says** the whole army is unsettled because it's known that Famagusta will never give up while it expects a relieving ship to arrive* (BNC)

[3] *I **told** him **that** I didn't know who it was he wanted to speak to, but he was quite insistent that he had seen you come in* (TEC)

[4] *I **told** him I thought it was a stupid thing for him to do* (BNC)

Explicitation has long been considered a feature of translation and has been investigated by a number of scholars (e.g. Vanderauwera, 1985, Blum-Kulka, 1986) who have identified different means or techniques by which translators make information explicit, e.g. using supplementary explanatory phrases, resolving source text ambiguities, making greater use of repetitions and other cohesive devices. In general, explicitation has referred to the spelling out in the target text of information which is only implicit in a source text. In these corpus-based studies, however, we are interested in the making explicit in a translation of information which is less likely to be made explicit in a non-translated text of the same language.

Scott Burnett (1999) examined the behaviour of some forms of other verbs of this type, and Olohan (2001) looked

at PROMISE, which can also take an optional *that*. The same pattern of heavier use of *that* in TEC compared with BNC was also found in these smaller-scale studies.

4.2. Other Optional Syntactic Features

Olohan (2001 and forthcoming a) presents a broad overview of some other optional syntactic features in English and their occurrence in TEC and the BNC. Since the focus of the research was subconscious processes of explicitation and their realisation in linguistic forms in translated texts, optional syntactic features were pinpointed, based on the hypothesis that, if explicitation is genuinely an inherent feature of translation, translated text might manifest a higher frequency of the use of optional syntactic elements than written works in the same language, i.e. translations may render grammatical relations more explicit more often – and perhaps in linguistic environments where there is no obvious justification for doing so – than authors in English.

Working with untagged corpora only, the analysis focused predominantly on frequency of occurrence of optional features and less so on the relationship between occurrence and omission. It can thus be regarded as a first step only. However, initial findings certainly encourage more detailed examination, suggesting for example that the use of the relative pronoun *which* is twice as frequent in TEC than in the BNC subcorpus. Similarly, a study of *who* (in the following constructions: *who is, who's, who've, who have, who'd, who did, who had* and *who would*) found that TEC has a significantly higher overall occurrence of the *who* form. Closer investigation of the co-text, which would be required to differentiate interrogative from relative usage, and to determine the optional vs. non-optional nature of the relative pronoun in each case, has not yet been carried out for all of these forms. However, in the case of *who is* and *who's*, a separation into interrogative and non-interrogative use showed that 44% of BNC occurrences were interrogative, as opposed to only 15% of TEC occurrences.

The occurrence of the complementiser *to*, which is optional following HELP, was analysed (see examples 5 and 6).

[5] *You have special skills and experience which will* ***help*** *us* ***to*** *achieve our objective.* (BNC)

[6] *She only wished Antonia were there with her to* ***help*** *her think over all the things Thomas said.* (BNC)

The data showed that although the word form *help* is more frequent in TEC, its verbal use in both corpora is quite similar. Of these verbal uses, the complementiser *to* is used in 37.5% of TEC instances, compared with only 26% of the BNC occurrences.

The use of *while* preceding a gerundial, i.e. *while *ing*,* and *after* preceding *having + participle* was measured in both corpora. *While *ing* was seen to occur more than twice as often in TEC than in BNC. A count of *after *ing *ed* (which obviously does not take irregularly formed past participles into account) also shows a tendency for TEC to use this construction more frequently than BNC, although the construction was relatively rare in both corpora.

Finally, *in order* may be omitted before *to* and may occasionally be omitted before *for* or *that*. While the investigation of every instance of the items *to, that* and *for* to see whether an *in order* has been omitted is not practical, it is possible to measure usage of *in order to, in order for* and *in order that* and compare results from the two corpora. This investigation showed a marked difference in usage of *in order to*, with 250 instances in BNC compared with 1,225 in TEC. The other forms, *in order for* and *in order that*, were infrequent in the two corpora but both occurred more often in TEC than in the BNC subcorpus.

4.3. Personal Pronouns

A small-scale study of the use of personal pronouns in both corpora is also presented in Olohan (forthcoming a). Frequencies of personal pronouns occurring with verb forms *will, have, am, is, has* and *are*, both within verb contractions and within non-contracted forms, were recorded. The data show that, when used in conjunction with these particular verb forms, personal pronouns *I, you, he, she, we* and *they* are more common in the BNC subcorpus than in TEC. The differences are extremely striking in the case of *I* (23,409 in BNC; 16,178 in TEC), and also quite marked in the case of *you, she* and *we*. The pronouns *he* and *they* occur with these verbs with almost the same frequency in the two corpora.

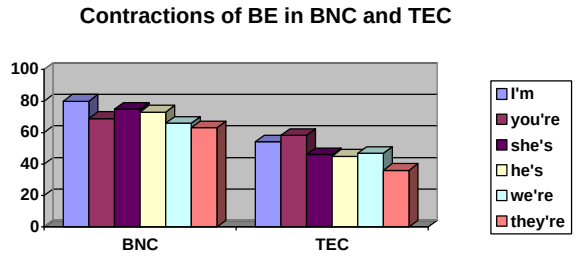
4.4. Contractions

As reported in Olohan and Baker (2000), the linguistics literature on use and omission of *that* with a range of verbs indicated that omission was more likely in informal contexts. Preliminary analysis of co-occurrence of *that* omission and contracted forms (as a crude measure of informality) revealed a definite correlation in both corpora between use of contracted forms and omission of *that.* Thus, despite lower incidence of contractions in TEC and higher incidence of that omission in BNC, the likelihood of co-occurrence of a contracted form and omission of *that* (in the same concordance line) was very similar in both corpora. In other words, the BNC texts were more likely to omit *that* and use contractions; the TEC texts were more likely to include *that* and not use contractions. This correlation suggested that contractions merited further investigation.

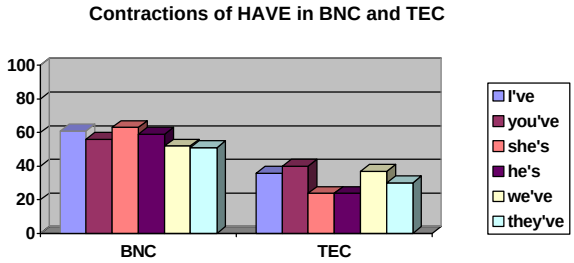
Further detailed analysis of all contracted forms in the corpora revealed that there are higher occurrences and a greater variety of contracted forms in BNC than in TEC. In many cases, the number of occurrences of a form in BNC is double that seen in TEC. (It is worth noting again at this point that the corpora under investigation are extremely similar in terms of size and composition.) In addition, there was a general preference for contracted forms over the corresponding long forms in BNC, while the TEC data showed a general tendency to use the long form in preference to the contracted one. For example, for all *'s* contractions (not including the possessive*'s*, thus for the following forms: *it's, that's, he's, there's, she's, what's, let's, who's, where's, here's, how's*), the contracted form is significantly more common than the long form in BNC. This is not true for TEC, where the long form is the more frequent in 8 out of the 11 forms. In TEC, the contracted form is more frequent only for *that's, what's,* and *let's*, but in these cases represents a smaller proportion of the

combined total occurrences of long and contracted forms than does the long form in BNC.

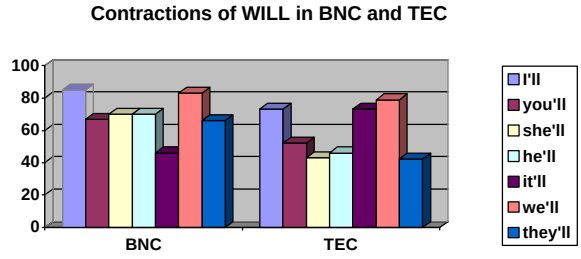
Splitting the analysis into verbs, we can see from Graphs 1, 2 and 3 that there is a greater incidence of contracted forms with personal pronouns in BNC than in TEC for present-tense forms of BE, HAVE and WILL.



Graph 1 Contractions of BE in BNC and TEC, represented as percentage of combined total for contracted and long forms



Graph 2 Contractions of HAVE in BNC and TEC, represented as percentage of combined total for contracted and long forms



Graph 3 Contractions of WILL in BNC and TEC, represented as percentage of combined total for contracted and long forms

As far as common *not*-contractions are concerned, the overall tendency in both corpora is to contract. However, the proportion of contracted forms is smaller in TEC than in BNC in all cases, and for 2 forms examined, *couldn't* and *wouldn't*, TEC is, in fact, more likely to use the long form. Biber et al. (1999: 1131) show that DO + *not* is contracted almost 100% of the time in conversation, around 75% in fiction, 60% in news text and 5% in academic text. From the data used in this study, on average across forms *don't, doesn't* and *didn't*, the rate of contraction of *not* with DO in BNC is 74%, thus very close to Biber et al.'s finding of 75% for fiction. In TEC, on the other hand it is 58%, thus considerably lower.

4.5. Dialectal features

Most of the contractions which featured in the analysis above were of verbs BE, HAVE and WILL or of the negation *not.* However, the BNC subcorpus had a selection of other types of contractions. Many are typical of spoken English, such as the contraction of multisyllabic modifiers e.g. *actu'lly, accident'lly, contradict'ry, prob'ly, fav'rite, gen'rous*. Some interjections also had contracted forms, e.g. *ah'm* and *fuck'em*, again characteristic of the spoken language, as were contractions of *ing* (e.g. *bleed'n*), *and* (e.g. *this'n*) and *than* (*better'n*). Some contractions were also clearly dialectal or sociolectal, with indicators of regional variations such as the dropped *h* in *be'aviour, be'ind, ware'ouse*, or the Scottish *does'na* and *hav'na* (where there is, in fact, no elision between the two words). There were 102 occurrences of *e's* in BNC (dialectal version of *he's*) and none at all in TEC. Finally, other forms found were *d'* (= *do*), *y'* (= *you*), *th'* (=*thou* or *thy*) and *t'* (= *to* or *to the*). All occur considerably more frequently in BNC than in TEC, e.g. *y'know* occurs 22 times in BNC and only once in TEC; *d'you* occurs 362 times in BNC, compared with 72 occurrences in TEC. The last two in particular indicate regional variation and do not occur at all in TEC; by contrast, *t'*, representing *to, to the* or *the* occurs in front of 99 different nouns or modifiers in BNC (see examples 7 and 8), and *th'* occurs 137 times (see example 9).

[7] *"It's a blessing it's a mild winter up ti now," he commented. "It would've been a bad time for* ***t'road*** *between* ***t'two*** *farms ti be blocked wi' snow."* (BNC)

[8] *"We're to go down* ***t'village****, to* ***t'stables****," George told his father, as he retrieved the reins.*(BNC)

[9] *"****Th'mind*** *what I say and* ***th'll*** *doubtless find there's no better place than Jarman House."* (BNC)

5. Directions of Future Research

The picture which emerges from these sets of data and the more detailed quantitative and qualitative analyses which have been done is one of a general preference for longer surface forms in TEC where there is an option between longer and shorter forms. This appears to apply as much to potential contractions of word forms as to syntactic explicitation of relations between clauses, for example in the use of the optional *that* with certain verbs or in the inclusion of relative pronouns where they are optional, i.e. in relative clauses where the co-referential NP is not the subject of the relative clause.

Furthermore, the tendency towards explicitation may extend to lexical choices, where some kind of repetition of nouns in translation may be preferred over use of pro-forms. In addition, TEC appears to contain a more standard variant of the English language, with fewer dialectal or sociolectal markers.

A tentative attempt has been made to link these findings with Biber's dimensions of English (1988 and 1995), with a view to determining to what extent TEC fiction is similar or different to the features of English fiction as analysed by

Biber. These preliminary findings seem to indicate that TEC fiction is not as typical of fiction in English as the works of fiction in the BNC subcorpus. Furthermore, some of the results suggest that TEC fiction may exhibit features more typical of academic prose in English. If this is borne out by future investigations it may contribute to an understanding of the nature of literary translation and its reception in the British literary system. However, there are many features to be investigated in the future to shed further light on this issue.

A criticism sometimes levelled at translation scholars is that we focus too much on literary text and literary translation. One area in which this research can be broadened is to add other genres to TEC. A subcorpus of non-fictional translated works of social science, politics, history etc. would provide an interesting contrast to the fiction subcorpus. Similarly, a bigger biography component would enable useful analyses of that genre to be carried out, taking into account in particular its position somewhere on the continuum between fictional and factual writing.

One aspect of research of this kind which has not been discussed in this paper is the investigation of individual translators. Due to the design of TEC and the incorporation of more than one translation by several translators, it is possible to compare translators and their practices; for example, Baker (2000) discusses the development of a methodology for investigating the style of a literary translator and Olohan (forthcoming b) examines the contraction patterns of two well-known translators across a number of translated works. There is much scope for further research of this kind.

At a conference workshop such as LREC where the emphasis is on practical application of technology in the translation process, one might question the relevance of this kind of detailed analyses of lexical or syntactic patterns in translated language. However, if studies of this nature ultimately give us a better understanding of how translators use language, i.e. how translators translate and what (cognitive) processes are involved, it will be of relevance, not just in the teaching of translation but also in the development of effective technological resources for translators in the future.

6. References

Baker, M. (1995) "Corpora in Translation Studies: An Overview and Some Suggestions for Future Research", *Target* 7(2): 223-243.

Baker, M. (1996) "Corpus-based Translation Studies: The Challenges that Lie ahead", in H. Somers (ed.) *Terminology, LSP and Translation: Studies in Language Engineering, in Honour of Juan C. Sager*. Amsterdam and Philadelphia, John Benjamins, 175-186.

Baker, M. (2000) "Towards a Methodology for Investigating the Style of a Literary Translator", *Target*, 12(2): 241-266.

Bell, R. T. (1991) *Translation and Translating: Theory and Practice*, London and New York: Longman.

Biber, D. (1988) *Variation across Speech and Writing*. Cambridge: CUP.

Biber, D. (1995) *Dimensions of Register Variation: A Cross-Linguistic Study*. Cambridge: CUP.

Biber, D., S. Johansson, G. Leech, S. Conrad and E. Finegan (1999) *Longman Grammar of Spoken and Written English*. London: Longman.

Blum-Kulka, S. (1986) "Shifts of Cohesion and Coherence in Translation", in J. House and S. Blum-Kulka (eds.) *Interlingual and Intercultural Communication: Discourse and Cognition in Translation and Second Language Acquisition Studies*. Tübingen: Gunter Narr, 17-35.

Burnett, S. (1999) *A Corpus-based Study of Translational English*. Manchester: unpublished MSc dissertation, UMIST.

Kenny, D. (2001) *Lexis and Creativity in Translation: A Corpus-based Study*. Manchester: St. Jerome.

Mason, I. (2001) 'Translator Behaviour and Language Usage', *Hermes* 26: 65-80.

Meta 43(4) (1998) Special Issue: The Corpus-based Approach, http://www.erudit.org/erudit/meta/

Olohan, M. (2001) "Spelling out the Optionals in Translation: A Corpus Study", *UCREL Technical Papers*, 13: 423-432.

Olohan, M. (forthcoming a) "Leave it out! Using a Comparable Corpus to Investigate Aspects of Explicitation in Translation". In *Cadernos de Tradução*, Vol. VI.

Olohan, M. (forthcoming b) "How Frequent are the Contractions? A Study of Contracted Forms in the Translational English Corpus".

Olohan, M. and M. Baker (2000) "Reporting *that* in Translated English: Evidence for Subconscious Processes of Explicitation?", *Across Languages and Cultures*, 1(2): 141-158.

Stubbs, Michael (2001) "Texts, Corpora, and Problems of Interpretation: A Response to Widdowson", *Applied Linguistics* 22(2): 149-172.

Tymoczko, Maria (1998) "Computerized Corpora and the Future of Translation Studies", *Meta* 43(4): 652-659.

Vanderauwera, R. (1985) *Dutch Novels Translated into English: The Transformation of a 'Minority' Literature*. Amsterdam: Rodopi.

Corpora in Translation Practice

Federico Zanettin

Università per Stranieri di Perugia
Palazzo Gallenga, Piazza Fortebraccio, 4 - Perugia
zanettin@unistrapg.it

Abstract

The aim of this paper is to trace links between work in the corpus linguistics community and the world of practicing translators. The relevance to translation work of corpora in general, and bilingual and parallel corpora in particular, is evaluated by comparing corpora and translation memories and by drawing an analogy between different types of corpora and more traditional reference tools, i.e. dictionaries. Corpus resources available to translators are placed along a cline going from "robust", stable corpora (e.g. large reference corpora such as the BNC) to "virtual", ephemeral corpora (e.g. DIY web corpora). Finally, a few suggestions are put forward in order to encourage a wider diffusion of corpora and concordancing software among professional translators.

1. Introduction

The translator's workplace has changed dramatically over the last ten years or so, and today the computer is undoubtedly the single most important tool of the trade for a translator regardless of whether he or she is a literary translator working for a small publisher, a technical translator working for a translation agency or a legal translator. Today, translators compose their texts on the computer screen, often receive their source texts in electronic format and sometimes their translations will only live as digital information as in the case of web site localization.

The specific hardware and software resources individual translators will resort to will vary depending on the task to be done. While in the case of most literary translators the translated text will probably take shape by means of a general purpose word processor, in the case of technical translators the target text will be produced with the help of the most sophisticated "translator workbench", equipped with all sorts of CAT tools, translation memory and terminology systems, and localization software.

The computer has also flanked, if not substituted, other technological supports in providing access to traditional tools and resources. Translation aids such as monolingual and bilingual dictionaries, terminologies and encyclopedias are now available not only on paper but also in electronic format. Colleagues and expert informants can now be consulted via e-mail and newsgroups besides via telephone, fax and face-to-face encounters. The storage capacity and processing power of personal computers have made access to linguistic and content information easier and quicker than ever before, and the Internet has opened up highways of communication and information retrieval. The problem is now not finding a piece of information, but finding the right and reliable piece of information without wasting too much time.

Corpora and concordancing software can be a way of gaining access to information about language, content, and translation practices which was hardly available to translators before the present stage of ICT development. Corpora and corpus analysis software have been around for quite a long time, but their use is only now beginning to extend beyond a restricted segment of language professionals, such as lexicographers, language engineers, as well as linguists in educational and training institutions.

I would like to suggest that corpora and concordancing software could find a larger place in the translator computerised workstation, and that more corpus resources could and should be made more accessible to professional translators. In order to do so, however, corpus builders and software producers should take into account the specific needs of this group of users. Learning to use corpora as translation resources should also be part of the curriculum of future translators and become part of their professional competence.

2. Corpora and translation

According to the EAGLES text typology elaborated by John Sinclair (1996) we can make a general distinction between Monolingual and Multilingual (including Bilingual) corpora. As regards bilingual (and multilingual) corpora a further distinction can be made between Comparable corpora (corpora compiled using similar design criteria but which are not translations) and Parallel, or Translation Corpora, which are texts in one language aligned with their translation in another. This picture can be further complicated by involving variables such as direction and directness[1] of translation, number of languages, number of translations per text, etc., producing bi-directional, reciprocal, control, star and diamond corpus models (cf. Johansson, forthcoming; Teubert, 1996; Zanettin, 2000; Malmkiaer, forthcoming). Still another type of translation related corpus is the Monolingual Comparable Corpus (Baker, 1993), or a corpus composed of two sub-sections, one of texts originally composed in one language and the other of texts translated into that same language (from a number of other languages). This type of corpus, however, while undoubtedly an extremely useful tool for translation theorists, researchers and students, is arguably of less immediate relevance for professional translators dealing with actual translation jobs.

Professional translators working in the technical sector are perhaps more familiar with the parallel concordancing feature of translator memory systems. A translation memory is data bank from which translators automatically retrieve fragments of past translations that match, totally or to a degree, a current segment to be translated, which must match, totally or to a degree - an already translated

[1] (i.e. whether a translation is produced directly from the original text or via an intermediate translation in another language).

segment. But it can also be seen as a parallel corpus which translators manually query for parallel concordances of (already translated) specific terms or patterns. Aligned translation units are conveniently displayed on the screen, offering the translator a range of similar contexts from a corpus of past translations. A translation memory is, however, a very specific type of parallel corpus in that:

a) it is "proprietory": TMs are created individually or collectively around specific translation projects. They are highly specialized and very useful when used for the translation or localization of program updates – indeed that is their origin – but are not much help when starting a new translation project on a different topic or text type.
b) TMs tend to closure, to progressively standardize and restrict the range of linguistic options. This may be an advantage from the point of view of terminological consistency and of processing costs for clients or translation agency managers, but is often detrimental for readability (texts translated using a "Workbench" can become very repetitive) and the translators eyesight (translators using a well-known Workbench often testify to a "yellow-and-blue-eye-syndrome).

Translation workbenches and translation memories have indeed become the most successful technological product to be created for professional translators, but – as it often happens with MT products – their use is best limited to specific text types, such as online help files, manuals and all types of reference work which do not require sequential reading and for which the scope of translation can be limited to the sentence of phrase level (and thus left to a machine). When dealing with other types of texts translators are perhaps better off with a different kind of language resource, i.e. the type of corpora which are more familiar to lexicographers and linguists and which are only now beginning to enter the selection of tools available to professional and trainee translators.

3. Corpora as translation aids

The respective potential uses on the part of professional translators of monolingual target corpora, bilingual comparable corpora, and of parallel corpora can be illustrated drawing an analogy with other respected tools of the trade, i.e. dictionaries: Monolingual target corpora can be compared to monolingual target language dictionaries, and comparable source corpora to monolingual source language dictionaries. While dictionaries favor a synthetic approach to lexical meaning (via a definition), corpora offer an analytic approach (via multiple contexts).[2] Translators can use target monolingual corpora alongside target monolingual dictionaries to check the meaning and usage of translation candidates in the target contexts. Like source language dictionaries, source language corpora can be consulted for source text analysis and understanding. Large reference corpora (BNC, CORIS/CODIS, etc.) can function as general dictionaries, while smaller, specialized and bilingual comparable corpora can be seen as analogous to specialized monolingual dictionaries (either or both in the source and in the target language).

Parallel corpora can instead be compared to bilingual dictionaries, with a few important differences: bilingual dictionaries are repertories of lexical equivalents (general dictionaries) or terms (specialized dictionaries and terminologies) established by dictionaries makers which are offered as translation candidates. Parallel corpora are repertoires of strategies deployed by past translators, as well as repertoires of translation equivalents. In selecting a translation equivalent from a general bilingual dictionary a translator has to assess the appropriateness of the candidate to the new context by starting from a definition and a few usage examples. A parallel corpus will offer a repertoire of translation strategies past translators have resorted to when confronted with similar problems to the ones that have prompted a search in a parallel corpus.

Parallel corpora can provide information that bilingual dictionaries do not usually contain. They can not only offer equivalence at the word level, but also non-equivalence, i.e. cases where there is no easy equivalent for words, terms or phrases across languages. A parallel corpus can provide evidence of how actual translators have dealt with this lack of direct equivalence at word level. For example, in the translations by two different Italian translators of a number of novels by Salman Rushdie (Zanettin, 2001b), the word "edges", which usually collocates with a preposition, as in the phrases "around the edges," or "at the edges," was never translated literally, but rather omitted:

1. …biting the skin around the edges of a nail…
   …*mordicchiandosi la pelle attorno all'unghia…*
2. …around the edges of Gibreel Farishta's head…
   …*intorno alla testa di Gibreel Farishta…*
3. …around the edges of the circus-ring…
   …*intorno alla pista da circo…*
4. …and there was a fluidity, an indistinctiness, at the edges of them…
   …*vicinissime a loro c'erano una fluidità e un'indeterminatezza…*
5. …the horses grew fuzzy at the edges…
   …*i cavalli diventavano sempre più sfocati…*
6. …blurred at the edges, my father…
   …*con la mente annebbiata, mio padre…*
7. …looking somewhat ragged at the edges…
   …*con l'aria di un uomo distrutto…*
8. …Mrs Qureishi, too, was beginning to fray at the edges…
   …*anche Mrs Qureishi si stava consumando…*

In all these cases, the two professional translators have consistently chosen to resort to "zero-equivalence", which being a translation strategy rather than a case of comparative linguistic knowledge would be hardly reported in any bilingual dictionary.

4. Corpus resources for translators

Not all dictionaries are the same, nor are all corpora. Apart from translation memories, corpus resources which are of potential use for professional translators could be classified along a scale which goes from "robust" to "virtual." A "corpus" is a collection of electronic texts assembled according to explicit design criteria which usually aim at representing a larger textual population. "Robust" corpora are ready-made corpora created and

[2] So-called "production dictionaries", which focus on usage information, can be thought of as standing somehow in between the two.

distributed by the research community and the language industry on CD-ROM or accessible through the Internet. Prototypical examples are large reference national corpora, such as the *British National Corpus* (BNC) for British English, and the *Dynamic Corpus or Written Italian* (CORIS/CODIS) for Italian. This type of resource, which requires a large building effort, is only now becoming available to the wider public outside the (corpus) linguistics community, and will probably require some "customisation" effort in order to become more widespread among language services providers.

Parallel corpora are usually smaller and even less available to the general public than monolingual corpora. Their construction requires more work than that of monolingual corpora. Among other factors, text pairs (rather than single texts) have to be located and before they can be used they need to be aligned, at least at the sentence level (cf. Véronis, 2000).

There are of course varying degrees of robustness, according to the effort and care which has been put in achieving a balanced and representative selection of texts, in providing explicit linguistic and extralinguistic information (corpus annotation) and the means (the software) to query the corpus for that information (McEnery & Wilson, 1996). Corpus design criteria also vary according to the purpose for which a corpus is built, e.g. a comparable monolingual corpus for descriptive translation research. In this sense, the less "robust" (i.e. the more "virtual") corpora are the most truly professional type, with reference to translators, since they are "rough-and-ready" products created for a specific translation project. A distinction is usually made by corpus linguists between "corpora" and "archives" of electronic texts. An "archive" is simply a repository of electronic texts: In this sense the WWW is an immense (multimedia) text archive. Virtual or "disposable" corpora are created by a translator using the WWW as a source "archive". The WWW and HTML documents need not to be the only source for small, specialized DIY corpora, and textual archives of various types and targeted to various users (newspapers, collections of laws, encyclopedias, etc.) are available on cd-rom. The WWW is however certainly the most familiar and user friendly environment for translators: it is always available; it is the most comprehensive source of electronic texts, and corpus creation, management and analysis can be a relatively straightforward operation (Austermühl, 2001; Zanettin, forthcoming). Building a corpus of web pages basically involves an information retrieval operation, conducted by browsing the Internet to locate relevant and reliable documents which can then be saved locally and made into a corpus to then be analysed with the help of concordancing software. The additional time required by creating and consulting a corpus is compensated for by saving in other translation-related tasks, such as dictionary consultation (both on paper and electronic), paper documentation (often in the form of "parallel texts", e.g. Williams, 1996), help from experts, and by the fact that the corpus contains information not available elsewhere. Moreover, the effort is rewarded by improving quality in terms of terminological and phraseological accuracy (Friedbichler & Friedbichler, 2000).

A number of studies have reported on experiments in translation and language teaching classes with DIY corpora, either made of "disposable" web pages (e.g. Varantola, 2000, forthcoming; Maia, 1997, 2000, forthcoming; Zanettin, forthcoming; Pearson, 2000) or of texts taken from other electronic sources such as newspapers (Zanettin, 2001a) or magazines (Bowker, 1998) on CD-ROM. Corpora created from sources other than web pages can require more time and effort to be built, and can be more or less "disposable" depending on the size of the translation project and on the resources available to create and manage them.

Reports on the use of corpora by professional translators are fewer: Friedichler & Friedbichler, drawing on their experience as translators of medical texts and trainers of technical translators, suggest that domain-specific target language corpora may usefully complement dictionaries and the Web as resources in the translation process, filling the gap between the two. Jääskläinen and Mauranen (2000) report on an experimental study involving a team of researchers from the University of Savonlinna and a team of professional translators translating for the timberwood industry. The researchers created a corpus from a variety of sources (web sites, PDF documents, etc.) following suggestions from the translators, and then trained them in using concordancing software (*WS Tools*, Scott, 1996) to analyse the corpus. In exchange, the translation team agreed to answer a questionnaire. One of the results of the study was learning that translators often complained that the user-friendliness of the concordancing software was very low. This complaint was seconded by translator trainees in other studies with "disposable" corpora where students, usually working in groups, collected a corpus of HTML documents and used them to help them translate a specific text.

These studies have underlined, nonetheless, the value of corpus building as a way of getting acquainted with the content and terminology of the translation. They have stressed the importance of type and topic of the text to be translated as well as of the target language (some text types, topics, and target languages are better helped with corpora than others) and also of adopting sound criteria in choosing suitable texts for inclusion in the corpus. Most of the corpora in these experiments were target monolingual corpora, though some use of bilingual comparable and even parallel corpora was reported.

The main benefits and shortcoming of DIY corpora may be summed up as follows:

Benefits:

- They are easy to make.
- They are a great resource for content information.
- They are a great resource for terminology and phraseology in restricted domains and topics.

Shortcomings:

- Not all topics, not all text types, not all languages are equally suitable or available.
- The relevance and reliability of documents to be included in the corpus needs to be carefully assessed.
- Existing concordancing software is not well equipped to handle HTML or XML files, i.e. web pages. There are no or few parallel corpora, since while some parallel texts (i.e. source texts + translations) can be found on the Internet, hardly all of them could be included in a parallel corpus designed to provide instances of professional standards (Maia, forthcoming).

DIY web corpora stand midway the WWW itself, which can be used as if it were a corpus and robust, "proper"

corpora. As for the Web, a "quasi-concordance" view of documents indexed and retrieved is provided by such as search engines Google (http://www.google.com) or Copernic (http://www.copernic.com). Corpus linguistics-oriented software currently being constructed for browsing the WWW as a corpus, such as *KwicFinder* (Fletcher, 2001) and *WebConc* (Kilgarriff, 2001), will certainly prove a useful tool for translators among other language professionals. However, while this "web as corpus" approach has certainly advantages in terms of time over DIY web corpora (the "corpus" is always already there), it necessarily looses in precision and reliability.

The advantages of "robust" corpora over "virtual" corpora can instead be summed up as follows:

- They are usually more reliable.
- They are usually larger.
- They may be enriched with linguistic and contextual information.
- If parallel, they are already aligned.
- They come with user-friendly, customised software (though, again, not necessarily targeted to the needs of professional translators).

5. Conclusions

Translators can tolerate the learning curve necessary to adopt corpora and concordancing software among their everyday working tools only if they derive benefits. These benefits are the fact that corpora provide information not available elsewhere at an affordable cost.

As a way of concluding, I would like to point out possible improvements for existing corpora and concordancing software:

a) "Robust " reference corpora need to become more accessible: for instance, a BNC license is still relatively expensive and the interrogation software might do with some customization; the CORIS/CODIS corpora and others have limited access.

b) In order for "virtual" corpora to become more widespread among translators, concordancing software for work with small monolingual corpora has to become capable of dealing with HTML and, increasingly, XML texts. For example, it may be useful to interface the concordancing software with the Internet browser to provide facilities for file downloading and management, and for allowing the user to switch between concordance lines and full text view, in order to take advantage of multimedia features of electronic texts.

c) Bilingual and parallel corpora are scarcely available and usually of limited size. Bilingual concordancers require bilingual corpora, and given what it takes to locate and align text pairs, it is not very likely that individual translators will resort to consulting parallel concordances unless parallel (aligned) corpora are already available. The creation of more corpora of this kind is a matter of computational resources (especially parallel concordancers and efficient aligning utilities) as well as of more awareness of the usefulness of this resource among translators and language resources providers.

6. References

Austermühl, F. (2001). *Electronic Tools for Translators*. Manchester: St Jerome.

Baker, M. (1993). "Corpus linguistics and translation studies. Implications and applications". In M. Baker, G. Francis & E. Tognini-Bonelli (eds.) *Text and technology*. Philadelphia/Amsterdam: John Benjamins, 233-252.

BNC web site, http://info.ox.ac.uk/bnc

Bowker, L. (1998). "Using specialized monolingual native-language corpora as a translation resource: a pilot study", in *META* 43:4, 631-651.

CORIS/CODIS web site, http://www.cilta.unibo.it

Fletcher, W. (2001). "Concordancing the web with KWiCFinder", presentation given at the *Third North American Symposium on Corpus Linguistics and Language Teaching*, Boston, MA, 23-25 March 2001. Available at http://miniappolis.com/KWiCFinder/Corpus2001.htm.

Friedbichler, I. & Friedbichler, M. (2000). in S. Bernardini & F. Zanettin (eds.) *I corpora nella didattica della traduzione*. *Corpus Use and Learning to Translate*, Bologna: CLUEB, 107-116.

Jääskeläinen, R. & Maurannen, A. (2000) *Work Package 5: Development of a Corpus on the Timber Industry - Final Report, Project SPIRIT MLIS-programme: MLIS-3008 SPIRIT 24637*, University of Joensuu, Savonlinna School of Translation Studies.

Johansson, S. (forthcoming). "Reflections on corpora and their uses in cross-linguistic research", in F. Zanettin, S. Bernardini, & D. Stewart (eds.) *Corpora in translator education*.

Kilgarriff, A. (2001). "Web as corpus". In P. Rayson, A. Wilson, T. McEnery, A. Hardie and S. Khoja (eds.) *Proceedings of the Corpus Linguistics 2001 conference, UCREL Technical Papers: 13.* Lancaster University, 342-344.

Maia, B. (1997). "Do-it-yourself corpora ... with a little bit of help from your friends!" in B. Lewandowska-Tomaszczyk & P. J. Melia (eds.) *PALC '97 Practical Applications in Language Corpora*. Lodz: Lodz University Press, 403-410.

Maia, B. (2000) "Making corpora: A learning process", in S. Bernardini & F. Zanettin (eds.) *I corpora nella didattica della traduzione*. *Corpus Use and Learning to Translate*, Bologna: CLUEB, 47-60.

Maia, B. (forthcoming) "Training translators in terminology and information retrieval using comparable and parallel corpora", in F. Zanettin, S. Bernardini & D. Stewart (eds.) *Corpora in translator education*.

Malmkiaer, K. (forthcoming). "On a pseudo-subversive use of corpora in translator training", in F. Zanettin, S. Bernardini & D. Stewart (eds.) *Corpora in translator education*.

McEnery, T. & Wilson, A. (1996) *Corpus linguistics*. Edimburgh: Edimburgh University Press.

Pearson, J. (2000). "Surfing the Internet: teaching students to choose their texts wisely". In Lou Burnard and Tony McEnery (eds.) *Rethinking Language Pedagogy from a Corpus Perspective*. Frankfurt am Maim et al: Peter Lang, 235-239.

Scott, M. (1996). *Wordsmith Tools*. Oxford: Oxford University Press.

Sinclair, J. McH. (1996) *EAGLES Preliminary recommendations on Corpus Typology, EAG--TCWG--CTYP/P*. Online: http://www.ilc.pi.cnr.it/EAGLES96/corpustyp/corpustyp.html.

Teuberg, W. (1996) “Comparable or parallel corpora?” *International journal of lexicography*, 9:3, 238-264.

Varantola, K. (2000). “Translators, dictionaries and text corpora” in in S. Bernardini & F. Zanettin (eds.) *I corpora nella didattica della traduzione*. *Corpus Use and Learning to Translate*, Bologna: CLUEB, 117-136.

Varantola, K. (forthcoming). “Translators and disposable corpora” in F. Zanettin, S. Bernardini & D. Stewart (eds.) *Corpora in translator education*.

Véronis, J. (2000) *Parallel text processing. Alignment and use of parallel corpora*. Dordrecht: Kluwer.

Williams, I. A. (1996) “A translator’s reference needs: Dictionaries or parallel texts”. *Target* 8, 277:299.

Zanettin, F. (2000). “Parallel corpora in translation studies: issues in corpus design and analysis”, in Olohan, M. (ed.) *Intercultural Faultlines. Research Models in Translation Studies I. Textual and cognitive aspects*, Manchester: St Jerome. 93-118.

Zanettin, F. (2001a). “Swimming in words: Corpora, translation, and language learning”, in G. Aston (ed.) *Learning with corpora*, Bologna/Houston,TX: CLUEB/Athelstan, 177-197.

Zanettin, F. (2001b). *IperGrimus*. In *inTRAlinea* (online) http://www.intralinea.it

Zanettin, F. (forthcoming). “DIY corpora. The WWW and the translator”, *Proceedings of the “Training the language services provider for the new millennium” International Conference, Porto, Portugal, 25-26 May 2001*.

BancTrad: a web interface for integrated access to parallel annotated corpora

Toni Badia, Gemma Boleda, Carme Colominas, Agnès González, Mireia Garmendia, Martí Quixal

Universitat Pompeu Fabra
Rambla 30-32 ,
E-08002 Barcelona
{toni.badia,carme.colominas,marti.quixal}@trad.upf.es, gemma.boleda@iula.upf.es

Abstract

The goal of BancTrad is to offer the possibility to access and search through (parallel) annotated corpora via the Internet. This paper presents the design of the whole process: from text compilation and processing to actually performing queries via the web, while it describes as well its technical architecture.

The languages we work with are Catalan, Spanish, English, German and French. Queries are possible from any of these languages to Spanish and Catalan and vice versa (but not between the language pairs formed by French, German and English). The texts go first through a pre-processing and mark-up stage, then through linguistic analysis and are finally formatted, indexed and made ready to be consulted. The web interface has been created through the integration some *ad hoc* applications and some ready-to-use ones. It provides three different levels of query expertise: basic, intermediate and expert.

The paper is structured as follows: section 1 gives an overview of the project; section 2 describes the text compilation process; section 3 explains the corpora building and parsing stages; section 4 details the search machine architecture; finally, section 5 describes foreseen applications of BancTrad.

1. Overview

The original idea of BancTrad[1] was to obtain a tool with pedagogic applications (see work done e.g. by Gaspari, Hansen, S.) especially thinking of translation and interpreting courses held at the Translation and Interpretation Faculty (FTI) of the University Pompeu Fabra (UPF). It was meant to be a translation databank that could serve both teachers and students to search for prototypical translations or texts containing special features that would make them interesting from the translator's point of view. Afterwards, the target user of BancTrad was broadened to e.g. professional translators and linguists (see section 5), through the creation of different search modes and the expansion of the expressiveness of the queries, in order to adapt to the user needs or knowledge.

As an annotated translation databank, BancTrad offers the possibility to work with Catalan, Spanish, English, German and French. Queries are possible from any of these languages to Spanish and Catalan and vice versa (but no queries are possible between the language pairs formed by French, German and English), as well as between Catalan and Spanish in both directions. The web page of the project can be accessed from http://glotis.upf.es/bt/index.html

2. Text collecting, extra-linguistic tagging and alignment

The corpora in BancTrad aim at being representative for translated texts. In other words, they don't have a normative character but a descriptive one. Therefore we have chosen to collect documents from very different sources, representing a variety of text types, subjects and registers.

The main sources we have focussed on are faculty professors, work done in translation courses, publishing houses and the Internet. Many faculty professors work also as freelance translators, which constitutes a good source of high quality translations. Besides, the fact that we include (supervised) work done in translation courses can have many advantages regarding academic self-evaluation. Specially, because they give evidence of the text types, subjects, etc., which have been worked on with pedagogical purposes. As for translations from the Internet, some supervision is done on them before they are selected to be introduced in BancTrad (for the sake of quality).

Selected texts are semi-automatically processed to be marked up with SGML tags and aligned with their respective original texts. Both the originals and the translations are marked up with some extra-linguistic information by means of a special MS Word form coded in Visual Basic (see Fig. 1).

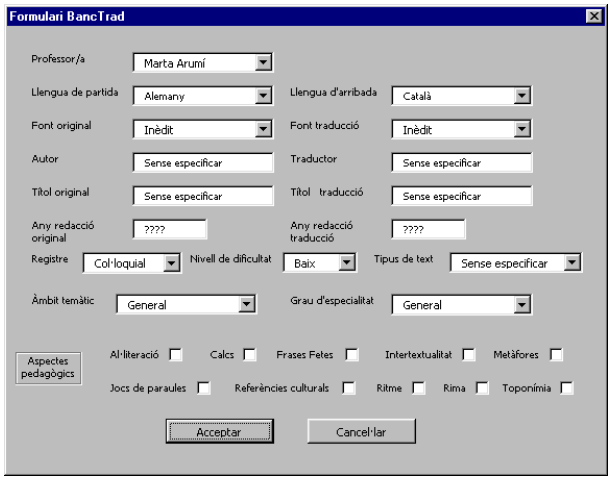


Figure 1: MS Word form used for the mark-up of extralinguistic features of the texts

[1] This project is running under the auspices of the "Programa d'Innovació Docent" (Educational Innovation Program) sponsored by our university (Universitat Pompeu Fabra) and has also been partially financed by the Spanish Government and by the 2001FI 00582 grant from the autonomous Government of Catalonia.

This mark-up takes the following parameters into account:

- name of the person who introduced the aligned texts (i. a., in order to track translation quality)
- source and target languages
- original and translation references
- publication date (for both the original and the translation)
- register (colloquial, standard, learned, etc.)
- type of text (normative, descriptive, literary, etc.)
- subject matter (economy, science, politics, etc.)
- degree of specialisation (low, middle, high).

Besides these parameters, and bearing in mind that BancTrad was originally conceived as a tool with pedagogic applications, we include information on certain aspects such as idioms, metaphors, puns, degree of difficulty, etc. All of these parameters, as well as the information coded within them, were consensuated with the teachers and researchers of the FTI. It is relevant to note that this mark-up allows us not to make a rigid classification of the texts in the corpus (see section 3).

By clicking on the *Acceptar* ("Accept") button, the options selected in the form are marked in the text in SGML format and a script tags the paragraph structure of the document. Otherwise, this very valuable piece of information on the text structure would be lost in the alignment step.

Texts are aligned at a sentence level with the align tool of the DéjàVu Database Maintenance, software by Atril (http://www.atril.com). DéjàVu aligns texts and allows editing in quite a user-friendly way.

The tasks described so far, although only semi-automatic, require neither special skills in computing nor much time (the time to go through them for a 400 word-long text -both source and target texts- is 5 to 10 minutes). We could have chosen to tackle the alignment task fully automatically instead, but the error rate of automatic aligners (notably errors in sentence identification) would have increased too much the error rate in the subsequent linguistic analysis. However, it should be kept in mind that, according to our architecture, the use of a particular tool for the mark-up and alignment independent of the rest of the process, so that other tools could be used in the future.

Finally, the texts are transferred to our Linux server to proceed with the text processing, which from this moment on will be completely automatic.

3. Linguistic Processing and Corpus Building

Once the texts are in the server, they undergo two further steps: linguistic tagging and corpus formatting. Both steps are completely automatic.

3.1. Linguistic Processing

Each language follows a different tagging process. On the one hand, Catalan texts are parsed with CATCG (Badia *et al.* 2000), a Catalan shallow morphosyntactic parser based on a constraint grammar developed by the Computational Linguistics group at UPF. Spanish texts will be handled with a Spanish version of it in a year's time. On the other hand, the linguistic analysis for English, German and French texts is made with TreeTager, a part-of-speech tagger developed at the IMS (see Schmid 1995, 1997). Both CATCG and TreeTager are shallow parsers.

La	**noia**	**de**	**el**	**port**	**de**	**Barcelona**	**dorm**
the	***girl***	***of***	***the***	***harbour***	***of***	***Barcelona***	***sleeps***

```
<s id="1">
La          el          Det         AFS         DN>
noia        noi         Nom         N5-FS       Subj
<contrac forma="del">
de          de          Prep        P           <NA
el          el          Det         AMS         DN>
</contrac>
port        port        Nom         N5-MS       <P
de          de          Prep        P           <NA
<enty>
Barcelona               Barcelona               Nom         N4G6S  <P
</enty>
dorm                    dormir      Verb        VRR2S-      VPrin
.           .           .           .           PT
</s>
```

Figure 2: Input and output of CATCG

It is important to note that, despite the use of different tagging tools for exploiting the linguistic information of our texts, all languages receive a minimum of uniform kind of information: lemma and POS tag (syntactic function is only there for Catalan). Thus, all the languages can be processed and made queries upon in the same fashion, independently of the tagging tool used. This favours modularity, for the linguistic processing of a certain language can be modified without changing any of neither the other linguistic processes nor the interface. We now proceed to roughly characterize CATCG and TreeTagger.

3.1.1. CATCG

CATCG is a linguistic-based parser that assigns each word a lemma, a POS tag and a syntactic function. It uses three major devices:

a) a Perl module for the preprocessing
b) a morphological tag mapping tool that uses a word-form dictionary created with a morphological generator developed at UPF (Badia *et al.* 1997)
c) three grammars using the Constraint Grammar formalism developed at the University of Helsinki (Karlsson et al. 1995, Tapanainen 1996), which perform the morphosyntactic disambiguation task and the partial syntactic analysis.

Fig. 2 gives an example of the input and output of our system. The SGML tags are the result of the preprocessing, and in the example they mark a contracted form, an entity and the sentence boundaries. The columns list the linguistic information: word form, lemma, part of speech tag, complete morphological information in an compressed tag and syntactic function (in order of appearance). The last piece of information is shallow and partial in the sense that it doesn't fully indicate dependency: note that the

preposition *de* (“from”) in the PP *de Barcelona* gets a tag indicating that it modifies a noun to its left (<*NA*, left adjoining Nominal Adjunct); however, no clue is given about whether it modifies *Barcelona* or *port*.

3.1.2. TreeTager

TreeTager is a probabilistic tagger that uses decision trees. It provides each word with a lemma and a POS tag (at the moment, no syntactic information is given).

3.2. Corpus formatting

After being annotated, the text files are eventually formatted and processed with the Corpus WorkBench (CWB) tools, a set of linguistic information exploitation tools developed at the IMS in Stuttgart (Christ 1994; Christ *et al*. 1999[2]). Thus we build the actual corpora making them ready to be consulted with CQP, the Corpus Query Processor, a tool from the CWB. This tool allows very flexible and expressive queries for any of the pieces of information encoded (be it the word form, lemma, POS tag or syntactic function). In fact, as a far as one gives corpora the adequate structure, one can have as a many attributes as one pleases.

One of the most significant (to us) features of the CWB is the fact that it can process aligned corpora. Not only is it possible to view the aligned sentences, but it is also possible to place restrictions both on the source and on the target language in a query (see section 5). It has also been crucial to us the special module that lets CQP interacting with the web (see next section).

4. The search machine and the web Interface

Technically speaking, the novelty of BancTrad is the integration of several tools that make available parallel annotated corpora via the Internet. This entails that the system has to be able to (1) interpret the query made by the user, (2) search for the query, (3) present the results. For this purpose, two devices were needed: a graphical user interface (GUI) with a fill-in form and an external program interface (to allow browser/server communication)

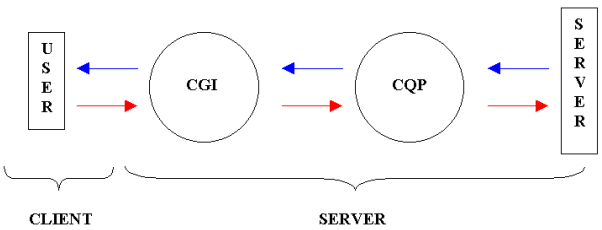


Figure 3: Query routing through the client/server architecture (query from left to right, results the other way round)

a) The GUI for query input

The GUI is intended to be adaptable to the user expertise, to have open access and to be platform independent. For our GUI to accomplish the two last features, an HTML-based interface seemed to be the best option. To qualify for the first one, the interface had to offer at least three search possibilities: common, intermediate and expert mode (see next section for details).

b) The external program interface

This is the module of the architecture that actually makes the query processing. It interprets the user's query, it searches for it in the corpora and gives the result back. The program that does the work is commonly called a cgi (Common Gateway Interface, term whose original sense has been extended to mean “external program interface”). Our cgi is composed of the following packages:

i) Common Gateway Interface (CGI)

The CGI (properly so named) is a standard device to interface with information servers (such as HTTP servers). It passes a web user's request on to an application program and gives the resulting data back to the user. Herewith the server interprets the user’s query.

ii) HTML::Entities

This formatting package ensures that special characters (tildes, cedillas, etc.) are properly transferred during the client/server session.

iii) WebCqp::Query, a web adapted version of the CQP

This package was designed by the creators of the CWB (see above) to let it interact with the web. It can perform the same kind of queries that CQP performs in its PC-Linux version. It thus allows a powerful query setting through regular expressions, access to linguistic tags (through the defined number of features in the corpora) and aligned corpus querying.

5. Exploiting BancTrad

This section outlines different ways in which to exploit BancTrad, from two different but related perspectives regarding its potential users. It describes the search possibilities that BancTrad offers (section 5.1), which relates to the user's level of expertise. Besides, it sketches some possible applications for which BancTrad is indicated (section 5.2), which relates to the user's professional or academic profile.

5.1. Search possibilities

5.1.1. Three levels of expertise

The web interface of BancTrad had to enable the users to access the corpora without having to be experts neither on linguistics nor on regular expressions. Moreover it had to offer the possibility of exploiting the full-fledged regular expressions that CQP allows, as well as the chance of profiting from the quite detailed linguistic annotation of the corpora. Therefore, BancTrad offers three different search modes (corresponding to levels of query expertise):

a) **basic mode:** allows searching for sequences of specific word forms (with possibly their equivalence in a target language).

b) **intermediate mode:** allows searching for sequences of five

[2] See also the web page of the CWB: http://www.ims.uni-stuttgart.de/projekte/CorpusWorkbench/

quadruples (form, lemma, morphosyntactic tag, and syntactic function), including the iteration of identical elements

Fig. 4 is a screenshot of a search in this mode: it searches for causative constructions from Catalan into English, that is, for the causative verb *fer* followed by any verb (see next section for the results).

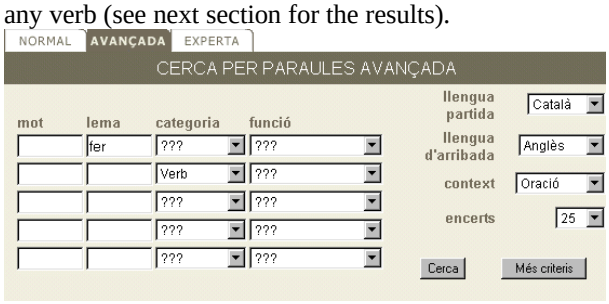


Figure 4: Screen shot of the intermediate query mode of BancTrad

expert mode: to set queries expressed in the full regular language provided by CQP.

5.1.2. Restrictions on extralinguistic features

Additionally to the word units searched for, the user can place restrictions on extra-linguistic features of the texts containing them. This is possible through the initial mark-up stage (see section 2) while formatting the corpora. Thus, through an extended web-form, the user can restrict the occurrences of e.g. the word "bank" to appear in economic texts.

This kind of mark-up gives rise to a different search possibility, planned for the original purpose of BancTrad (which was being useful for teaching purposes at the FTI): the full text query, which allows the user to search for complete texts and their translation, restricting them by the extra-linguistic features mentioned above. Fig. 5 shows a text query in which the user wants to retrieve essays (*Assaig*) on Arts originally written in German (*Alemany*) and translated into Spanish (*Castellà*).

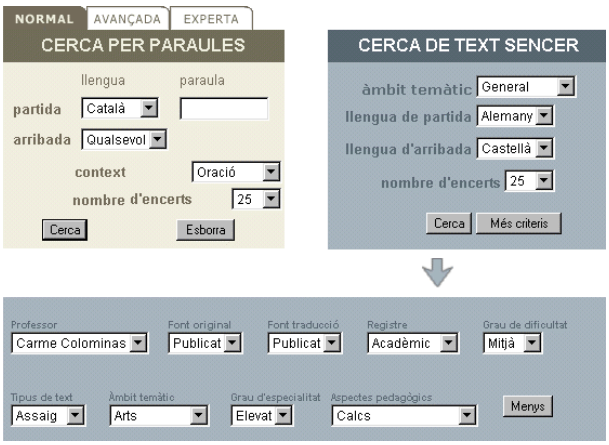
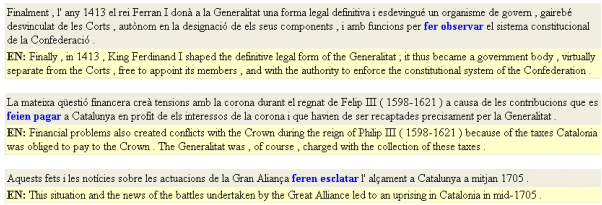


Figure 5: Screenshot of the text query mode of BancTrad

5.1.3. Showing the results

As for the presentation of the results, they are shown by default as aligned full sentences, although it is foreseen that the user can switch to other presentation forms: a full paragraph or just some words to the left and/or right sides of the query target. Of course all the capabilities listed so far are indebted to the Corpus Query Processor that we use as a searching engine.

Fig. 6 shows some of the results for the query on causative constructions made on section 5.1.1:


Finalment , l' any 1413 el rei Ferran I donà a la Generalitat una forma legal definitiva i esdevingué un organisme de govern , gairebé desvinculat de les Corts , autònom en la designació de els seus components , i amb funcions per **fer observar** el sistema constitucional de la Confederació .
EN: Finally , in 1413 , King Ferdinand I shaped the definitive legal form of the Generalitat ; it thus became a government body , virtually separate from the Corts , free to appoint its members , and with the authority to enforce the constitutional system of the Confederation .

La mateixa qüestió financera creà tensions amb la corona durant el regnat de Felip III (1598-1621) a causa de les contribucions que es **feien pagar** a Catalunya en profit de els interessos de la corona i que havien de ser recaptades precisament per la Generalitat .
EN: Financial problems also created conflicts with the Crown during the reign of Philip III (1598-1621) because of the taxes Catalonia was obliged to pay to the Crown . The Generalitat was , of course , charged with the collection of these taxes .

Aquests fets i les notícies sobre les actuacions de la Gran Aliança **feren esclatar** l' alçament a Catalunya a mitjan 1705 .
EN: This situation and the news of the battles undertaken by the Great Alliance led to an uprising in Catalonia in mid-1705 .

Figure 4: Screen shot of the intermediate query mode of BancTrad

5.2. Applications of BancTrad

There are several uses one can think of for BancTrad. Of course, the most direct and obvious one is the one for which the parallel databank was thought: educational use. But there are at least two other kinds of applications that were held in mind while developing the project: research and professional applications. The three of them are outlined, with some examples, in this section.

5.2.1. Teaching

For educational purposes, all of the search modes (be it string or text queries) outlined in the previous subsection are relevant. However, as the full text query has already been exemplified, we will concentrate on the first one. The string equivalence query, which we foresee to be the most significant application for the corpora included in BancTrad, is the search of bilingual equivalences among language pairs. This includes the search of word equivalence, restricted by its form in one of the languages, by its lemma, or by its form or lemma and its morphosyntactic tag. Thus typical searches (which demand different levels of expertise in the search mode) could be:

a) translation of the English form 'stores' into Catalan. Result: *botigues* (noun), *guarda* (verb).
b) translation of the English lemma 'store' into Catalan. Result: *botiga*, *botigues* (noun), and the whole paradigm of the verb *guardar*.
c) translation of the lemma 'store' with part-of-speech 'verb' into Catalan. Result: the whole paradigm of the verb *guardar*.

Note that as in standard corpus search engines, word forms and lemmata can be searched for in specific contexts, as well as particular combinations of forms, lemmata or part-of-speech tags. For example:

d) translation of the gerundive form of the verb 'indicate' right after a colon.

In addition, a specific search condition on the aligned text can be set. For example:

e) translation of the gerundive form of the verb 'indicate' just after a colon provided that in the translated sentence into Catalan no gerundive is present; alternatively, provided that the verb 'indicar' is used.

5.2.2. Professional and research applications

In fact, these kind of applications just follow from the examples described above and the characteristics of the corpora in BancTrad. On the one hand, as far as the corpora are real translated texts (see section 2), and provided the search possibilities sketched above, BancTrad appears to be a useful tool for professional translators. They could look for evidence of previous translation decisions and even have the information of the person in charge for that translation.

On the other hand, linguists and translation theorists (see work done by Baker, M. and Teubert, W.) could also take advantage of this search engine. In fact, this is something we have already been doing with the grammar-developing task we have been carrying on for the last three years. We can retrieve data such as most frequent readings, syntactic structures, etc. This helps us concentrate on problems arising when dealing with written text and develop more data-driven linguistic-based grammars. It is also interesting to note that searches can be made on a sole language, that is, they must not be bilingual.

Other possible applications for BancTrad include creating further Language Resources, such as multilingual dictionaries, chunkers, stochastic-based machine translation systems, etc.

5.2.3. An added value

Finally, it is important to note that an added value to BancTrad's web interface is the fact that it can incorporate other corpora (also monolingual ones) with little amount of work. This would enable our users to query on several corpora, not only the ones prepared at the FTI, in a user-friendly and familiar web interface. For instance, we already have the British National Corpus as part of our searchable corpora and we are planning to integrate the Frankfurter Rundschau corpus soon as well.

6. Conclusions and future work

We have presented a parallel-annotated corpora web interface that integrates several linguistic tools, both for exploiting linguistic information and for exploiting the linguistically enriched texts. It was originally thought to be a translation teaching help tool, but its possibilities have been so extended that it can be of use to both common public and professional users.

Technically speaking, BancTrad integrates tools from different techniques and fields. On the one hand, we use parsing tools developed at our centre, which have been developed with linguistic techniques. Moreover, we are planning to use parsers developed with stochastic techniques (TreeTagger, see above). On the other hand, we have been taking advantage of several ready-to-use packages for client/server interaction. Thus, we feel our project provides evidence of the necessity of academic co-operation to produce tools for the exploitation of linguistic information.

7. Acknowledgments

Thanks all teachers of the FTI for their collaboration. Feedback from the anonymous reviewers was also very useful.

8. References

Badia, T., À. Egea & T. Tuells (1997) CATMORF: Multi-two level steps for Catalan morphology. In *Demo Proceedings of the Conference on Applied Natural Language Processing*. Washington

Badia, T., Boleda, G., Bofias, E. & Quixal, M. (2001) A modular architecture for the processing of free text. *Proceedings of the Workshop on 'Modular Programming applied to Natural Language Processing'* at *EUROLAN 2001*. Iasi, Romania.

Christ, Oliver (1994) "A modular and flexible architecture for an integrated corpus query system", *COMPLEX'94*, Budapest

Christ, Oliver, Schulze, Bruno M. and König, Esther (1999) *Corpus Query Processor (CQP). User's Manual*, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart, Stuttgart

Karlsson, F. et al. (1995) *Constraint Grammar: a Language-Independent Formalism for Parsing Unrestricted Text*, Mouton De Gruyter: Berlin/New York

Schmid, Helmut (1995) Improvements in Part-of-Speech Tagging with an Application to German, in *Proceedings of the ACL SIGDAT-Workshop*, pp. 47-50

Schmid, Helmut (1997) Probabilistic Part-of-Speech Tagging Using Decision Trees, in Daniel Jones and Harold Somers, editors, *New Methods in Language Processing Studies in Computational Linguistics*, UCL Press, London, pp. 154-164

Tapanainen, P. (1996) *The Constraint Grammar Parser CG-2*, Department of General Linguistics, University of Helsinki, Helsinki, Publications, number 27.

ParaConc: Concordance Software for Multilingual Parallel Corpora

Michael Barlow

Rice University
Dept. of Linguistics
Houston, TX 77005
barlow@rice.edu

Abstract

Parallel concordance software provides a general purpose tool that permits a wide range of investigations of translated texts, from the analysis of bilingual terminology and phraseology to the study of alternative translations of a single text. This paper outlines the main features of a Windows concordancer, ParaConc, focussing on alignment of parallel (translated) texts, general search procedures, identification of translation equivalents, and the furnishing of basic frequency information. ParaConc accepts up to four parallel texts, which might be four different languages or an original text plus three different translations. A semi-automatic alignment utility is included in the program to prepare texts that are not already pre-aligned. Simple text searches for words or phrases can be performed and the resulting concordance lines can be sorted according to the alphabetical order of the words surrounding the searchword. More complex searches are also possible, including context searches, searches based on regular expressions, and word/part-of-speech searches (assuming that the corpus is tagged for POS). Corpus frequency and collocate frequency information can be obtained. The program includes features for highlighting potential translations, including an automatic component "Hot words," which uses frequency information to provide information about possible translations of the searchword.

Keywords: alignment, parallel texts, concordance software

ParaConc is a tool designed for linguists and other researchers who wish to work with translated texts in order to carry out contrastive language studies or to investigate the translation process itself.

1. Alignment

The successful searching and analysis of parallel texts depends on the presence of aligned text segments in each language corpus (and, of course, on the availability of parallel corpora). The alignment, an indication of equivalent text segments in the two languages, typically uses the sentence unit as the basic alignment segment, although naturally such an alignment is not one in which each sentence of Language A is always aligned with a sentence of Language B throughout the texts, since occasionally a sentence in Language A may, for example, be equivalent to two sentences in Language B, or perhaps absent from Language B altogether. (More difficult problems arise in cases where the translation of one sentence in Language A is distributed over several sentences in Language B.) The size of the aligned segments is not set by the software, however. It would be possible to work with paragraphs as the basic alignment unit, but then the results of a search will be more cumbersome because the translation of a word or phrase will be embedded within a large amount of text, which is especially difficult in cases in which the language is not well-known.

The alignment utility in *ParaConc* is semi-automatic. When files are loaded, the user enters information about the format of the files either through reference to SGML tags or via specifications of patterns. The user specifies the form of headings and the form of paragraphs. *ParaConc* uses the information to align the documents at this level and the user can make adjustments by merging/splitting units, as appropriate. Sentence level alignment, if it is not indicated by SGML tags, is performed using the Gale-Church algorithm (Gale and Church, 1993). The alignment information is saved to a file as part of the workspace, as described in Section 6.

No use is made of bilingual dictionaries or of any kind of language-particular information, but the user can enter pairs of anchors, such as cognates, numerals and dates, which the program will track. These anchors are not used in the alignment process itself, but aligned units which do not contain the appropriate corresponding anchors are highlighted for manual checking by the user.

If the parallel texts are pre-aligned, then it is simply necessary to indicate the manner in which the alignment is marked.

2. Loading the Parallel Corpus

When the LOAD CORPUS FILE(S) command is given, a dialogue box appears, enabling particular parallel files to be loaded, as shown in Figure 1.

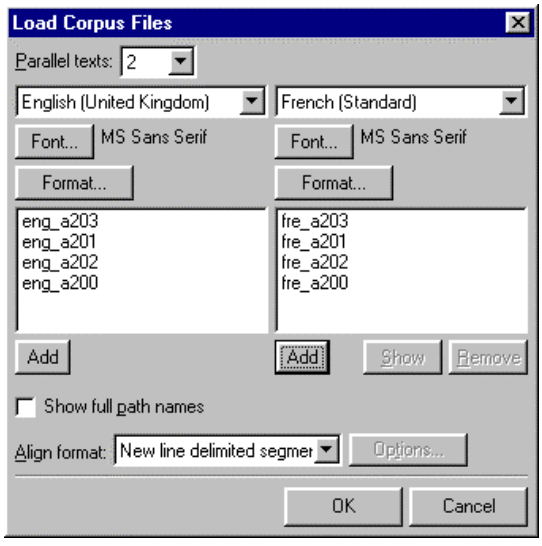


Figure 1. Loading Corpus Files

The heading PARALLEL TEXTS at the top of the dialogue box is followed by a number in the range 2-4 (i.e, two to four different languages). The FORMAT buttons allow the user to describe the form of headings, paragraphs, and sentences, as discussed above. Filenames can be reordered by dragging them to the appropriate position.

3. Searching and Analysing Parallel Texts

The program processes the files as they are loaded, counting words, recording the position of alignment indicators, and processing other format information.

Once a corpus is loaded, some new menu items related to the analysis and display of the text appear on the menu bar. These are FILE, SEARCH, FREQUENCY, and INFO. In addition we can obtain information in the lower left corner of the window relating to the number of the files loaded and in the lower right corner a word count for the two corpora is provided.

Selecting SEARCH from the SEARCH menu initiates the search process and the program starts to work though the loaded files looking for the search string. The search can be based on any of the languages represented: either English or French in this example. (The basic search is fairly simple: a word or a phrase can be entered, including simple wildcard characters if necessary. The symbols acting as wildcards are user-defined, but the default symbols are ? for one character; % for zero or one characters; and * for zero or more characters. The symbol @ covers a specified range of words. Information on the span covered by @ and other information such as a list of characters that act as word delimiters is available in SEARCH OPTIONS.)

Below the results of a search for *head* are illustrated. The instances of *head* are displayed in a KWIC format in the upper window. Clicking on one particular example of *head* in English highlights both the English and French lines. (Double-clicking on a particular line evokes a context window, which provides an enlarged context for the particular instance of the searchword.)

The lower part of the window contains the French sentences (or text segments) that are aligned with the hits displayed in the top window. This display of equivalent units in the two languages is, of course, a consequence of the alignment process. Thus if the first instance of *head* occurred in segment 342 of the English text, then the program simply throws segment 342 of the French text into the lower window, and this process is repeated for all instances of *head*.

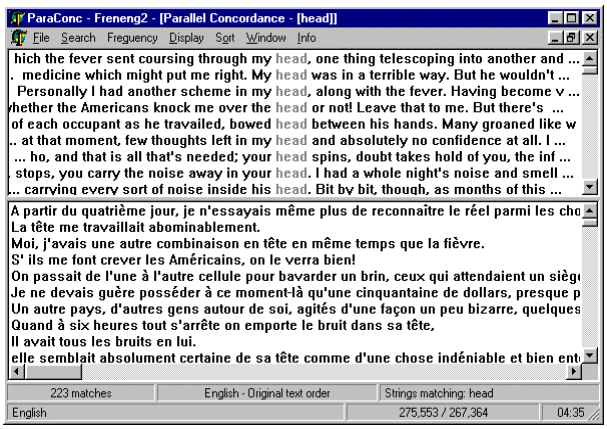


Figure 2: The Results of a Simple Search

Let's follow this example further. Once the search is ended, we can bring to bear the usual advantages of concordance software to reveal patterns in the results data. One may be interested, for example, in different uses (and translations) involving *head: big head, company head, shower head*, etc. One way to find out which English words are associated with *head* is to sort the concordance lines so that they are in alphabetical order of the word preceding the search term. The advantage of performing this 'left sort' is that the modifiers (adjectives) of *head* that are the same will occur together. One easy way to achieve this ordering is to select 1ST LEFT, 1ST RIGHT, from the SORT menu.

It can perhaps be seen from Figure 2. that while all the instances of *head* are clearly displayed, it is difficult to look through the equivalent French segments in order to locate possible French translations of *head* within each segment. To alleviate this, we can highlight suggested translations for English *head* by positioning the cursor in the lower French results window and clicking on the right mouse button. A menu pops up and we can select SEARCH QUERY which gives access to the usual search commands and hence allows us to enter a possible translation of *head* such as *tête*. The program then simply highlights all instances of *tête* in the French results window.

We can now change the context for the French results so that the results in the lower window are transformed into a KWIC layout (at least for those segments containing *tête.)* First, we make sure that the lower window is active. Next we choose CONTEXT TYPE from the DISPLAY menu and select WORDS. Finally, we rearrange the lines to bring those segments containing *tête* together at the top of the French results window. To achieve this, we choose SORT and sort the lines by searchword, and 1st left. The sorting procedure will then rearrange the results in lower window. (The SORT and DISPLAY commands are applied to whichever window is active.) The two text windows then appear as shown in Figure 3. Naturally, only those words in the French text that have been selected and highlighted can be displayed in this way. By sorting on the searchword, all the KWIC lines are grouped together at the top of the text window; the residue can be found by scrolling through towards the bottom of the window. This is a revealing display, but we have to be careful and not be misled by this dual KWIC display. There is no guarantee that for any particular line, the instance of *tête* is in fact

the translation of *head*. It could simply be accidental that *tête* is found in the French sentence corresponding to the English sentence containing *head.*

The idea behind dual KWIC display is to let the user move from English to French and back again, sorting and resorting the concordance lines, and inspecting the results to get a sense of the connections between the two languages at whatever level of granularity is relevant for a particular analysis.

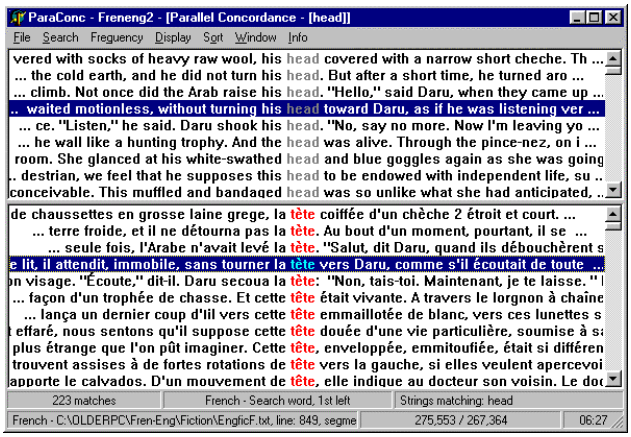


Figure 3: Parallel KWIC displays

4. Hot Words

In the previous section, we described the use of SEARCH QUERY to locate possible translations in the second window. In this section we will look at a utility in which possible translations and other associated words (collocates) are suggested by the program itself. We will refer to these words as *hot words*. First we position the cursor in the lower (French) half of the results window and click using the right mouse button. If we used SEARCH QUERY earlier, we need to select CLEAR SEARCH QUERY and then choose HOT WORDS, which invokes a procedure which calculates the frequency of all the words in the French results window and then brings up a dialogue box containing the ranked list of hot words. The ranked list of candidates for hot words based on *head* are displayed as shown in Figure 4.

To select words as hot words, the program looks at the frequency of each word in the results window and ranks the words according to the extent to which the observed frequency deviates from the expected frequency, based on the original corpus. The words at the top of the list might include translations of the searchword, translations of the collocates of the searchword, and collocations of translation of the searchword.

In addition to the basic display of hotwords, a paradigm option (if selected) promotes to a higher ranking those words whose form resembles other words in the ranked list. This is a simple attempt to deal with morphological variation without resorting to language-particular resources.

Some or all the hot words can be selected. Clicking on OK will highlight the selected words in the results window, and again the words can be sorted in various ways.

Figure 4: Hot Word List

5. Frequency information

ParaConc furnishes a variety of frequency statistics, but the two main kinds are corpus frequency and collocate frequency. The command CORPUS FREQUENCY DATA in the FREQUENCY menu creates a word list for the whole corpus (or parallel corpora), according to the settings in FREQUENCY OPTIONS. The results can be displayed in alphabetical or frequency order and the usual options (such as stop lists) are available.

Choosing COLLOCATE FREQUENCY DATA from the FREQUENCY menu displays the collocates of the search term ranked in terms of frequency. In *ParaConc,* the collocate frequency calculations are tied to a particular search word and so the frequency menu only appears once a search has been performed. The collocation data produced by the COLLOCATE FREQUENCY DATA command is organised in four columns, spanning the word positions 2nd left to 2nd right. The columns show the collocates in descending order of raw frequency.

One disadvantage of the simple collocate frequency table is that it is not possible to gauge the frequency of collocations consisting of three or more words. To calculate the frequency of three word collocations, it is necessary to choose ADVANCED COLLOCATION from the FREQUENCY menu and select one or more languages. The top part of the dialogue box associated with ADVANCED COLLOCATION allows the user to choose from up to three word positions, for example, SEARCHWORD 1ST RIGHT, 2ND RIGHT. The program counts and displays the three-word collocations based on the selected pattern.

6. Workspace

The loading and processing of a parallel corpus in particular can take some time since the program has to process alignment and annotation data before searching and analysis can begin. Since the same sets of corpus files are often loaded each time *ParaConc* is started, it makes sense to freeze the current state of the program, at will, and return to that state at any time, rather than starting *ParaConc* and reloading the parallel corpora afresh. This is the idea behind a workspace. A workspace is saved as a special (potentially large) ParaConc Workspace file (.pws), which can then be opened at any time to restore

ParaConc to its previous state, with the corpus loaded ready for searching. Searches and frequency data are, however, not included in the saved workspace. (Only the search histories are saved.)

A workspace can be saved at any time by selecting the command SAVE WORKSPACE or SAVE WORKSPACE AS from the FILE menu. The usual dialogue box appears and the name and location of the workspace file can be specified in the normal way. Once a filename for the saved workspace has been entered, the user is asked to choose some different workspace options. The line/page and the tracked tag info can be saved as part of the workspace. (The saved workspace consists of a saved file and an associated folder of the same name.)

7. Advanced Search

The simple searches described in Section 3 will suffice for many purposes and are especially useful for exploratory searches. The basic TEXT SEARCH is also very useful when used in conjunction with a sort-and-delete strategy. Particular sort configurations can be chosen to cluster unwanted examples (words preceded by *a* and *the* perhaps), which can then be selected and deleted. For more complex searches, however, we need to use the ADVANCED SEARCH command. This command brings up a more intricate dialogue box (displayed in Figure 5), which at the top contains the text box in which the search query is entered.

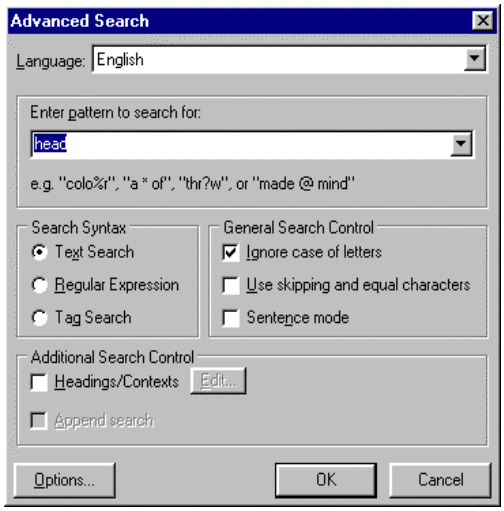


Figure 5: Advanced Search

The most important part of the ADVANCED SEARCH dialogue box is labelled SEARCH SYNTAX. The three radio buttons allow users to specify the kind of search they wish to perform. The first, TEXT SEARCH refers to the basic searches described in the section above.

The REGULAR EXPRESSION search allows for search queries containing boolean operators (AND, OR and NOT). For example, a regular expression to capture the *speak* lemma might be given as **sp[eo]a?k[se]?n?**. This expression will match the string *sp* followed by *e* or *o*, an optional *a*, a *k*., an optional *s* or *e*, followed by an optional *n*. (Word boundaries or spaces would also have to be specified in order to eliminate words such as *bespoke*.) The software also supports the expanded set of regex metacharacters: \d, \w, \s, \S, etc.

The third option in the advanced search dialogue box is TAG SEARCH, which allows the user to specify a search query consisting of a combination of words and part-of-speech tags, with the special symbol **&** being used to separate words from tags in the search query. This search syntax is used whatever particular tag symbols are used in the corpus. (Thus it is necessary to enter the form of the tags in TAG SETTINGS before a tag search can be performed.) To give an example: the search string **that&DD** finds instances of *that* tagged as a demonstrative pronoun, which may appear in the corpus as *that<w DD>*. Similarly, a tag search for **&JJ of&** will find all instances of adjectives followed by the word *off*. (The dialogue box in Figure 5 contains a variety of other options controlling the search function, which will not be discussed in this paper.)

Finally, one kind of search tailored for use with parallel texts is a parallel search, which is one of the options within the SEARCH menu. This type of search, shown in Figure 6, allows a search to be constrained based on the occurrence of particular strings in the different parallel texts.

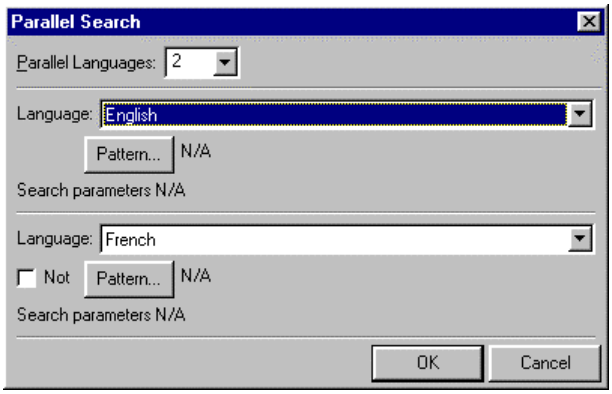


Figure 6: Parallel Search

Clicking on the Pattern box under Language: English brings up the normal advanced search dialogue box and a search query can be entered. In this case, the search term **head** has been entered. Moving to Language: French and again clicking on Pattern, it is possible to enter another search string such as **tête**. Clicking OK initiates the search routine and the software locates examples in which head occurs in the English text segment and tête is also found in the corresponding French segment. If the NOT box (under Language: French) is selected, then the search routine will display head only if tête does not occur in the equivalent French segment.

8. Summary

This paper has provided a brief overview of a Windows parallel concordance program which can be used by a variety of researchers working on the analysis of multilingual texts for translation or linguistic purposes. This article has focussed on the overall design and operation of the software and no linguistic analyses have been presented here, but the potential for cross-linguistic analyses and for the investigation of the translation process is, we hope, reasonably clear.

The main factor impinging on the usefulness of the software is probably the availability of aligned parallel corpora and of parallel corpora in general.

9. References

Gale, W. A. & Church, K. W. (1993). A program for aligning sentences in bilingual corpora. In *Computational Linguistics*, 19, 75—102.

Corpora for Terminology Extraction – the Differing Perspectives and Objectives of Researchers, Teachers and Language Services Providers

Belinda Maia
Faculdade de Letras
Universidade do Porto
Via Panorâmica s/n
4150-563 Porto
Portugal
bmaia@mail.telepac.pt

Abstract

Using corpora to find correct terminology is an activity that is interpreted rather differently according to the final objectives of those involved. This paper will try to show how the perspectives and objectives of researchers, teachers and language services providers do not always coincide, and how this lack of mutual appreciation and understanding can sometimes cause confusion. We shall first look at the more speculative aspects of current terminology research for the possibilities they offer in the future, even though some of this work is not directly related to translation, and consider the reasons why correct terminology is growing in importance in the lives of both domain specialists and language services providers. We shall then briefly consider both the older prescriptive notions of standardisation and the descriptive approach made feasible by technology and corpora today. Corpora in the broadest sense – from formally constructed and officially approved collections of texts to the disposable, do-it-yourself corpora anyone can now collect off the Internet for information on a specific subject – come as part of the information revolution provided by technology. They provide possibilities for any user of language and knowledge that were unthinkable a few years ago, but there are also problems and drawbacks.

1. Introduction

The compilation of terminology used to consist largely of collecting the words and phrases considered to be specific to a certain domain and bringing them together to form glossaries, with or without definitions or information on how or where the information was gathered. Since translators often had a vested interest in finding, or providing recognised equivalents in several languages, these glossaries would often become bi- or multilingual at a later stage. With the increase in availability of electronic text, the advantages of using corpora for term extraction are now generally recognised, particularly since the prescriptive view of terminology work has given way to a more descriptive approach, and the storage of definitions and other information on the terms has been made possible by relational databases.

This paper assumes that there are three classes of people with a particular interest in this terminology work. First there are the researchers in various areas of linguistics in general, as well as more specific terminology research. Many, but not all of these people, are also the teachers who try to train the professional language services providers needed today. The word 'linguist' as someone proficient in two or more languages has become ambiguous since the advent of 'linguistics' as an academic discipline, and the tasks required of someone with a good knowledge of languages are increasingly varied. I have therefore chosen the term 'language services provider' to refer to those who not only provide traditional translation and interpreting services, but also those who write and revise texts professionally, specialise in localisation, sub-titling, dubbing and making web pages, create terminological databases and translation memories, work with machine translation, and both use and take advantage of the information technology now available for a wide variety of projects and customers.

2. Terminology research

Those involved in this workshop on translation work and research will tend to see terminology research as primarily interested in supplying the needs of the translator for specialised terminology, but this is only one aspect of the overall picture. A good deal of terminology research is monolingual in nature and directed at the standardisation and categorisation of the relationship between concepts belonging to certain domains of knowledge and the terms used to describe them. This type of work is typically carried out by the domain experts, with or without the assistance of linguists, and, more often than not, in major languages like English, French and German. The subsequent translation of these standardised terms into other languages is by no means as simple or as well organised as it might be, despite official efforts to the contrary.

Standardisation of terminology has a long history, and its objectives have typically been to prevent confusion in the transmission of knowledge, with all the economic, social, legal and political consequences involved. Some areas of knowledge, like engineering, have a long-standing tradition in producing standardised terminology, but even they find it difficult to keep up with technical and scientific developments. Many other domains have little or no organised terminology resources and what exists is often 'local' in nature, in the sense that it is the property of certain organisations, companies and other entities, of varying size and importance.

The information revolution caused by the Internet, however, has led to demands for better systematisation of knowledge and improved accessibility. For this reason, the computational side of terminology research today is increasingly orientated towards facilitating information retrieval and knowledge engineering (see Budin, 1996, and Charlet et al, 2001). Traditional terminology work tends to be painstaking and slow, and is not adapted to coping with the exploding need for retrieving knowledge. For this reason, efforts are being made by computational linguists and computer scientists to speed up the process of identifying, extracting and processing terminology (see Bourigault et al (Eds.) 2001, and Veronis (Ed). 2000).

3. Computational terminology

So much information is now processed in computer-readable form that there are obvious advantages to be drawn from this for machine (assisted) translation, translation memories and their related terminology databases. The corpora required for this type of research need to consist of texts that are not just well written, in the sense that they represent texts normally produced in a particular domain of knowledge: they need to use terms that are generally accepted in the community that works in that domain. When translations exist of these texts, they, too, need to conform to the same standards of text and terminology in the target language if one is to produce good aligned parallel corpora.

The experimental work done in computational terminology usually involves standardised texts in which both originals and translations are considered to be of high quality. Some of these texts have been provided by organisations like XEROX (see Bourigault 1994). The texts are often chosen for their linear compatibility (See Blank, 2001), which allows for easy alignment at, at least, sentence level, and the standardisation of their technical terminology. This is understandable, since it will only be possible to proceed with the analysis of a wider variety of texts when some sort of procedure has been worked out on the basis of these controlled corpora – rather as machine translation is better at translating controlled language than Shakespeare.

There is, of course, a lot of textual material that apparently conforms to the needs of this type of research. The European Commission has worked hard at making as many of its multilingual texts available as possible. In order to do this, the translation services have effectively created enormous translation memories full of texts translated by themselves, and one can presume that the terminology used is usually supported by the EURODICAUTOM database, which is itself the result of many years of effort by a large number of people. The large multinational companies that have invested heavily in translation memory software and terminology databases could also provide a vast amount of material. Organisations like the International Standards Organisation could provide invaluable material once its standards are efficiently translated in other languages. After all, not only do these standards and their translations represent ideal parallel corpora, but the very purpose of the texts themselves is to standardise the terminology used.

4. 'Real-life' terminology

There can be no doubt that a lot of the work to which we have just referred is impressive and of high quality and, therefore, a reliable source of information for the most necessary function of all these texts – the communication of knowledge. However, anyone who has worked seriously on producing terminology with the collaboration of experts will realise that the notion of 'one concept = one term' is an ideal, not a reality. International classifications that do exist have sometimes tried to escape the problems of normal language in different ways, as when natural species are classified in Latin, or chemical and mathematical concepts use formulas and symbols.

There are various reasons why the 'one concept = one term' notion is an ideal. It is easy enough for the linguist to understand the fluidity of the lexicon. After all, one of the perennial problems of general linguistics is how to deal with it in an easily classifiable way, hence all the work with projects like Wordnet (at: http://www.cogsci.princeton.edu/~wn/). On the other hand, experts in any particular domain are also aware of the fluidity of concepts and probably spend a good deal of time arguing about how to stabilise them for practical purposes - and stable terminology is only one aspect of this problem. In practice, they often resort to diagrams, images and other pictorial representations in order to circumvent or supplement the limitations of language. The general public, however, likes to believe in the stability of both language and concepts, and, for the practical purposes of communication, we all accept that there has to be some sort of 'social contract' whereby we agree to this stability in order to understand each other.

Prescriptive terminology has usually aimed at providing this stability in an organised fashion and most specialised dictionaries and glossaries are the result. The technology of databases, however, allows for a more descriptive approach, with all the implications this has for including all the information terminologists collect in the course of their work. When one is no longer limited by space on paper – a major factor in previous lexicographical work – the prospects of including all the information available and/or prescribed by international standards for terminological databases are, to say the least, tempting. These prospects may seem unnecessary to the more immediate problems of communication, but they contribute in no small way to various visions of the systematisation and documentation of knowledge.

Terminology is not the simple accumulation of words, their equivalents in other languages, definitions and a certain amount of grammatical information. Nor is it the simple matching of term to concept. One has to deal with all the usual problems of language - social, geographical, historical, political, and other aspects of style and register. At the level of standardisation, one can even become involved in authentic battles between academics or commercial companies who want to see the words they use to describe their particular theories or products prevail.

5. 'Real-life' corpora

When one is not working for the interests of computational terminology, one will probably not have access to the type of standardised corpora already described, except for the online documentation of the European Commission. Besides this, these standardised texts, no matter how well written or translated, tend to reflect a degree of deliberate homogenisation of style and register across languages. In the more routine terminology work carried out in universities and other institutions, every terminology project will come up against a different situation, and circumstances will play an important role.

First of all, one has to find what texts are available in the domain one is studying and it is more than likely that the most important ones will not be in digital form. We have found that this is often the case when one wants to use first-class academic texts published by well-known publishers. Working with industrial or commercial institutions or companies is one way of obtaining texts, but we have not yet tried this, partly because it will

require careful negotiation, and partly because we have found several academic partners interested in cooperating on a serious and more unbiased basis.

One can always scan texts, and there are, of course, plenty of texts already in digital form. It is often easy enough to obtain permission to use these texts if one explains why one needs them and what one intends to do with them, as there is plenty of interest among domain experts to see their terminology systematized. The Internet, as we all know, can provide an enormous amount of material in certain areas, but is less useful in others. For example, we have found it of limited interest for certain engineering terminology projects because both the high level expert-to-expert type of academic article and the more didactically orientated teaching text are not freely available to the general public. Too often one ends up with commercial sites trying to sell certain types of engineering equipment, and the information thus obtained is not necessarily very reliable. In the area of population geography, however, where one is dealing with a subject that cuts across the disciplines of geography, sociology and demography, one project group was able to find a sizeable amount of material in several languages, of both a parallel and comparable nature, precisely because there are plenty of official or governmental institutions who want to publish such material on-line. The other interesting aspect of this area is that the subject is relatively new and the relative instability of the terminology was observable in the texts found.

As our projects must have a Portuguese component, one of the problems we have found is that some languages are more equal than others. If the languages involved are English, French or German, there is a chance that one will be able to find reliable texts of a parallel or comparable nature, but the same will not be true of less used languages. We have found this to be true at all levels of text we look for. We have also found that the translations of websites - whatever the original language - are often of poor quality and cannot be used as parallel corpora.

6. Teaching and Project work

The type of project work we have done over the years started as a typical translation exercise in vocabulary research that owed much of its dynamics to the fact that the translation classroom contained PCs connected to the Internet. Our curriculum had been formulated by believers in the notion that 'general translation', together with six months placement at the end of the course, was sufficient for training Modern Languages students to become translators. Our experience, and that of our graduates, soon told us that this was far from enough and we developed specialised subject project work as a way of training students in LSP (see Maia, 1997 and Maia, 2000) within the limitations of the curriculum. We have now moved on to interdisciplinary postgraduate training in terminology and translation work, working with professors from the Engineering Faculty and History and Geography departments. Our early wordlists processed in Word have now developed into more sophisticated terminology work in Excel and Multiterm, and include definitions, sources, images and other data fields. We soon hope to have our own database system and make it available online.

Corpora have always been obligatory elements of our project work but, although we have collected quite a lot of specialised mini-corpora over the years, we admit that they have not always been the most successful part of the projects. There are various reasons for this. On the one hand, perhaps the biggest enemy of terminology related corpora work is the large number of existing on-line glossaries on everything under the sun that our students soon discover from each other. One can, of course, argue that these glossaries, which are often easy to copy or download, are in themselves language resources of the type we are discussing here. However, they are usually monolingual, largely in English, often rather general in scope, and infrequently backed up by any form of official recognition. When the glossaries are good, complete, and officially recognised, adding Portuguese terminology to them is usually beyond the scope of an undergraduate project. Of course, one might argue that beginners could do worse than discover how to convert them into their own languages.

The big problem here is that such work merely encourages the idea that finding the 'right word' is enough. This means they miss out on the didactic strengths of making mini corpora - the understanding of the subject itself, brought about by having to find and read texts, the appreciation of different types and styles of text gained while doing this, and the extraction of terms in context. Although students are encouraged to use software like Wordsmith to look for keywords and to study concordances of both general language words and specialised terminology, there is always a preliminary stage when the actual reading of the texts is necessary – at least from a pedagogical point of view. If they are lucky, they will also find definitions in the texts, although these are not as frequent, or as reliable, as the literature on the subject would have us believe.

There are successful types of glossary work that do not require corpora, such as some excellent ones our students have done on tools of various types – e.g. carpentry and gardening tools - in which the 'corpora' were largely catalogues with images, and students had to work hard to make the words in both languages match the pictures provided, a process that involved plenty of questioning of individuals, but little text work.

7. Conclusions

Corpora and terminology research can work well together, but they are not always equal partners. Ideally, students should be able to find good texts and extract terms, definitions and other information from them. When mini-corpora form the basis for terminology work, the process of producing the terminology project is didactically more valuable, and it is an easy step from collecting and aligning texts, and then using concordancing, to understanding the theory behind translation memories and other software and making them work in practice. As we have said, however, valuable terminology work can be done without resort to corpora. Perhaps the most important attitude to adopt towards project work is flexibility, since each domain brings its own circumstances and problems. If at the end of the experience our undergraduate students have learned how to take special languages seriously, the main objective has been achieved. Our postgraduate students already know

how important they are and need to learn how to progress further, and perhaps even join the process of research into computational processes that will speed up the accumulation of valuable resources for all of us who do not want to see the world speaking only one language.

8. References

Austermühl, Frank, 2001 *Electronic Tools for Translators*, Manchester: St. Jerome Publishing.

Blank, I., 2001. Terminology extraction from parallel technical corpora. In D. Bourigault, C. Jacquemin and M-C. L'Homme. 237-252.

Bourigault, D., 1994. *LEXTER, un Logiciel d'Extraction de TERminologie, Application à l'acquisition des connaissances à partir de textes.* PhD thesis. Paris: École des Hautes Études en Sciences Sociales.

Bourigault, Didier, Christian Jacquemin, & Marie-Claude L'Homme, (Eds.) 2001. *Recent Advances in Computational Terminology*. Amsterdam & Philadelphia: John Benjamins Publishing Co.

Budin, G., 1996. *Wissensorganisation und Terminologie*. Tübingen: Gunter Narr.

Charlet, J., M.Zacklad G.Kassel D.Bourigault, 2001. *Ingénierie des connaissances.* Paris: Éditions Eyrolles.

Maia, B. 1997. Do-it-yourself corpora ... with a little bit of help from your friends! In Barbara Lewandowska-Tomaszczyk and Patrick James Melia (Eds.) *PALC '97 Practical Applications in Language Corpora.* Lodz: Lodz University Press. 403-410.

Maia, B. 2000, Making corpora – a learning process. In Bernardini, S. & F. Zanettin, (eds). 2000: *I corpora nella didattica della traduzione*. Bologna: CLUEB pp.47-6.

Maia, B., (forthcoming),'Comparable and parallel corpora – and their relationship to terminology work and training', paper presented at the *CULT - Corpus Use And Learning To Translate*. Bertinoro, Italy, November 3-4, 2000.

Maia, B. (forthcoming). ' Terminology – where to find it, and how to keep it', Proceedings of *III Jornadas sobre la formación del traductor e intérprete,* Universidad Europea de Madrid 7 -10 March 2001.

Veronis, Jean (Ed). 2000. Parallel Text Processing – Alignment and Use of Translation Corpora. Dordrecht: Kluwer Academic Publishers.

Wright, Sue Ellen and Gerhard Budin, 1997. Handbook of Terminology Management – Volume 1: Basic Aspects of Terminology Management. Amsterdam & Philadelphia: John Benjamins Publishing Co.

2001. Handbook of Terminology Management – Volume II: Applications-oriented Terminology Management. Amsterdam & Philadelphia: John Benjamins Publishing Co.

Working Together: A Collaborative Approach to DIY Corpora

Lynne Bowker

School of Translation and Interpretation, University of Ottawa,
70 Laurier Avenue East, Room 401, Ottawa, Ontario, K1N 6N5, Canada
lbowker@uottawa.ca

Abstract

Corpora can be invaluable resources for translation students, but creating DIY corpora on a frequent basis can be a time-consuming exercise. This paper describes an experiment whereby the students in a translation class worked in collaboration to build corpora for use in their technical translation course. The guidelines used for this collaborative approach are outlined, and the results of the experiment are discussed. A general discussion on the value of the World Wide Web as a resource for building DIY corpora is also include.

1. Introduction

Researchers such as Zanettin (1998), Yuste (2000), and Bowker and Pearson (2002) have amply demonstrated the value of using corpora as translation resources in the context of translator training. However, there are relatively few "ready-made" or "off-the-shelf" corpora available for use in specialized domains, so translator trainers and/or students typically need to construct their own. This paper outlines an experiment that was conducted with 4th-year undergraduate students in a French-to-English technical translation course. The purpose of this experiment was to see if it was possible for the class to collectively build "DIY" or "disposable corpora" (Varantola, forthcoming) that could be used as resources for their translation course work.

My previous experiments with corpus building had proceeding following either a teacher-centred approach or a learner-centred approach. Both of these approaches had a number of drawbacks. In the case of the teacher-centred approach, the translator trainer was responsible for constructing all the corpora – a job which proved to be very time consuming (resulting in relatively small corpora) and which excluded the students from the design phase of the corpus building process. In the case of the learner-centred approach, each student was individually responsible for building his or her own corpora. This approach also proved to be inefficient, with students building corpora that were often small and generally poorly designed.

It was hoped that by adopting what Kiraly (1999 and 2000) and Yuste (2001) refer to as a learning-centred and collaborative approach, the resulting corpora would be larger and more useful, and the students would engage in active discussions with the trainer and with each other and would move towards becoming empowered critical thinkers and more independent learners.

2. Setting the parameters

In order to ensure that things ran smoothly during the collaborative exercise, it was necessary to first establish a number of guidelines or ground rules. The following strategy was developed and refined based on our experience over the academic year. It addresses the following issues: a) coordinators, b) number of texts contributed by each student per corpus, c) quality of texts, d) time frame, and e) file format.

2.1. Coordinators

For each corpus, two students would act as coordinators. When students were acting as coordinators, they did not have to contribute texts to the corpus (but they still had to do the actual translation homework). Essentially, the coordinators were to act as a sort of clearing house. Students in the class would e-mail their texts to a special account set up for the coordinators, who would 1) evaluate these texts for relevance, and 2) eliminate duplications (i.e., cases where the same text had been submitted by multiple students). The remaining texts would then be collated into a single corpus that would be posted on the class Web site.

2.2. Number of texts contributed by each student per corpus

Each student (with the exception of the coordinators) would try to identify three relevant texts that would make a good addition to the corpus. Given a class of between 20 and 30 students (this class had 22 students), this number was considered to be a reasonable goal; however, it was not an absolute. If a student could only identify two suitable texts, these would still be welcome; likewise, if a student located four or five relevant texts, they could all be submitted.

2.3. Quality of texts

The students agreed to put some time and care into selecting their three texts. It was noted that if everyone were to simply submit the texts corresponding to the first three hits that came up using a Web search engine, then there would be a lot of duplication and the texts may not be pertinent, which would limit the value of the corpus.

2.4. Time frame

In order for the process to run smoothly, a reasonable amount of time had to be given for both the contributions and the coordination. It was agreed that each target text would be distributed three weeks in advance. Students would have one week to identify suitable texts and e-mail them to the coordinators. The coordinators would have one week to check the texts for relevance and for

duplication, to amalgamate the texts into a corpus, and to post this corpus on the class Web site. All the students would then have one week to consult the corpus.

2.5. File format

Students e-mailed their contributions to the coordinators as attachments in plain text (ASCII) format. This simplified the job of the coordinators as it meant that they did not have to worry about having access to different types of computers or software packages and they did not have to manipulate different file formats. It also ensured that the corpus would be in a format that could be manipulated by the corpus analysis software to which the students had access (i.e., WordSmith Tools). In addition, it reduced the chances of spreading viruses.

3. Results of the Collaborative Corpus Building Exercise

In order to give some coherence to the course, the theme of "computer security" was selected and seven different source texts – each of a different text type and each focusing on a different subject relating to computer security – were chosen. Table 1 summarizes the corresponding comparable corpora that were compiled as part of the exercise.

4. Discussion

This section will outline strategies used by the students in selecting the texts and compiling the corpora; difficulties that were encountered and solutions used to overcome them will also be discussed. In addition, some general comments will be made on the suitability of the World Wide Web as a resource for building comparable corpora. Specific details about techniques used to extract translation-related information from the corpora have been detailed elsewhere in the literature (e.g., Bowker, 2000; Bowker and Pearson, 2002) and so will not be repeated here.

The first corpus to be constructed was on the subject "passwords", and the text type was a FAQ, which is a list of *Frequently Asked Questions* (and answers) about a given subject. In total, the students submitted 58 texts for possible inclusion in the corpus; however, there was a high degree of duplication and the final corpus ended up containing only 23 texts.

A class discussion following the creation of this first corpus revealed that most students preferred to use the Web to identify comparable texts. Other resources, such as CD-ROMs and online databases, were available in the university library; however, many students had Internet access from home and found it more convenient to work from there. Their preferred method of identifying texts for inclusion in the corpus was to read the source text and then select potential subject key words to enter into a search engine. In the course of the discussion, it was revealed that most students used the Alta Vista search engine, and many of them had not been very discerning when it came to selecting the three texts that they contributed – they often simply took the first three hits that came up. In order to identify a wider selection of texts for future corpora, students agreed to make an effort to look beyond the first three hits. Moreover, we discussed the fact that different search engines index different Web sites, which means that the hits returned by one search engine may be different than those returned by another. Students agreed to use a wider range of search engines (and meta search engines) when looking for comparable texts, and it was hoped that by doing this, there would be less duplication in future corpora.

The next three corpora were intended to help translate an instructional text on "antivirus programs", a popularized informative text about "encryption", and a buyer's guide for "firewalls". In the world of computer security, these are all popular subjects and common text types, so there was a lot of information available. In particular, popularized informative texts are among the most common type of text on the Web, and many of the texts identified by the students were quite long, which elevated the word count of the encryption corpus considerably. Given that there were many texts to choose from, a number of students submitted more than three texts each. Moreover, the degree of duplication for these three corpora was reduced as a result of the students' efforts to use different search engines and to look beyond the first three hits.

The corpus on "steganography" was supposed to be used to help students translate a product description. Steganography is much less common than other security measures and there are a limited number of products on the market. Consequently, the students found that there were fewer texts to choose from with the result that only 35 texts were submitted, and of these, only 14 were retained. Of the texts that were rejected, many were duplicates; however, the coordinators also rejected a number of texts that were not of an appropriate text type. Given the relative scarcity of comparable texts, some of the students had submitted texts that were about steganography, but which were not product descriptions. Similar behaviour has been observed by Pearson (2000), who notes that translation students sometimes show poor judgment when sourcing terminology and phraseology from comparable texts. For example, they are often primarily concerned with identifying texts that deal with the subject matter in question, but they do not ensure that the texts they choose are comparable to the source text with respect to its other features, such as register, technicality and text type. In a class discussion, the matter was raised and it was emphasized that in order for a text to be "comparable", it had to take into account text type as well as subject matter.

The source text on biometrics was an extract from a research article. There were 29 comparable texts submitted, but only 12 of these were retained. However, since research articles tend to be long, the word count was still reasonably high. The main problem that the students had was in finding the relevant text type on the Web. Although there were a number of hits that looked promising, many of these links led to Web sites that required a paid subscription in order to gain full access to the contents of the site (e.g., online journals). This led to a discussion about other non-Web resources that may be useful for building corpora, including the *Computer Select* CD-ROM, INSPEC abstracts and a variety of online

journals that were part of the university's library collection. It was noted that although students would rather work from home (hence their preference for consulting the Web rather than the library databases), it was not unreasonable to expect them to make a trip to the library in order to consult more appropriate resources.

Subject	Text type	Texts submitted	Texts rejected	Number of texts / words in corpus
Passwords	FAQ Web page	58	35	23 texts / 40,600 words
Antivirus programs	Instructional	78	22	56 texts / 170,919 words
Encryption	Informative/popularized	74	19	55 texts / 216,522 words
Firewalls	Buyer's guide	63	18	45 texts / 136,017 words
Steganography	Product description	35	21	14 texts / 7,401 words
Biometrics	Research article	29	17	12 texts / 69,651 words
Cookies	Technical encyclopedia entry	41	19	22 texts / 11,754 words

Table 1: A brief description of the corpora produced as part of the collaborative corpus building exercise.

Finally, the source text on "cookies" consisted of an entry taken from a technical encyclopedia. Once again, there were relatively few submissions (41 texts), coupled with a high degree of duplication (only 22 texts were retained). This was because there are a limited number of electronic technical encyclopedias that could serve as comparable texts. Furthermore, it was observed that the entries in such encyclopedias tend to consist of short texts, which resulted in a relatively low word count for the corpus as a whole.

5. General observations about using the Web as a resource for building DIY corpora

In addition to discussing particular problems that came up when creating specific corpora, the class also discussed a number of more general points, many of which concerned the nature of the Web and its suitability as a resource for building DIY translation corpora. For example, it was noted that there are many texts on the Web that are of poor quality and which therefore do not make good translation resources. When asked to reflect on potential reasons for this poor quality, students came up with the following possibilities. Firstly, they noted that anyone can post information on the Web, including non-subject field experts and non-native speakers, and that Web documents are not always subject to an editing process in the same way that printed documents usually are. Furthermore, the Web is seen by many as an ephemeral resource; people are interested in communicating information, but unlike the case with printed documents, this information may not be preserved for long (i.e., a Web page can be revised, updated or removed very easily) and so people are less willing to invest much time or effort in formulating that information. In other words, many people feel that a Web page does not need to be elegant (or even grammatically correct!) as long as it adequately conveys the essential information.

Another comment focused on the types of texts that are commonly found on the Web. Given that the Web is most often used as a means of disseminating information to a non-expert audience, it contains primarily informative or instructional texts that are popularized. More specialized material and different text types can be accessed via the Web, but such information is often available only by paid subscription. This means that while the Web can a valuable resource for constructing corpora that deal with popularized informative texts, it may prove less helpful for constructing corpora that must comprise other types of texts.

A similar observation was made about the languages of texts available on the Web. The students in this class were attempting to compile comparable corpora containing English-language texts, of which there are many on the Web; however, they noted that for translators working in less widely-used languages, there may be fewer texts available (at least for the present, though hopefully this will change over time).

The very nature of the Web gave rise to two other observations. Firstly, the idea behind hypertext is that people can jump from page to page to view associated information. Good Web design dictates that there should be a limited amount of information on each page so that people are not required to scroll unnecessarily; related pieces of information should be provided on separate pages with relevant links between them. When compiling a corpus from the Web, each page must be copied/saved separately and then later amalgamated into a corpus. Therefore, from a corpus builder's point of view, it would be preferable to have a single page containing a lot of information, as this page could be copied/saved in one operation, rather than having that same information spread over several pages, which would then need to be copied/saved separately. This basically means that good Web design is not conducive to easy corpus building! Secondly, the multimedia nature of the Web is another characteristic that is not always conducive to building text-based corpora. On a number of occasions, students rejected Web pages that would have been extremely useful sources of information but which could not easily be incorporated into a text-based corpus because their primary value resided in their graphical or audio content. This raises an important point: a corpus can be an invaluable resource, but it is not a panacea. There are many other complementary types of resources that can

also provide helpful information, and these should not be ignored.

Finally, the sheer volume of information that is available on the Web made students aware of the importance of formulating search queries carefully in order to be able to focus in on relevant material. As previously mentioned, students tended to read the source text first in order to get ideas for potential key words. These words were then entered into a search engine, and the resulting hits were examined for relevancy as well as for ideas for other key words that could be used for further searches. In addition to key words that dealt with the subject matter, students also found that it could be useful to enter key words relating to the text type. For instance, a search using only the subject key word "cookie" returned many irrelevant texts such as recipes; however, a more carefully formulated search that combined subject and text type key words, such as *+cookie +computer +encyclopedia*, returned hits for entries for "cookie" in resources such as *The Grand Encyclopedia of Computer Terminology*, *TechEncyclopedia* and *PC Webopedia*. Other tricks, such as remembering to search for alternate spellings (e.g., encyclopedia/encyclopaedia) also helped to increase the number of relevant hits. In addition, as mentioned previously, the students also found it useful to conduct a search using a variety of different search engines or a meta-search engine. Bergeron and Larsson (1999) provide additional tips for effective Internet search strategies for translators.

6. Concluding Remarks

Overall, the collaborative corpus building exercise proved to be a worthwhile experience. The students demonstrated that they were eminently capable of working together to construct valuable translation resources, which they could then consult to identify relevant lexical, phraseological and stylistic information. Not surprisingly, of the seven collective corpora that were built, the larger ones, such as those on antivirus programs and encryption, tended to contain a greater number of examples. Of more interest, however, is the fact that even the small corpora, such as those on steganography and cookies, contained useful information. This supports the point made by researchers such as Rogers and Ahmad (1994), who note that when working in specialized fields, it is not necessary to have the sort of multimillion word corpora that are typically required for general language work.

In addition to furnishing students with an opportunity to explore the merit of corpora as translation resources, this exercise also provided a valuable opportunity for a shift in pedagogical strategy. The collaborative corpus building exercise made it relatively easy for the trainer to take on the role of facilitator (rather than information provider), which in turn allowed the students to become independent learners and critical thinkers, who were encouraged to reflect on the characteristics of different text types and on the suitability of the World Wide Web as a translation resource. Acting as both contributors and coordinators, students learned to identify relevant features of texts and to be more discerning with regard to the appropriateness of a text (e.g., in terms of quality, text type, nature) for use as a resource for the translation at hand.

7. Acknowledgements

The work described here has been partially funded by grants awarded to Lynne Bowker by the Faculty of Arts of the University of Ottawa and the University of Ottawa Research Fund.

8. References

Bergeron, Manon and Susan Larsson, 1999. Internet Search Strategies for Translators. *The ATA Chronicle* 28(7): 22-25.

Bowker, Lynne, 2000. Towards a Methodology for Exploiting Specialized Target Language Corpora as Translation Resources. *International Journal of Corpus Linguistics* 5(1): 17-52.

Bowker, Lynne and Jennifer Pearson, 2002. *Working with Specialized Language: A Practical Guide to Using Corpora*. London: Routledge.

Kiraly, Don, 1999. From Teacher-centered to Learning-centered classrooms in translator education: control, chaos or collaboration? In *Innovation in Translator and Interpreter Training (ITIT)* – an online symposium held from January 17-25, 2000) http://www.fut.es/~apym/symp/kiraly.html

Kiraly, Don, 2000. *A Social Constructivist Approach to Translator Education*. Manchester: St. Jerome.

Pearson, Jennifer, 2000. Surfing the Internet: Teaching Students to Choose their Texts Wisely. In L. Burnard and T. McEnery (eds), *Rethinking Language Pedagogy from a Corpus Perspective*. Frankfurt: Peter Lang.

Rogers, Margaret and Khurshid Ahmad, 1994. Computerised Terminology for Translators: The Role of Text. In M. Brekke, O. Andersen, T. Dahl and J. Myking (eds), *Applications and Implications of Current LSP Research, Vol. II*. Norway: Fagbokforlaget.

Varantola, Krista, Forthcoming. Translators and disposable corpora. In F. Zanettin, S. Bernardini and D. Stewart (eds), *Corpora in Translator Education*, Manchester: St. Jerome.

WordSmith Tools: http://www1.oup.co.uk/elt/catalogue/Multimedia/WordSmithTools3.0/download.html

Yuste, Elia. 2000. Translation Instruction in the Y2K – Electronic Corpora, Internet and Translation Technology. In *CD-ROM Proceedings of the Seventh Conference of the International Society for the Study of European Ideas (ISSEI 2000)*, Workshop 501 - Teaching Translation in the Information Age. University of Bergen, Norway.

Yuste, Elia. 2001. Technology-Aided Translation Training. *Hieronymous* (3). Bern, Switzerland: ASTTI.

Zanettin, Federico. 1998. Bilingual Comparable Corpora and the Training of Translators. In *Meta* 43(4), 616-630.

Language resources and the language professional

Elia Yuste

Computerlinguistik (CL)
Institut für Informatik (IfI) der Universität Zürich
Winterthurerstrasse 190, CH-8057 ZÜRICH, Switzerland
yuste@ifi.unizh.ch

Abstract

This paper aims at raising awareness about electronic language resources (henceforth LR) in the translation community at large. Examining how technological advances in the profession have transformed the notion of translating itself and what is expected from a qualified translator today, the paper goes on to focus on resources, rather than tools. It then discusses what type of LR should feature in the training of professional translators, and how these should be tackled in various translation-training settings. It contains several useful pointers throughout the article and an extensive bibliography covering the various issues addressed herewith.

Keywords: translation profession, language professional, qualified translator, translation training, tools, resourceful, resources, language resources (LR), corpora, translation technology and HLT, academic training, vocational training, collaborative approach, real-life scenarios, translation workflow, multi-user access, corporate language, content management, resource creation / maintenance / evaluation / validation / exchange, exchange standards

1. Introduction

Traditionally speaking, translation has been regarded as a craft, a fairly unusual gift that, for some, did not even require formal academic training, let alone continuous education on (technological) advancements in the profession. From that standpoint, the translator's major asset, and only utensil, is his or her own competence for translating, that is, some special ability to transpose meaning from one language to another. But even if natural linguistic talent is always desirable, translators cannot solely rely on it to succeed as language professionals today. Translating has become a complex and permeable professional activity, which among other things requires plenty of intercultural sensitivity and disposition to adapt to new work patterns.

In fact, professional and qualified translators (against the unqualified intruders that slip in the translation profession) do usually gain respect and recognition (and in practical terms, are more employable) for being *resourceful* and acquainted with the *tools of the trade*. But what do we mean by 'resourceful' here? 'Resourceful' in that they are expected to be capable of resolving linguistic problems (and/or cultural misinterpretations) efficiently and at once? Or perhaps, 'resourceful' in that they ought to be familiar with resources that allow them to find the right information at a mouse click? What 'tools of the trade' do we refer to? Are commercial translation memory[1] packages the hot tools for translators, the one and only?

1.1. Tools …AND resources, please!

Up to the late 20th century's information revolution, heavily characterized by the advent of the personal computer (PC), the so-called ICT[2], and the Internet, the translator's *tools* had primarily been pen and paper (without forgetting about the now old typewriter and the Dictaphone®). Of course, *paper* understood in its broad sense (different sizes, textures, colours…) as a means to **manually** catalogue, archive and, hopefully, retrieve – throughout the years – translation notes, bibliographical references, and laborious samples of terminographic work. Undoubtedly, these were extremely valuable (and praiseworthy) *self-made resources* under a not very convenient support.

Other conventional translators' *resources*, linguistic and non-linguistic, consist of printed dictionaries and reference materials (such as voluminous encyclopedias –now online[3]), as well as certain cultural and/or domain-specific knowledge, gradually acquired through reading, visits to libraries, travelling, life experience and, sometimes, long discussions with fellow translators and subject experts over a cup of coffee.

Although the latter still works for some translators to some extent, the newer generations normally resort to other (quick-access) information sources and data processing applications, usually computer (e.g. on CD-ROM or DVD) or Web based, in order to accomplish their translations. Not surprisingly, 'tools' and 'resources' often get listed as *useful links* in Web sites and other publications for the translator, without making much of a distinction between them. However, I would like to see these two concepts differentiated (despite their undeniable affinity – and even interdependence[4] – in today's translation workflow),

[1] *Translation tools* have become the buzzword in translation educational and work contexts. By and large, they are usually identified with translation memory (TM) packages, the apparently ideal solution for a cost-effective and consistent translation. Yet, apart from these tools managing and reusing previously translated repetitive input, translators also ought to get to know about tools that allow them to create, retrieve, exploit, interconnect, and exchange…*language resources* (LR) – simply because LR are their most precious resources.

[2] Acronym of 'Information and Communication Technologies'.

[3] E.g. *Encyclopaedia Britannica* (http://britannica.com).

[4] If a translator uses a terminology management program to manage their terminology records, then the program itself would be the *tool* whereas the resulting records would be the

since this paper will concentrate upon *resources*, rather than *tools*.

In essence, *tools* should refer to those instruments or equipment (e.g. ball-pen, computer, printer, software program, etc.) that translators use in their daily work or that they need for a particular job assignment (e.g. a concordancer[5] for automatic term extraction, the comment utility of a word-processing software for proof-reading, etc). But equally important are *resources* (e.g. corpora, dictionaries and reference materials, glossaries and terminological databases, etc.), especially *language resources* (henceforth LR), since these are useful elements in the translation process and contribute to enhancing the translator's professional profile.

1.2. LR and HLT applications – something to equip the language professional, too

Moreover, in the area of HLT[6], where translation technology indisputably has its place, LR can be essential *components*. Without them, many research and real life systems would not see the light. Godfrey, J. J. and A. Zampolli (1996) thus define LR as '…(usually large) sets of language data and descriptions in machine readable form, […] used in building, improving, or evaluating natural language (NL) and speech algorithms or systems. Examples of linguistic resources are written and spoken corpora, lexical databases, grammars, and terminologies, although the term may be extended to include basic software tools for the preparation, collection, management, or use of other resources.'

Apart from offering us an overview of LR, Godfrey & Zampolli stress the fact[7] that LR may be extended and used to elaborate other resources, then including or interacting with tools. This is an important aspect for translation work and research. LR are usually conceived with a purpose in mind, but they may serve other purposes later, by being expanded, tailored to the needs of another user-group or integrated in a system. For instance, a paper-based glossary is linguistically-enriched (i.e. annotated or marked-up) and transformed into electronic form to become available in an organization's intranet; a few navigation and edition tools are added to allow for rapid cross-referencing and regular content updates. Some time later, this and other LR are part of a new terminological workbench, also accessed by translators and domain expert validators working for the same organization. Since the time this resource gets digitized, its lifecycle varies dramatically according to its functions and targeted user-groups.

The *resourceful* language professional[8], interested in the advances of the profession, should thus be able to create, use, and evaluate those LR serving their job or area of specialisation needs better. Translation training programmes should then prioritize topics related to LR creation, manipulation, and evaluation.

2. Goal of the paper

This paper therefore aims at discussing the importance of resources, in particular LR, shaping every facet of translation (the training of translators, the profession itself, translation as part of global content production, etc.). Ideally, our translator will be conceived as an eclectically evolving, and qualified language professional, rather than as a word artist exclusively.

3. LR in the training of translators

In order to response to revolutionized translation work patterns, most translation training institutions have incorporated some technology-related elements within their syllabuses, but it still remains unclear whether they are sufficient and efficient enough. Whereas at the beginning much emphasis was given to introductory modules on IT[9], most recently some commercial translation memory packages seem to be getting all the attention.

In 1999, the LETRAC[10] commission reported that in the surveyed translation training institutions[11] across Europe, 'LE/IT [not expliciting the concept of LR, though] in translator curricula vary from nothing but basics in word processing to a broad range of sophisticated software tools (terminology management, translation memory, machine translation, Telecommunications / Internet, CD-ROM-based information systems...).' Also of interest are their

resource. But, obviously, given this interdependence between tool and resource, one might argue that there is a very fine line between the two.

[5] A *concordancer* is a software application aimed at retrieving *concordances* (an automatic display of a word or phrase occurrence/s, known as KWIC – key word in context, surrounded by left and/or right accompanying words) from a text or corpus previously loaded. As this tool allows for rapid linguistic insight of any word, it is of great value for the linguist, lexicographer, or translator. This is why most translator workbenches include now a concordancer among their growing panoply of utilities.

[6] Acronym of 'Human Language Technologies'.

[7] Fact also reported by OVUM (1995): 'In order to provide users with a working system adapted to their environments, […] linguistic resources may also include the ability to create other bi-lingual, multi-lingual or reversible dictionaries to provide terminology quickly in other language pairs'. The potential multi-user access is also highlighted.

[8] The term *language professional* is normally applied to translators, who do not perceive their professional activity restricted to translation in its traditional sense. It may also be applied to other professionals working with language, such as terminologists, proof-readers, cross-cultural multilingual advisers, content managers, etc. They all are key *language industry* players.

[9] Acronym of 'Information Technology'.

[10] LETRAC - *Language Engineering for Translators Curricula*. EU-funded research project that run from 1998 to 1999, whose aim was to survey best practices in the training of translators enhanced by language engineering (LE) components.
http://www.iai.uni-sb.de/LETRAC/home.html

[11] Reuther, U. (ed.). April 1999. 'LETRAC survey findings in the Educational Context', Deliverable D1.2.

observations[12] on how (the type of) training has a determining effect on translators' professional success:

- 'A translator does not only perform translation.
- Training in IT should be obligatory.
- Translators do not feel well prepared by their institutions for the real world of work.
- Translators gained their present LE/IT knowledge mainly from work experience, by means of "learning by doing".
- Among freelancers, two extremes can be observed: those translators who follow the principle *as little IT as possible*, and those who can cope with virtually all aspects of new technologies. The latter are those who do better economically.
- Most translated texts are LSP[13] texts; therefore specialised translation and terminology should be an essential element in curricula.
- There is a lack of qualified IT-specialists on the translation market. Translators with LE/IT-skills have far better professional prospects.'

These reflections show the big challenge for translation training institutions posed by global language market needs, described by Shreve (1998:5) as 'an evolution in fast-forward', highly dominated by the areas of multilingual technical communication and software/web localization.

3.1. LR in academic training

Plenty of translation scholars and researchers have advocated the use of corpora in the classroom, presented them as invaluable analytical resources in TS[14] (among others, see Austermühl 2001, chapter #8, Baker 1992/3/6/9, Bernardini & Zanettin 2000, Bowker 2000a/b and 2001, Kenny 1998, Laviosa 1997, Pearson 1996/8, and 2000, Tognini-Bonelli 2000, Ulrych 1997, Yuste 2000/1, Zanettin 2000/1 and forthc., as well as Hansen and Teich, Olohan, Zanettin, Maia, and Bowker, in order of appearance in this vol.), and also created tools for their exploitation or access (see Badia et al., and Barlow, this vol.). However, it appears that a generalized systematic inclusion of LR, mainly corpora, in translation training curricula still remains a necessity (Yuste, forthcoming), especially in places where English is not an official or a tuition language.

It is beyond the scope of this paper to advocate again for corpora in translation training scenarios. Yet, it is relevant to bear in mind that translators 'need, above all, to acquire a sound knowledge of the raw material with which they work: to understand what language is and how it comes to function for its users' (Baker, 1992: 4). This is better achieved through meaningful training activities whereby future translators look into authentic (against pre-fabricated) language instances in context. Besides, corpora allow the translator trainer to keep a steady balance between theoretical linguistic insights and practical applications.

Most importantly, one should not forget that many aspects of corpus linguistics (e.g. concordancing, alignment, parallel corpora) are present in current and future language/translation technology applications. Future translators should be made aware of the fact that the commercial TM package available in their lab contains such and such corpus linguistics features. It is only when modern tools for the translator are presented comprehensively and, if necessary, theoretically backed-up, that the translator can fully understand the mechanisms behind the tool. He or she is then also empowered to make the most out of the tools or applications at hand.

Tools such as translation and localization workbenches, knowledge and content management systems, to name but a few, are usually solutions which get constantly fed with linguistic data, i.e. LR such as corpora. Under such circumstances, it is important to promote research linked to market needs, e.g. fostering of LR *exchange*[15] *standards* or *reusability* (see Kübler, this vol.). An ideal first step is to get future language professionals involved in the creation and maintenance of resources, such as (DIY) corpora (see Zanettin forthc. and Zanettin, Maia, and Bowker, this vol).

In this line of work, it is important to follow *collaborative* (see Kiraly 1999/2000 and Yuste 1999/2001) and *project-based* training approaches, whereby future translators do not only learn about how to create or exploit shared LR but also get used to teamwork, project management, etc. – skills so highly appreciated in corporate and institution settings where cross-site language work is crucial.

3.2. LR in vocational or continuous training

Most technology-related vocational or continuous training courses on offer for future and practicing translators deal with TM systems or localization tools, sometimes with little reference to LR, such as corpora. Software tools producers (usually their marketing-oriented training departments), translation training academic departments (often postgraduate course modules devoted to translation technology, which may be opened to an external audience), and occasionally translators' societies or bodies organize these courses, whose training quality and price can vary considerably.

Their merit is mainly to aim at compensating for the lack of up-to-date technology-aided translation training in formal academic settings. These courses, heavily market-oriented, should nevertheless employ solid

[12] Reuther, U. (ed.). April 1999. 'LETRAC survey findings in the Industrial Context', Deliverable D2.2.

[13] Acronym of 'Language for Specific Purposes'.

[14] Acronym of 'Translation Studies'. Note that the impact of corpora in TS has lead to *Corpus-based Translation Studies* (CTS), with M. Baker as one of the main precursors.

[15] In that corpora, terminologies, language ontologies, output from TM systems, etc. may represent valuable LR not for the resource creator or first intended user-group only, LR have to be conformant to formats so that they can be exchanged, made accessible to other user-groups or integrated into other applications. For more information on recently agreed standards, such as TMX and TBX, see specifications drawn from the SALT Initiative and Abaitua (2001), Budin et al (1999), Budin & Melby (2000), Budin (2002), and Zerfass (this vol.).

training principles (see previous section) and real-life application scenarios (i.e. full description of interrelated components, usefulness of the tool within overall workflow, satisfaction and benefits for the translator, etc. instead of a mere exposition of reduction of costs).

3.2.1. Training on LR at the workplace

When the course takes place at the workplace, it is of utmost importance to analyze what the needs for LR (and any form of translation/language technology) are, not only for translators or linguists, but also for other staff members, such as resource evaluators and domain experts.

Similarly, it is advisable to look at LR from the *corporate language* (or even institution-wide language) perspective, and see how they may contribute to optimizing (global) multilingual documentation production. For example, to learn how to create corporate databases (product names, enterprise-wide terminology) helps reinforce a company's image, promoting clear, consistent communication and aiding cross-cultural understanding. *Controlled language* (see Fankhauser 2000) schemes (e.g. multilingual corporate style guides for written documents of all kinds) and *content management* (see Budin, this vol.) strategies may have to be implemented.

Finally, similar initiatives/solutions developed by other language industry players and organizations of the same sector will have to be carefully examined, not to reinvent the wheel. Ideally, language professionals (and other LR user-groups) will have to be able to maintain, customize and tailor existing LR, as budget controls may prevent them to create their own. Sharing and exchanging LR with other partners will be essential, and so it will be training focused upon LR exchange standards (see footnote #15).

4. Conclusion

Despite the length limit of the paper, we have attempted to discuss the relevance of language resources (LR) for the translator and the rapidly evolving translation profession in a comprehensive and up-to-date manner.

LR are crucial to transform the qualified translator into a *resourceful* language professional, able to respond to any challenge, and enhance any translation-related workflow. But, of course, nothing of this is possible without adequate and tailored LR training be it in an academic setting or at the workplace.

5. References

Abaitua, J. 2000. Tratamiento de corpora bilingües. Paper presented at the *La ingeniería lingüística en la sociedad de la información* Seminar, hosted by the Fundación Duques de Soria, Soria, Spain. 17th - 21st July 2000. http://www.serv-inf.deusto.es/abaitua/konzeptu/ta/soria00.htm and http://www.serv-inf.deusto.es/abaitua/konzeptu/ta/sorefs00.htm (paper references).

Abaitua, J. 2001. Memorias de traducción en TMX compartidas por Internet. In *Revista Tradumática*. Número 0 – October 2001. http://www.fti.uab.es/tradumatica/revista

Austermühl, F. 2001. *Electronic Tools for Translators*. Translation Practices Explained series (A. Pym, series editor). Manchester: St. Jerome.

Baker, M. 1992. *In other words – a coursebook on translation.* London / New York: Routledge.

Baker, M. 1993. Corpus Linguistics and Translation Studies: Implications and Applications. In M. Baker, G. Francis and E. Tognini-Bonelli (eds). *Text and Technology: In Honour of John Sinclair*. Amsterdam / Philadelphia: John Benjamins, 233-250.

Baker, M. 1995. Corpora in Translation Studies. An Overview and Suggestions for Future Research. In *Target* 7(2): 223-43.

Baker, M. 1996. Corpus-based Translation Studies: The Challenges that Lie Ahead. In Sommers, H. (ed.). *Terminology, LSP and Translation: Studies in Language Engineering in Honour of Juan C. Sager*, Amsterdam / Philadelphia: John Benjamins.

Baker, M. (ed.). 1998. *Routledge Encyclopedia of Translation Studies*. London / New York: Routledge.

Baker, M. 1999. The Role of Corpora in Investigating the Linguistic Behaviour of Professional Translators. In *International Journal of Corpus Linguistics* 4(2): 281-298.

Bernardini, S. & F. Zanettin (eds.). 2000. *I corpora nella didattica della traduzione. Corpus Use and Learning to Translate*. Bologna, Italy: CLUEB.

Bowker, L. 2000a. The translator as LSP learner: Using an electronic LSP corpus as a translation resource. In M. Ruane and D.P. Ó Baoill (eds). *Integrating Theory and Practice in LSP and LAP*. Dublin: IRAAL, 85-91.

Bowker, L. 2000b. A Corpus-based Approach to Evaluating Student Translations. In *The Translator*, Vol. 6(2), 183-210.

Bowker, L. 2001. Towards a Methodology for a Corpus-Based Approach to Translation Evaluation. In *Meta*, Vol. 46(2), 345-364.

Budin, G. et al. 1999. Integrating Translation Technologies Using SALT. In *Proceedings of the 21st International Conference on Translating and the Computer*. London, 10th – 11th November. London: ASLIB.

Budin, G. and A. Melby. 2000. Accessibility of Multilingual Terminological Resources – Current Problems and Prospects for the Future. In Zampolli et al. *Proceedings of LREC*. Athens, June 2000. 837 ff.

Budin, G. 2002. Der Zugang zu mehrsprachigen terminologischen Ressourcen – Probleme und Lösungsmöglichkeiten. In Mayer, F., K.-D. Schmitz, and J. Zeumer (eds.). *eTerminology* – Akten des Symposions. Deutscher Terminologie Tag e.V. Cologne, 12th – 13th April, 2002.

Chriss, R. 2000. *Translation as a Profession*. Available online at:

http://www.foreignword.com/Articles/Rogers/default.htm

Esselink, B. 2000. *A Practical Guide to Localization*. Amsterdam / Philadelphia: John Benjamins.

Godfrey, J. J. and A. Zampolli. 1996. Overview [of Language Resources]. Subsection 12.1 of Chapter 12, *Language Resources*, by Cole, R. (ed.). In G. B. Varile and A. Zampolli (managing eds.). *Survey of the State of the Art in Human Language Technology*. Sponsored by the National Science Foundation and the European Commission. http://cslu.cse.ogi.edu/HLTsurvey/HLTsurvey.html

Fankhauser, R. 2000. Corporate Language Management. Paper presented at the *tecom Schweiz* Seminar with the same title, organised by Iris Jahnke (Trados Switzerland). Hotel Mövenpick, Zurich. 31st August, 2000. Paper available in German from *tecom Schweiz* (Swiss Society for Technical Communication) at: http://www.tecom.ch ('Publikationen' section).

Kenny, D. 1998. Corpora in Translation Studies. In Baker, M. (ed) *Routledge Encyclopedia of Translation Studies*, 50-53.

Kiraly, D. 1999. From teacher-centered to learning-centered classrooms in translator education: Control, chaos or collaboration? In *Innovation in Translation and Interpreting Training - ITIT*, an online symposium (17th – 25th January, 2000) organised by Anthony Pym. Intercultural Studies Group. Universitat Rovira i Virgili, Tarragona, Spain. http://www.fut.es/~apym/symp/kiraly.html

Kiraly, D. 2000. *A Social Constructivist Approach to Translator Education – Empowerment from Theory to Practice*. Manchester: St. Jerome.

Laviosa, S. 1997. How Comparable Can 'Comparable Corpora' Be? In *Target* 9(2): 289-319.

OVUM Report. 1995. Mason, J. and A. Rinsche. *Translation Technology Products*. OVUM Ltd., London.

Pearson, J. 1996. Electronic texts and concordances in the translation classroom. In *Teanga* 16. Dublin: IRAAL. 86-96.

Pearson, J. 1998. Teaching terminology using electronic resources. In S. Botley, A. McEnery and A. Wilson (eds.) *Multilingual Corpora in Teaching and Research*. Amsterdam: Rodopi. 92-105.

Pearson, J. 2000. Surfing the Internet: Teaching students to choose their texts wisely. In Burnard, L. and McEnery, T. (eds). *Rethinking language pedagogy from a corpus perspective*. Papers from the 3rd. International Conference on Teaching and Language Corpora. (Lodz Studies in Language). Hamburg: Peter Lang, 2000.

Robinson, D. 1997. *Becoming a Translator – An Accelerated Course*. London / New York: Routledge.

Sager, J.C. 1994. *Language Engineering and Translation: Consequences of Automation*. Amsterdam / Philadelphia: John Benjamins.

SALT: Standards-based Access service to multilingual Lexicons and Terminologies. More information about this project is available at the TTT – Translation, Theory and Technology site (http://www.ttt.org/salt/), hosted by A. Melby, and at the project's site (http://www.loria.fr/projets/SALT/)

Shreve, G. M. 1998. The Ecology of the Language Industry: Prospects and Problems. Keynote Address of the Language in Business / Language as Business Conference. Institute for Applied Linguistics, Kent State University, Kent, Ohio, USA. October 8, 1998. Available as a .pdf document at: http//appling.kent.edu/ResourcePages/ConferencesandWorkshopsPast/LanguageinBusiness/Thursday/OI-Shreve.PDF

Theologitis, D. 2000. Language Technology in EU Institutions. In Proceedings of the EAMT Machine Translation Workshop "Harvesting existing resources". Ljubljiana, Slovenia. May 2000. Online presentation at: http://www.eamt.org/archive/ljubljana/Theologitis.ppt

Tognini Bonelli, E. 2000. *Corpus Linguistics at Work*. Amsterdam / Philadelphia: John Benjamins.

Yuste, E. 2000. Translation instruction in the Y2K - Electronic Corpora, Internet and Translation Instruction. In CD-ROM *Proceedings of the 7th Conference of the International Society for the Study of European Ideas (ISSEI)*. Section V, Workshop No. 501 "Teaching Translation at the Information Age". Bergen, Norway: University of Bergen HIT Centre, 14-18 August 2000.

Yuste, E. 2001. Technology-aided translation training. In *Hieronymous* (3/2001) Bern: Switzerland: ASTTI (Swiss Association of Translators, Terminologists, and Interpreters).

Yuste, E. forthcoming. *Translation technology understanding and use in Switzerland*. Internal project report.

Ulrych, M. 1997. The impact of multilingual parallel concordancing on translation. In Lewandowska-Tomaszczyk, B. and P. J. Melia (eds) *PALC '97. Practical Applications in Language Corpora*, Lodz: Lodz University Press, 421-436.

Zanettin, F. 2000. Parallel Corpora in Translation Studies. In Olohan, M. *Intercultural faultlines*. Manchester: St. Jerome. 105-118.

Zanettin, F. 2001. Swimming in Words: Corpora, Translation, and Language Learning. In Aston, G. (ed). *Learning with corpora*. Houston, TX: Athelstan, 177-197.

Zanettin, F. forthcoming. DIY corpora: the WWW and the translator. In *Proceedings of the Conference on Training the Language Services Provider for the New Millenium.* Porto. May 2001. http://www.federicozanettin.net/DIYcorpora.htm

NOTE.

All online bibliographical references were last ckecked in April, 02.

Textual and terminological bridgeheads for traversing the language gap

Marita Kristiansen, Magnar Brekke

Norwegian School of Economics
and Business Administration
Department of Languages, Helleveien 30, N-5045 Bergen, NORWAY
Marita.Kristiansen@nhh.no Magnar.Brekke@nhh.no

Abstract

We describe here the basic modules of a concept-oriented bilingual text-and-term-based knowledge management system (KB-NHH) to which students, teachers, researchers, domain experts, terminologists, linguists, translators and writers of various categories can turn for content learning, reference and documentation. The aim is to ensure that the interface between English and Norwegian is being handled with efficiency and consistency.

Primary user context of the implementation described here is an on-campus e-learning system.The aim is to facilitate the representation, learning, teaching and dissemination of relevant domain knowledge, to monitor changes in and development of the subdomain languages and to document all through authentic citations. Conceptual linkage of terms and authentic segments in the text bank allow source inspection and evaluation by user. Focus is on corpus-based term extraction, definitions, terminological representations, Norwegian-English equivalence problems and contrastive phraseology.

This paper makes a distinct contribution by proposing the integration of a conceptual knowledge-base with the textual manifestation of its underlying domain knowledge and its terminological representation in one or more languages, all in the context of a standard e-learning system. This should greatly facilitate learning by bridging the language gap experienced by native and non-native students alike in approaching a new knowledge domain.

1. The general problem

Communication in very specific domains of activity is crucially dependent on possession of specific domain knowledge and mastery of the specific domain language through which such knowledge is conventionally represented and transmitted. Whereas translation of general text between two national languages remains a general challenge for both human and machine translation, the translation of special domain text presupposes far greater proficiency in handling the content and representation of that domain knowledge.

Thus translation work undertaken along the interface between two special domain languages, each of which being entrenched in its respective national language, puts heavy demands on the translator's ability to control content and expression on both sides of the gap. Similarly a student of a specific domain faced with teaching or textbooks in a foreign language will have a dual problem: He or she will need to connect the technical terms and concepts encountered on the far side to equivalent concepts and terms on the near side, which in principle involves learning new content also in the native language. The need for a content and terminology management system at this point should be obvious, while the practical solution is not.

2. Specific obstacles

The potential problems arising in the contrastive language situation just described can be further aggravated if there are marked asymmetries between the two languages involved. In the domains referred to above English tends to be the source language for the overwhelming majority of communication involving bilingual text and terminology, and the pace at which new concepts and terms are created and disseminated can be quite hectic. This puts under considerable pressure a number of "lesser spoken languages", and particularly their cultural resilience and readiness for "terminological self defense". This makes it all the more important to compensate for the asymmetry by providing efficient and user friendly tools for managing the relevant language resources.

Fortunately the rapid development of information technology has placed tools within our reach which may enable even a lesser-spoken language such as Norwegian with 4.5m speakers to cope, partly at least, with such major challenges. We will describe here the basic modules of a concept-oriented text-and-term-based knowledge management system (KB-NHH) to which students, domain experts, terminologists, linguists, translators and writers of various categories can turn for content learning, reference and documentation. The aim is to ensure that the interface with English is being handled with efficiency and consistency.

The project described here is being developed in the context of the TERMINEC project, a three-year effort to establish the foundations of such a resource database for Norwegian and English special language as used in economic-administrative domains. What follows below is a description of a particular implementation of tools for bilingual data capture, terminology handling and application in a research and teaching environment dependent on economic-administrative communication. Due to space limitations the focus will be on modules involved in a web-based e-learning system.

3. The building blocks[1]

3.1. Data capture.

The foundations of the TERMINEC database are being implemented in the form of two parallel text corpora, one English, one Norwegian, of representative text from about

[1] "Modules" referred to in this section are shown in appended diagram

30 economic-administrative subdomains (see table 1), and a parallel term database whose contents are largely being derived from and dynamically linked to the text corpora.

One of the modules is thus a textbank (module 4 in appended diagram), a representative corpus of indexed full texts in the chief genres associated with didactic, expository and popularizing text types drawn from textual representations of the universe of subdomain knowledge (module 1).

Accounting and Costing, Capital markets, Corporate analysis, Corporate strategy and Ethics, Economic geography, Economic history, Economics, Economy systems and management, Finance, Investment, Information systems and management, Law (Corporate law, EU/EEA law, Tax law, etc), Macroeconomics, Management, Market communication, Market economics, Market research, Marketing, Mathematics and statistics, Microeconomics, Organization and management, Organizational behavior, Public economy, Quality management, etc.

Table 1: Tentative economic-administrative domains/subdomains

Typologically the text bank will contain English and Norwegian parallel texts both in the sense that they are original texts which share subdomain and genre, as well as in the sense of aligned translation pairs of source text and target text, which will increase the research value of the collections considerably.

3.2. Knowledge representation.

Terminological research is normally based on the onomasiological principle, the grouping of terms according to their conceptual meaning. Thus any knowledge subdomain can be characterized by a (partially) structured set of basic concepts which are represented linguistically through domain-focal terms (cf. Brekke, 2000). Establishing or extending conceptual systems (cf. module 8) becomes essential in achieving authentic representations of the knowledge which constitutes a given subdomain. This activity presupposes close cooperation between a domain expert and a trained terminologist (cf. "module" 2 & 3) in identifying and delimiting what the basic concepts are, conventional term usage, acceptable synonymy etc. The repository for their work is a termbank (cf. module 9) holding terminological units defined, classified as to subdomains, and mapped to their respective key concepts and conceptual hierarchies in module 8. Using the concept as a term record pivot (as is done in e.g. Trados MultiTerm, which is employed in the pilot project) facilitates the inclusion of other language equivalents (French, German and Spanish are obvious candidates for inclusion later on).

3.3. Term extraction (cf. module 5).

The slow time-honored techniques of "excerption" has long since been supplemented by increasingly sophisticated computational methods. Many of the results are impressive but have not allowed us to dispense entirely with the services of the domain expert in tandem with the terminologist. Given that the selection of input texts has yielded a representative corpus, it remains a sampling and thus very far from being exhaustive of the knowledge constituting a given subdomain. The problem is twofold: On the one hand, any automatically generated list of term candidates (cf. module 6) will reflect massive overgeneration of spurious combinations which will need to be pruned. On the other, no automatic term extractor will point out which basic terms are NOT represented in the sampled text, which takes an alert and knowledgeable human being.

The TERMINEC pilot project has allowed room for experimentation along these lines using SystemQuirk's suite of terminology tools. The point of departure is frequency lists followed up by selective concordance work. A typical subcorpus (of about 17000 words) yields the following (table 2):

90 internet	41 countries	31 investment
81 america	40 prices	30 companies
73 new	39 funds	30 capital
71 economy	37 years	30 fund
63 growth	37 economic	28 markets
60 productivity	32 technology	28 japan
45 firms	32 high	26 big
44 business	32 year	26 commerce
44 investors	32 risk	26 shares
41 market	31 share	26 pension

Table 2: Top of System Quirk's standard frequency list.

Some of these one-word units of fairly general scope can be identified as Economics terms, which is useful but of limited value. SystemQuirk provides two different functions for enhancing frequency lists to improve on our term enquiry.

3.3.1. Weirdness.

SQ exploits a "weirdness"-function based on a comparative ratio which expresses the likely occurrence of a given item in the text being scrutinized compared to the same for a large general corpus. Where the latter occurrence is zero the ratio will of course be infinite, indicating either a typo, a nonce word, or in fact a technical term, which is also indicated by a very high ratio. As a result a number of items occurring only once in a given text will be brought to the top of the frequency list, and such lists usually give significant inputs to the ensuing frequency studies. Table 3 (over) reveals a typical situation. It should be noted in table 3 that of the top 30 items on the list, 2/3 of them occur only once, which would effectively drown them out of the investigator's attention had not the "weirdness"-function been active (cp table 2).

While both tables contain terms which are immediately recognizable by an economist they only share one (*investment*), and those on the "Weirdness"-list are clearly of a more specific domain-related scope (and presumably less recognizable by a nonexpert). Table 3 has

12 inf!-terms, i.e. items not occurring in a large corpus of general English, while the remainder occur between 151 and 3 times more often than they would in that corpus. Thus their degree of specialization is approaching general usage.

3.3.2. Terms as strings of content words.

The other tool offered by SQ for sniffing out potential multi-word terms, aptly named Ferret, is based on a very simple algorithm: It takes a general list of function words as boundary signals and proceeds to identify any string of content words uninterrupted by such boundary signals as a term candidate. Table 4 displays the results obtained from examining the same text as above.

Frq	Match	SL/GL Ratio
10	capital markets	inf!
3	business cycle	inf!
2	annual report	inf!
2	central bank	inf!
1	new york stock exchange	inf!
1	dow jones industrial average	inf!
1	cost of capital	inf!
1	capital stock	inf!
1	european union	inf!
1	institutional investor	inf!
1	balance sheet	inf!
1	fiscal policy	inf!
1	solvency	151.2382
6	equity	88.5297
1	annuity	75.6191
1	takeover	75.6191
1	futures	50.4127
31	investment	35.1849
2	premium	32.7001
2	inventory	31.8396
1	liquidity	30.2476
1	diversification	27.4978
1	downstream	19.5146
1	revenues	10.2534
3	yield	8.0303
4	bond	7.8565
1	float	6.8745
1	commodity	5.5500
1	options	4.4482
1	margin	3.2700

Table 3: Top of System Quirk's frequency list with "weirdness"-function active.

For reasons which are unclear Ferret missed two of the occurrences of *capital markets*, and it does seem to invite some obvious refinements of its list of boundary signals, but otherwise the high end of the frequency list does throw up some promising term candidates.

3.3.3. Equivalence checking: Plugging the terminological holes.

In economic domains the terminological pressure from English has increased in proportion to the rapid globalization processes seen through the nineties and continuing unabated, while the readiness to invest in professional means for handling the textual interface has been lacking. Most of the recent efforts have gone into developing a speech interface, and the systematic monitoring and creation of suitable terminology for use in translating economic texts has been left to private initiative. Some subdomains thus appear well looked after, while others tend to end up with haphazard and ad hoc equivalents for newly formed concepts and terms from English-speaking

8 capital markets	7 pension funds	7 mutual funds
5 see chart	5 less than	5 life insurers
5 past decade	5 information technology	4 institutional investors
4 share prices	4 s economy	3 this year
3 on average	3 recent years	3 since america
3 point out	3 but there	3 this survey
3 other countries	3 retail sales	3 but even
3 cost savings	3 poorest countries	3 world bank
3 foreign aid	3 emerging economies	3 supply chain
3 short term	3 b2b e	3 s gdp
3 hedge funds	3 state street	3 an annual average

Table 4: Ferreted strings

cultures. Since the two languages have very close historical and lexical affinities, one should not be surprised to encounter a variety of terminological misfits, from simple (and humorous) "folk translations" through cognate shifts to serious "false friends" which may create hazardous and expensive mistakes.

Cognates constitute a rich quarry for terminological misfits. Consider the following examples:

*1. Federal Reserve Bank of Minneapolis President Gary Stern warned on Friday against the ``**moral hazard**'' that may prompt banks to undertake too much risk amid excessive confidence of government safety nets.*

Anyone connected professionally with hedging and insurance will recognize the special term (in bold). While each member of the phrase has a cognate with several

meanings in Norwegian, it is rather obvious that the connotations they bring along are quite different from the English ones. Nevertheless the temptation to use the "direct method" is clearly irresistible, as the following sample (from a sizable collection) will show:

*2. Kombinasjonen av usikrede lokale banker, **moralsk hasard** i utlandet, av kortsiktige utenlandske kapitalplasseringer og Pengefondets innstrammingspolitikk, ga kraftige negative utslag.*

A linguistically sensitive person familiar with the concept underlying the original expression in 1 (including their use as separate English words) will realize that the "calque" in 2 creates undesirable connotations. Unfortunately many will fail to see the problem, which allows the emergence of a Norwenglish (quasi-Norwegian) terminology lacking professional and cultural quality assurance. Arriving at the Norwegian equivalent "**åtferdsrisiko**" requires professional handling, time, and relevant domain knowledge (another subdomain prefers "**subjektiv risiko**"). It also requires an efficient dissemination channel to ensure its adoption and use.

Equivalence checking is thus serious and important business for anyone purporting to traverse the knowledge gap as well as the language gap through translation or related forms of text production. It appears to be one stage of the bridge building which cannot easily dispense with the bilingual human expert/terminologist or their term creation principles, be they linguistically, politically or culturally motivated. In other words, the bridge heads on either side must be anchored in their respective professional context, and the quality of work assured through a content and terminology management system. Only then can our efficient computer-based tools for processing and dissemination come into their own.

4. Dissemination and use.

At the outset the material held in the KB-NHH database will form the basis for student oriented bilingual domain glossaries with definitions, as well as genre-related material for learning and teaching. Both textbank and termbank will be SGML conformant, adhering as far as possible to the TEI guidelines, which allows interactive access via a Web-browser or ftp downloading. In addition all or parts of the termbank may be distributed on CD-ROM. Printed versions are possible, but the main emphasis will be on interactive use via electronic networks. This will take full advantage of the dynamic aspects of electronic media, allowing e.g. fuzzy matching of any search to the nearest form.

The diagram referred to in Appendix outlines the current architecture of KN-NHH, a "proof-of-concept" implementation of the e-learning oriented application of TERMINEC. The student enters the e-learning system (cf. module 11, a "Blackboard"-type system) via a standard web-browser (cf. module 10), accesses the course catalog and proceeds to the description/presentation of the course content in either English or Norwegian. All domain focal terms have active links to the central conceptual system. At this point the student may follow the link to the relevant term record in the desired source language, study definitions etc. and go from there into the text bank to inspect authentic text samples illustrating usage, phraseology etc. This is particularly useful for a non-native student. Alternatively the student may proceed directly from conceptual system to the text samples, and from there via clickable text-embedded terms across to the full term-bank representation of the desired concepts to study definitions, synonyms, acronyms etc.

Students approaching a new knowledge universe will easily detect concepts not adequately covered or explained. All searches will be logged to allow a study of user behavior and user needs, with a view to enhancing the intuitiveness of the user interface. Following an unsuccessful search the user will be asked (through automatic routines) to report unfound terms and submit a relevant text segment with source reference, and will have a chance to include responses or comments. It will be considered whether users also should be invited to join an «official» discussion group. Success in engaging the user in such dynamic interaction will not only provide a way of monitoring a continuous growth of the collection but may also create greater user identification with the KB-NHH, which in turn may have a standardizing effect.

5. Maintenance and development.

New concepts are constantly being created in the professional community and migrate towards general usage, sometimes even grabbing front page headlines: *unit-link*, *derivatives* and *hedge funds* have recently enjoyed such instant attention. At the time of writing *e-business* is very much in vogue (along with almost any noun with an *e-* prefix), and *creative accounting* is already a cliché in the financial headlines.

This implies that simply registering the constitutive concepts of a given domain, including their manifestation through the terminology of one or more national languages, is not done once and for all. What is required is a more or less continual monitoring of the entire life cycle of any given term, from creation through extension and expansion to disuse and eventual death. The above are random examples of an ongoing process which is in fact quite normal, although the speed and intensity may vary with the times and the subdomain. Ideally the new or altered terms would need to be absorbed by writers, their underlying concepts defined and systematized by domain experts and terminologists, standardized by professional bodies, and their usage documented through carefully vetted citations. At the receiving end of this process would be speakers of other languages (be they experts, journalists or textbook authors) who would ideally have to establish procedures for finding or creating equivalent terms and determine proper usage.

6. Outlook

This paper makes a distinct contribution by proposing the integration of a conceptual knowledge-base with the textual manifestation of its underlying domain knowledge and its terminological representation in one or more languages, all in the context of a standard e-learning system. This should greatly facilitate learning by bridging the language gap experienced by native and non-native

students alike in approaching a new knowledge domain. A well documented and web-accessible clearinghouse for English-Norwegian economics text and terminology as envisaged here would also establish a significant point of reference for empirically based term-formation and possibly standardization, thus providing Norwegian export-oriented corporations with a much needed quality assurance of the linguistic interface. The same would hold for Norway's administrative and political cooperation with the outside world, as well as for the global language industry, which depends on the availability of multilingual databases and some form of translation. The realism in trying to stem the flood of English usage in conducting the professional affairs of people whose normal mode of communication is something other than English is highly debatable, but the virtue of avoiding linguistic domain losses in Norwegian is not.

7. References

Ahmad, K. & M. Rogers (1994). "Computerised terminology for translators: the role of text", in Brekke, Andersen, Dahl & Myking (eds) *Applications and implications of currenct LSP research*. Bergen: Fagbokforlaget.

Brekke, M, J. Myking og K. Ahmad (1996). "Terminology Management and Lesser-Used Living Languages: A Critique of the Corpus-Based Approach", in *Proceedings of 4th International Congress on Terminology and Knowledge Engineering* (TKE '96). Vienna: Indeks Verlag.

Brekke, M. (1998) "When «Empiry» strikes back: A Corporal Confrontation". *Proceedings from Workshop on Adapting Lexical and Corpus Resources*, First International Conference on Language Resources and Evaluation, Granada, Spain.

Brekke, M. (1999). "TERMINEC: The dual linkage of text and terminology", in *Proceedings of 5th International Congress on Terminology and Knowledge Engineering* (TKE '99). Vienna: Indeks Verlag.

Brekke, M. (2000). "On the Lexical Identification of Domain Focal Text and Terminology". *Proceedings of COMLEX 2000*, University of Patras, Greece.

Brekke, M. (forthcoming) "TERMINEC. A Clearinghouse for Economics Text and Terminology", to be published in Proceedings of ICAME 2000, Sydney, Australia, April 2000.

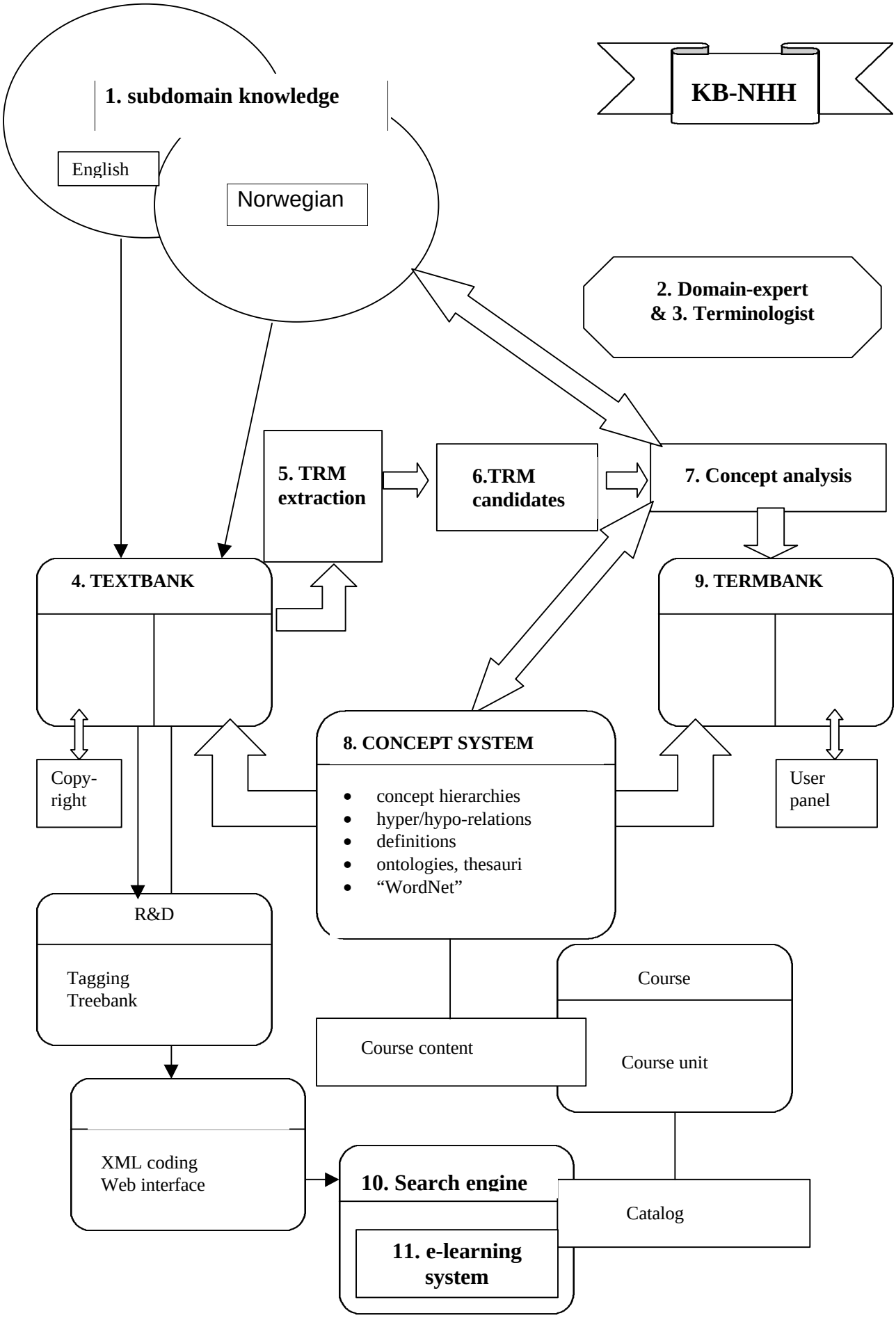
Church, K. et al. (1991). "Using statistics in lexical analysis". In Zernik, U. (ed.) *Lexical Acquistion: Exploiting on-line resources to build a lexicon.* Hillsdale, New Jersey: Lawrence Erlbaum Associates.

Lebart, L., A. Salem and L. Berry (1998). *Exploring textual data*. Dordrecht: Kluwer Academic Publishers.

Oakes, M. (1998). *Statistics for corpus linguistics*. Edinburgh: EUP.

SystemQuirk: http://www.mcs.surrey.ac.uk/Research/CS/AI/SystemQ

Appendix: **Outline of the KB-NHH architecture** (see next page)


1. subdomain knowledge
English
Norwegian
KB-NHH
2. Domain-expert
& 3. Terminologist
5. TRM extraction
6.TRM candidates
7. Concept analysis
4. TEXTBANK
9. TERMBANK
8. CONCEPT SYSTEM
concept hierarchies
hyper/hypo-relations
definitions
ontologies, thesauri
"WordNet"
Copy-right
User panel
R&D
Tagging
Treebank
Course
Course content
Course unit
XML coding
Web interface
10. Search engine
Catalog
11. e-learning system

Creating a Term Base to Customise an MT System: Reusability of Resources and Tools from the Translator's Point of View

Natalie Kübler

Intercultural Centre for Studies in Lexicology
University Paris 7
2, Place Jussieu, 75251 Paris Cédex 05, France
kubler@ccr.jussieu.fr

Abstract

This paper addresses the issue of combining existing tools and resources to customise dictionaries used for machine translation (MT) with a view to providing technical translators with an effective time-saving tool. It is based on the hypothesis that customising MT systems can be achieved using unsophisticated tools, so that the system can produce output of sufficient quality for post-translation proofreading. Corpora collected for a different purpose, together with existing on-line glossaries, can be reused or reapplied to build a bigger term base. The Systran customisable on-line MT system (Systranet) is tested on technical documents (the Linux operating system HOWTOs), without any specialised dictionary. Customised dictionaries, existing glossaries completed by adding corpus-based information using terminology extraction tools, are then incorporated into the system and an improved translation is produced. The dictionary will be augmented and corrected as long as modifications generate significant results. This process will be described in detail. The resulting translation is good enough to warrant proofreading in the normal way. This last point is important because MT results require specialised editing procedures. Compared with the time taken to produce a translation manually, this methodology should prove useful for professional translators.

1. Introduction

The growth in the volume of documentation for translation and the constant enhancement of tools have brought about great changes in the world of translation. Corpus linguistics has opened up new perspectives for both translation studies and the process of translating. As Baker (1993) pointed out as early as 1993, corpora can offer new insights into the theoretical and practical aspects of translation. The different stages in which various types of corpora can help in the translation process have been investigated by Aston (2000), while Varantola (2000) evaluates the use of dictionaries and specialised corpora, and other researchers investigate issues in the area of translator training, which is currently undergoing deep changes. The use of corpora and MT in the translation classroom has become a subject in its own right (Zanettin 1998; Yuste 2001, and Kübler forthcoming).

The translator is no longer seen as an isolated individual, working with a paper dictionary. A range of new resources are available for translators, particularly for translating technical documents[1]. However, there is a fear that machines, especially MT, will eventually replace translators[2]. MT has already changed the way professional translators work, but will not replace human beings. Today, it can be used as a tool to provide translators with quick on-the-fly versions that need thorough proofreading. The experiment described in this paper deals with the **next step**: Customising MT systems to provide translators with **a time-saving tool producing good quality results**.

We shall show how MT systems can be customised using existing resources, such as on-line glossaries and existing or self-made corpora, initially collected for a different purpose. A combination of resources, such as terminology extraction and conventional corpus linguistics tools, can be applied in the building of complete dictionaries containing sophisticated linguistic information. The recycled resources will be described, together with the tools used. The Systran user-customisable on-line MT system is then presented, with the linguistic features that can be integrated. The methodology applied in the creation of new dictionaries is detailed, and samples of improved translations are provided. A time-based evaluation of manual and MT outcome is included. The conclusion points to some work that remains to be done.

2. Resources

The project was carried out by **recycling** existing language resources, and using on-line Web-based resources. The tools that were used are simple to implement and do not require specific programming knowledge. The language resources that are readily available for assembling dictionaries can be divided into three categories:

- on-line bilingual technical glossaries;
- monolingual and parallel technical **corpora**;
- the Web as a corpus[3].

In this computer-science-based project, all three types of language resource were used .

2.1. Bilingual glossaries

On-line Web-based bilingual glossaries generally propose aligned lists of English terms and equivalents in French. These dictionaries are normally small, containing a few hundred headwords, usually with few verbs, adjectives or multiword units. They do provide useful lists

[1] Translation memory, term extraction tools, term base management software can all help when translating Languages for Specific Purposes (LSP), including Web sites, user manuals, help files, and financial documents.

[2] *Ouaibe et traduction: que craindre du Systran?* http://www.geocities.com/aaeesit/art21.html

[3] i.e. making linguistic queries with search engines, and search tools like WebCorp (see section 2.3. below).

of bilingual entries in the specialised area of computing, though they partly have the same headwords. Three glossaries were selected initially, because they contain terms that do not cross LSPs because they are domain-specific. They were downloaded, corrected, and formatted, to be compiled as customised dictionaries in Systranet. Here is the list of selected glossaries and the number of headwords for each:

- The HOWTO translation project glossary[4]: a small glossary of 200 words discussed and agreed upon in the project discussion list .
- Netglos Internet Glossary[5]: a multilingual glossary of Internet terminology compiled in a voluntary, collaborative project, containing 282 terms.
- The RETIF[6] site glossary. This short glossary contains 73 terms approved of by the French Governmental Terminology Commission for Computing and the Internet.

2.2. Corpora

Corpora make up the core resource exploited by the Systran team. Smaller corpora, exploited with simple tools, produce interesting results on a more individual scale. The smaller corpora used in the experiment had been collected to teach computer science English to French-speakers (Foucou & Kübler 2000). The texts used are highly technical and freely available on the Web:

- Internet RFC[7]: 8.5 million words: monolingual English corpus. This corpus consists of the Internet Request For Comments available on the RFC documentation site.
- Linux HOWTOs: English to French aligned corpus, ca. 500 000 words. The English HOWTOs and their translations in several languages are available on the Linux documentation site[8].

The above-mentioned corpora are embedded in a Web-based environment that can be accessed on our Wall[9] site.

2.3. The Web

The Internet has become a necessary resource for linguists, lexicographers, translators, and other language researchers, providing them with on-line dictionaries, reference documents, newsgroups. The Web can also be considered as an open-ended, unstructured corpus which can be queried using search engines, though these are not tailored for linguistic search. A specific linguistic search tool is Webcorp[10] (Kehoe & Renouf, forthcoming), which provides users with concordances, collocates, and lists of words found on Web pages; we have used this for a variety of purposes. A Web-based search strategy should be used in conjunction with the off-line, finite, corpus-based approach, since they yield complementary information.

[4] http://launay.org/HOWTO/Dico.html
[5] http://wwli.com/translation/netglos/
[6] http://www-rocq.inria.fr/qui/Philippe.Deschamp/RETIF/19990316.html
[7] http://www.rfc-editor.org/rfc.html
[8] http://www.linuxdoc.org
[9] http://wall.jussieu.fr
[10] http://www.webcorp.org.uk

2.4. Tools

The first tool used is an on-line concordancer featuring perl-like[11] regular expressions, which gives access to aligned paragraphs of French and English texts from which a concordance has been extracted. Another on-line tool is a tokeniser, which allows the user to sort the words of a text in alphabetical order, or by frequency.

As the general philosophy of this experiment was to use simple tools, a commercially available term extraction tool was selected: Terminology Extractor[12], which works for French and English. It uses a dictionary to lemmatise the vocabulary of a text and produce four different output types:

- *Canonical forms:* recognised by the program and sorted by alphabetical order or by frequency; the most frequent forms are to be considered as potential terms.
- *Non words*: not recognised by the system; most of them are specialised terms.
- *Collocations*. Collocational extraction is based on a very simple principle: any sequence of at least two -- and at most ten -- words, that is repeated at least once is considered as a collocation. Stop words are discarded to avoid sequences, such as *sauvegarde de la* [save the], in which *la* is a determiner preceding the second part of the term, as in *sauvegarde de la configuration* [save the settings]. Collocates are good candidates for technical terms.
- *KWIC (key word in context):* for the combined three lists. This feature is used to extract lexico-grammatical information, on verb structures, for example.

3. Systranet: customisable dictionaries

Systran MT has been much improved in recent years (Sennelart et al. 2001). Systranet is an on-line service offered by Systran. Users have access to a dictionary manager which allows them to create and upload their own multilingual linguistically-coded dictionaries into Systran, in order to improve translation results. These multilingual dictionaries contain a list of subject-specific terms that are analyzed prior to using Systran in-house dictionaries. This feature is based on the assumption, demonstrated by Lange & Yang (1999), that domain selection and terminology restriction are beneficial to translation results.

Linguistic information, such as part-of-speech, number and gender, subcategorisation, or low-level semantics can be added to the user's dictionary entries. Once the dictionary has been compiled, its accuracy and linguistic coverage can be tested by translating subject-specific texts.

The translation results can be improved by modifying the dictionary, a recurrent process which can be continued so long as the modifications produce significant improvement. Systranet offers specific features that allow

[11] Perl is a particularly appropriate programming language for handling word strings or finding language patterns.
[12] http://www.chamblon.com

the user to see which terms have been translated using customised dictionaries, and which terms are not recognised at all. It allows the user to check whether the dictionary entries have really improved the translation results as expected. Another feature used to complete the dictionary is the non-word feature: all the words that have not been recognised by Systran or the user's dictionaries appear in red. They can then be integrated into the user's dictionary.

4. Experiment and methodology

We chose technical documents written by experts for experts, the Linux HOWTOs, which are the user manual of the Linux operating system. This experiment is part of a larger project that consists in translating all the new HOWTOs using MT. HOWTOs are documents of various size, describing the way to install the system and software related to it. Existing software is constantly updated and augmented, so the corresponding documents are updated and new documents are written with each new program. These documents have been translated into several languages by the various Linux communities. The French Linux community has developed a translation project[13] in which the translation is usually done by non professional, voluntary translators. People choose the document they want to translate and do the job. Today, most HOWTOs have been translated, which makes it possible to align the French translations with the English source and use them as a parallel corpus.

The task set for the experiment was to provide a complete and appropriate dictionary to translate the remaining untranslated Linux HOWTOs. This is based on the assumption that the initial dictionaries will be augmented in the light of each new text to translate. Since a comparative study of the translation results -- with and without customised dictionaries -- had to be established, each text was first translated without using any specific dictionary.

4.1. Creating the dictionaries

The methodology is a combinatorial approach, recycling data and using terminology extraction tools.

First, the three glossaries mentioned above were downloaded and converted into dictionary files, augmented with linguistic information, giving more than 500 entries. These glossaries were selected when translating a HOWTO. Then, a more complete and corpus-based approach was applied. It produced two types of dictionary: *step-one dictionary* and *step-two dictionary*.

4.1.1. Step-one dictionaries

The step-one dictionaries were created using term extraction software, corpora, and a concordancer. This sort of dictionary can be produced using large corpora, but the most efficient solution for the individual user is to apply it to the texts to be translated.

The candidate texts were processed using Terminology Extractor. Initial candidates for headwords in the dictionaries were selected from the non-word and collocation lists. Unlike the existing glossaries, Terminology Extractor outputs do not provide French equivalents for the English words. On-line term banks, such as *Le Grand Dictionnaire Terminologique*[14] or *Termium*[15] proved insufficient for translating most terms. A corpus-driven approach was adopted to find French equivalents: the RFC corpus was used to find more information about context, the aligned HOWTO corpus was queried with the regular expressions concordancer (Wall) to find appropriate translations, as illustrated below.

The term README in the computing context is used as a noun, as shown in the following context, in which the term is the head of a subject NP:

links which Linus describes in the **README** are set up correctly. In general, if a

Figure 1. The noun *README* in context

The term *addon* was in the non word list, but by using the HOWTO corpus, we found contexts and a French translation:

The FWTK does not proxy SSL web documents but there is an **addon** for it written by Jean-Christophe Le fwtk ne route pas les documents web SSL, mais il existe un **module complémentaire** écrit par Jean-

Figure 2. The noun *addon* and its French translation

This stage was necessarily completed by using Web search engines to verify some translations found in the HOWTOs, or to deduce new translations from indirect queries. Since the documents are translated by various people who are usually not professional translators, but computing experts, the French versions of the HOWTO are not homogeneous. This means that one English term can be translated by several different words that are true synonyms in French. Only one equivalent must be chosen for the MT dictionary. Another problem is the case of borrowings. In spoken computing French, the English term is often used. Even in written texts, and especially in translations, usage leads translators to keep the English term and give the French equivalent once at the beginning of the document.

When no answer can be found in the HOWTO corpus, WebCorp can provide solutions. By looking for collocates and concordances for an English term in French language documents, possible translations can be traced back to the French sites. The collocates of *network* in French-speaking sites, for instance, allowed us to trace back *home network* and the French *réseau domestique* (Kübler, forthcoming).

4.1.2. Step-two dictionaries

Once a set of dictionaries has been produced for each HOWTO, it must be tested not only to correct possible

[13] http://www.traduc.org

[14] http://www.granddictionnaire.com

[15] http://www.termium.com

errors in the entries, but also to add the new words that are neither in Systran's nor in the customised dictionaries. The more HOWTOs are translated, the fewer words have to be added until the dictionaries are saturated, i.e. no new word can be added to improve translation results.

Step two is illustrated with the Home-Network-Mini-HOWTO, one of the not yet translated HOWTOs. Below is an example of translation results with and without customised dictionaries:

Source text	This page contains a simple cookbook for setting up Red Hat 6.X as an internet gateway for a home network or small office network.
Without cust. dict.	Cette page contient un cookbook simple pour le chapeau rouge 6X d'établissement en tant que Gateway d'Internet pour un réseau à la maison ou le petit réseau de bureau.
With cust. dict.	Cette page contient un cookbook simple pour l'établissement ***Red Hat 6.***X en tant que ***passerelle Internet*** pour un ***réseau domestique*** ou un petit ***réseau de bureau***

Fig. 3: Comparing translation results with and without customised dictionaries

In the next table, the customised dictionaries were completed with the words badly or not at all translated with the first version of customised dictionaries.

Source Text	This page contains a simple ***cookbook*** for ***setting up Red Hat 6.X*** as an ***internet gateway*** for a ***home network*** or small ***office network.***
Step-one dict.	Cette page contient un **cookbook** simple pour l'établissement ***Red Hat 6.***X en tant que ***passerelle Internet*** pour un ***réseau domestique*** ou un petit ***réseau de bureau***
Step-two dict.	Cette page contient ***des recettes*** simples pour ***l'installation Red Hat 6.***X en tant **que** ***passerelle Internet*** pour un ***réseau domestique*** ou un petit ***réseau de bureau***.

Fig. 4: Comparing translation results with step-one and step-two dictionaries

4.2. Translation outcome

Comparing the translation outcome with and without customised dictionaries shows encouraging results. Testing existing customised dictionaries on another text in the same subject area demonstrates that the text-based dictionaries can be reused, and that fewer headwords have to be added. Little by little, translators can add to their own dictionaries in various LSPs.

Obviously, as in any translation process, those translation results must be proofread. However, the points that need correcting are quite different from a translation done by a human being. If the MT errors are obvious and often serious, they have the advantage of always occurring in the same context. Most errors in this particular MT system are due to the same syntactic failures and can easily be corrected by the translator, once recognised.

Conjunction and disjunction are two of the main problems in MT systems that have yet to be solved. The garbled translation is however easily corrected, since the errors are similar each time a conjunction or a disjunction appears in an NP context:

Source text	*Translation result*	*Correct transl.*
Your internal and external networks	votre interne et des réseaux externes	vos réseaux interne et externe
a fulltime Cable or ADSL connection	une connexion en continu d'AADSL	une connexion en continu par le câble ou l'ADSL

Fig. 5: Conjunction and disjunction in an NP context

Another characteristic of MT systems is the overgeneralisation of transfer rules which leads to errors. Again, it is quite easy to check and correct those errors, for instance, the system translates a zero article in English by a definite article in French, although, in most cases, it should be the indefinite article:

Source text	*Translation result*	*Correct transl.*
decoded by specific individuals	décodé par les individus spécifiques	décodé par des individus spécifiques

Fig. 6: An example of transfer rule overgeneralisation

4.3. Human vs machine?

We selected two HOWTO totalling 9357 words in English. The expansion coefficient (15% in French) brings the total up to 10 750, i.e. ca. 36 standardised pages. This should take a professional translator from 5 to 7 days, depending on the tools used. Systranet took less than two minutes to produce an outcome. Professional translators assess the proofreading necessary at ca. 2 days. MT can therefore be included in the set of tools professional translators can actually use.

5. Conclusion

It has been demonstrated that the quality of translation can be significantly improved by importing customised dictionaries. Individual translators can thus create their own customised dictionaries with user-friendly and publicly available resources and tools.

These dictionaries recycle already existing resources, and their upgrading is corpus-driven. Translators working in LSPs can take advantage of a customised MT system because they can obtain quickly translated texts, and proofread them in a short time, as the errors generally have similar morpho-syntactic patterns. Although considerable work needs to be done in the beginning, after processing a few documents, the dictionaries are more or less saturated, and just a few words have to be added.

Further work will focus on reusing customised dictionaries to translate cross-LSP texts, such as digital cameras. More testing on the coding of Systranet customisable dictionaries is currently being done with students to improve coding rules and their applications.

6. References

Aston, G. 2000. I corpora come risorce per la traduzione e per l'apprendimento. In Bernardini S., Zanettin F. (eds) *I corpora nella didattica della traduzione*, Bologna: Cooperativa Libraria Universitaria Editrice Bologna, 21-29.

Baker, M. 1993. Corpus Linguistics and Translation Studies: Implications and Applications. In Baker, M., G. Francis and E. Tognini-Bonelli (eds.) *Text and Technology: in Honour of John Sinclair*, Amsterdam and Philadelphia: John Benjamins, 233-250.

Foucou P.-Y. et Kübler N. 2000. A Web-based Environment for Teaching Technical English. In Lou Burnard and Tony McEnery (eds.) *Rethinking Language Pedagogy: papers from the third international conference on language and teaching.* Frankfurt am Main: Peter Lang GmbH.

Kehoe, A. & A. Renouf (forthcoming) `Webcorp: Applying the Web to Linguistics and Linguistics to the Web'. In Proceedings of the WWW 2002 Conference, Honolulu, Hawaii, 7-11 May 2002.

Kübler N. (forthcoming-a). How Can Corpora Be Integrated Into Translation Courses ? Proceedings of CULT2 (Corpus Use and Learning to Translate). In Zanettin, F., S. Bernardini & D. Stewart, (eds.) forthcoming Corpora in translator education, Manchester: St Jerome.

Kübler N. (forthcoming-b). In Aijmer, K. (ed) forthcoming Proceedings of 21st ICAME Conference, Univ. Gothenburg, May 22-26 2002, Amsterdam & Atlanta: Rodopi.

Lang E. & Jin Yang 1999. Automatic Domain Recognition for Machine Translation. In *Proceedings of the MT Summit VII*, Singapore.

Renouf, A.J. (forthcoming). WebCorp: providing a renewable energy source for corpus linguistics, in Granger, Sylviane and Stephanie Petch-Tyson, (eds) *Extending the scope of corpus-based research: new applications, new challenges,*. Amsterdam & Atlanta: Rodopi.

Senellart, J. Dienès P., Varadi T. 2001. New Generation Systran Translation System. In *Proceedings of the MT Summit VII*, Santiago de Compostela, 18-22 September 2001.

Varantola, K. 2000. Translators, dictionaries and text corpora. In Bernardini S., Zanettin F. (eds) *I corpora nella didattica della traduzione*, Bologna: Cooperativa Libraria Universitaria Editrice Bologna, 117-133.

Yuste Rodrigo E. 2001. Making MT Commonplace in Translation Training Curricula –Too Many Misconceptions, So much Potential. In *Proceedings of the MT Summit VII*, Santiago de Compostela, 18-22 September 2001.

Zanettin, F. 1998. Bilingual Comparable Corpora and the Training of Translators. In *Meta*, 43(4), 616-630.

Zanettin, F. 2000. Parallel Corpora in Translation Studies: Issues in Corpus Design and Analysis. In Olohan M. (ed.) *Intercultural Faultlines*. Manchester : St Jerome Publishing.

Evaluating Translation Memory Systems

Angelika Zerfass

Freelance Translation Tools Consultant
Holzemer Str. 38
53343 Wachtberg
Germany
azerfass@debitel.net

Abstract

Since the mid 1980s, translation tools have taken over more and more of the daily lives of translators and translation project managers. But a lot of time now has to be spent on evaluation, training and administrative tasks.
Translation tools were designed to make the translator's work easier, faster and more efficient. They range from conversion utilities to terminology management, translation memories, machine translation as well as workflow and project management systems.
They were developed with the aim to reduce repetitive translation work, but on the other hand they add different tasks to the workload, like administrating databases and the like.
This presentation will give an overview of one area of translation tools - the different translation memory systems on the market today and the technologies they use. It includes a comparison of common basic features like word count, analysis/statistics function and pre-translation, some tools' specialities as well as the description of data exchange possibilities between the systems by use of the TMX format.
As there is no "one best tool" for everything, the aim of this workshop is not, to recommend one tool, but to provide some guidelines for evaluating translation memory systems according to individual requirements.

1. Translation Memory Tools - Overview

Translation memory systems, as the name implicates, "memorise" the translations made by a human translator. Most translation memory systems (often also called "TM-systems"), consist of a database that stores the original text along with its translation - a database of segment pairs.

"Segment" here indicates that the units that is being translated and stored to the database can range from a single word (for example a heading or an item in a bulleted list) to phrases, complete sentences or even whole paragraphs. The tools recognise a segment by a set of internal rules that define, for example, that a segment ends with a full stop or a paragraph mark.

During translation itself, the tool will automatically look up every new source language segment to be translated in that bilingual translation memory. If the same segment is found in the database, the system will offer the translation that was saved with this segment as a suggestion to the translator for reuse. If it does not find the very same segment, it will start looking for similar segments. These are the so-called "fuzzy" matches, as the source language segments (in the document and in the database) only match to a certain percentage. When the translator gets such a fuzzy match from the database, they can decide if and how much of it can be reused for the current translation. Usually the translator can even set the level of "fuzziness", that is the percentage of similarity, so that the system will only offer translations that can be reused without having to make too many changes to the suggested translation.

Thus the use of a translation memory system can increase consistency and it cuts the time for writing a translation. This is especially true for the translation of repetitive documents like technical documentation, manuals, instructions and updates of already translated material.

Translation memory tools are usually the main component of a tools' suite. These suits also offer recycling tools, so called alignment systems. These are used to prepare translations made without translation memory systems for reuse in such a translation memory tool. They read in the source and the target language files, display them in parallel and propose connections of the source language segments to the corresponding target language segments. A translator will then review these connections. Then, the segment pairs can be imported into the translation memory. From now on they can be used just as if they had been translated interactively with the system itself. Another component of such a tools' suite is the terminology management system - another database that stores single terms (or phrases) together with their translation(s) into the target language(s). The translation memory database and the terminology database work together during translation. The translator will not only get suggestions for the translation of whole segments but also a list of all the terms within that segment that were found in the terminology database. Other components of such a tools' suite could be workflow or project management systems as well as filters and utilities for file format conversion.

Translation memory systems also start to be customisable for use with document or content management systems and some are even programmable via an API (application programming interface - programming commands that enable the user to call the translation memory system from other applications).

2. Translation Memory Tools Basic Principle

Basically all translation memory tools were developed with the same goal in mind: Something that has been translated before should not have to be translated again from scratch. It should come out of the database or reference material so that the translator only has to decide whether the previous translation can be reused or needs to be modified.

The technologies used to achieve this are different. Some tools use a model of referencing the files of a previous project, The referencing model uses those previously translated files (original source language files and translated files) as the source for suggestions of new translations. This model works especially well for projects with many updates containing a lot of small changes.

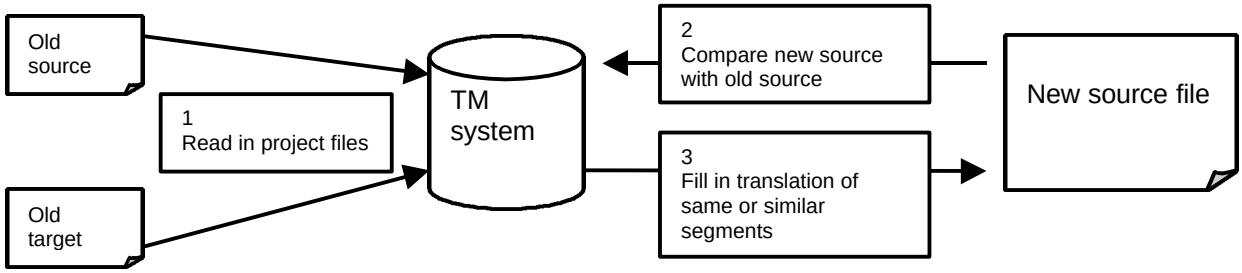


Figure 1. Reference Model

The database model on the other hand stores all translations ever made in one database, independent of context, which is useful if the same or similar segments appear in different projects and document types. Most of the commonly used translation memory systems are able to work with any language installed on the user's machine and they usually also allow the user to add project or user specific information to each translation.

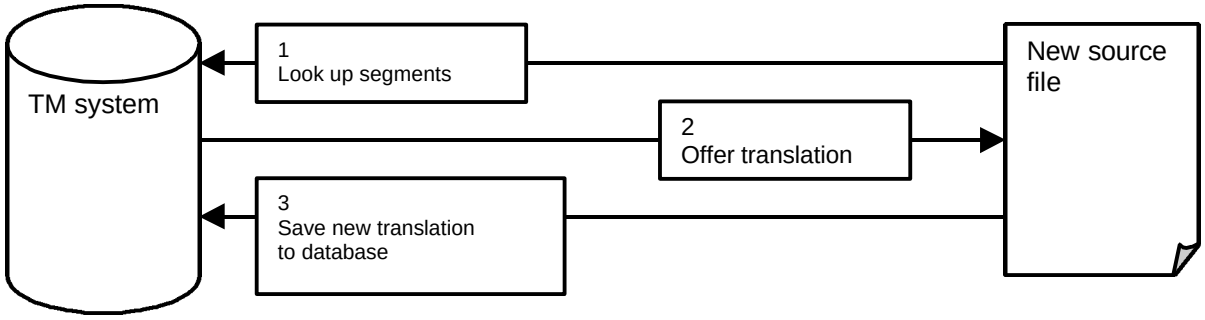


Figure 2. Database Model

3. Translating with Translation Memory Tools

The text to be translated consists of smaller units like headings, sentences, list items, index entries and so on. These text components are called "segments". Translation memory systems are equipped with a set of rules, which enables them to recognise, where a segment starts and where it ends. When translating with a translation memory system, the system goes through the text segment by segment, offering each of them to the translator together with any translation for this or a similar segment that has been stored in the database or can be found in the reference material. The translator decides whether to reuse the proposed translation, to adapt it or to create a new translation and then saves it to the system. Thus the translator builds up a store of segment pairs that can be referenced for future translation.

This store of segment pairs can also be used for analysing new files to determine the rate of recycling that can be achieved. Or it can be used to run a pre-translation, which creates files that contain segments with more or less matching translations already in them. This is very useful when working on a large batch of files or preparing files for other translators who are not working with a translation memory tool.

To be able to use translation memory tools on different file formats, from common Word files to DTP (desktop publishing) files, for example FrameMaker or Interleaf or files for the web in HTML, XML or SGML, some of these formats need to be converted to a format that the translation memory tool can work with. This happens either by use of separate conversion tools or filters integrated into the translation memory systems. Selecting a TM system therefore also depends on what file formats have to be worked on and how much time and effort needs to be spent on preparing and converting them to a usable format for translation and back to the original format afterwards.

Also, when it comes to software localisation for example, different tools have to be used for different parts of the project. The project might consist of text within the software from the user interface (GUI) to dialogs and messages as well as online-help files, documentation, packaging and marketing material and so on. And here different types of text require the use of different tools. GUI, software dialogs and messages are best translated with a software localisation tool, that is a translation memory tool that can read those special software file formats. They usually also contain testing features to check for consistent use of hot keys for example, or length related problems that might arise, if the translated text does not fit the button space it is supposed to appear on. But those systems are mostly specialised on the software itself.

For translation of the documentation, another translation memory tool is needed. And here the question arises how those tools for translating software and documentation interact, because what has been translated for one part might also be reusable in the other (this will be covered in the section about data exchange further down).

Online-Help files for example, could be translated with either a software localisation tool or with a translation memory system for documentation as both system types support this format.

4. Feature Comparison

All translation memory tools offer basic functionalities like word count or an analysis of recycling potential (how many of the segments in the file to be translated are present in the database or reference material as 100% matches or as similar, fuzzy matches). They also provide features for automatic pre-translation, search functionalities within the segment database, as well as access to terminology management components during translation. But each and every tool also has its specialities. These are the features that can influence the choice of tools.

Most translation memory systems read the files to be translated into the system itself, converting them into a table where one column contains the source language segments and another column that will be filled with the respective translation. Others connect to Microsoft Word so that any file that can be opened in Word does not have to be converted before translation and can be worked on in a WYSIWYG (what you see is what you get) mode. The translators can work in an environment that they are used to. Other file formats, for example DTP formats or so called tagged file formats like XML, HTML or SGML, are either converted or displayed in a separate editor. Colours are used to mark text to be translated as well as tags that make up the structure and formatting of the file.

More and more developers are enhancing the functionalities of the translation memory tools by adding new features like context sensitive pre-translation or machine translation-like components (for segments that have no match from the translation memory) as well as project management components.

5. Data Exchange between Translation Memory Systems

For some time, translators did not have the possibility to bring the data from one translation memory system into another system for reuse. A situation that was alleviated to some extent by the tools manufacturers by adding export functionalities for some proprietary formats of other manufacturers. But it was not feasible for each tool to support all export/import formats of all other tools - especially with new tools being developed and marketed all the time.

Now, the tool manufacturers have agreed to use one standard format for representing the data in their systems or at least to offer this format as one of the export formats. This allows an easier transfer of translation memory data from one system to another - even though the results are not always completely satisfactory. This standard is called TMX - Translation Memory Exchange format. It is an XML based representation of the data stored in a translation memory system.

5.1. Example of data representation in TMX format:

Segment pair:

This is a test. (English segment)
Dies ist ein Test. (German segment)

TMX representation:

```
<tu>
    <tuv lang="EN_US">
    <seg>This is a test.</seg>
    </tuv>

    <tuv lang="DE_DE">
```

```
        <seg>Dies ist ein Test.</seg>
        </tuv>
</tu>
```

Each segment pair is represented with a <tu> and </tu> tag that denote beginning and end of the segment pair. ("tu" stands for "translation unit", as those segments pairs are often called.) Then come the individual languages of the segments and the textual contents. This format could be produced and read by any translation memory system that works with TMX.

There are three levels of TMX compliance today. The first level only represents the text itself. The second level is able to represent the formatting information as well. And level three would be used to represent additional tool specific data like user IDs, project names and everything else the user has specified. Today, most tools comply at least to TMX level 1 or even to level 2.

6. Conclusion

Before investing in any translation tool, it is necessary to list the individual user requirements. This includes the file types that are to be translated. As most translation memory tools rely on structural and formatting information in the file, to segment and display the text, it should be tested if the way the files for translation are constructed, work well with this or that translation memory system. It could even mean to adapt the way of writing the documents in the first place, so that, at the translation stage, the tools that are used can handle the files more easily.

Another point is the networkability and the list of supported languages as well as the different supported file types.

Pricing for licenses, training and support should also be taken into consideration.

Then the tools should be tested for some time with real life examples to be able to evaluate, which tools answer the user's requirements best. Most tool manufacturers offer a trial period of about 30 days or a limited demo version of the software or, in case a longer evaluation period is needed, an extended trial with the full version of the software. This usually includes the need to buy a training session as well, to prepare the people who will be evaluating the software in the best possible way.

7. References

Some download sites for demo versions of translation memory tools:

- Trados - Translator's Workbench
  www.trados.com/products/download.htm

- Atril - Déjà Vù
  www.atril.com

- SDL - SDLX
  www.sdlintl.com

- Cypresoft - TransSuite2000
  www.cypresoft.com
  (supports only European languages)

- Star - Transit
  no download, contact Star for a demo CD at www.star-group.net

- Champollion - Wordfast
  Freeware
  www.geocities.com/wordfast/cat.htm

Some download sites for demo versions of software localisation tools:

- Pass Engineering - Passolo
  www.passolo.com

- Alchemy - Catalyst
  www.alchemysoftware.ie/demo4/

More information on TMX:
www.lisa.org/tmx

Language Resources at the Languages Service of the United Nations Office at Geneva

Marie-Josée de Saint Robert

United Nations Office at Geneva
1211 Geneva 10
mjdesaintrobert@unog.ch

Abstract

The language staff at the United Nations makes a very selective use of language technologies. So far no computer-assisted translation software has been installed on translators" workstations even though tests have been conducted for several years on the two major computer-assisted translation (CAT) systems at United Nations Headquarters in New York, for instance. The aim of this paper is twofold : 1) to show why CAT systems are not considered as potential sources of improvement of quality nor quantity in translation work at the United Nations, and 2) to present the kind of language resources that are considered essential for the adequate rendering of content in any of the six official languages of the United Nations (Arabic, Chinese, English, French, Russian and Spanish). This paper analyzes the particular linguistic and technical constraints specific to an international setting and argues in favour of a selected number of language resources used at the United Nations other than translation tools readily available on the market. Among such language resources, one finds search engines, government and research institutions" websites, and, in a not too distant future, institutional knowledge bases.

1. Introduction

In an international, multilingual environment such as the United Nations, surprisingly enough, translators and language staff in general are not considered on the same footing as substantive departments, which prepare reports and organize conferences. Wherever technological innovations are designed and developed, the primary concern is the diplomatic community or the international community at large, not the language staff. Although translators do have a major role to play in the preparation of parliamentary documentation, their needs, such as prompting automatic alignment of two language versions of the same document whenever desirable, are very seldom taken into consideration by United Nations designers and developers. This low profile for linguists may well explain why so few technological innovations have made their way through to the translator and the terminologist. More reasons can be found in the very nature of the translation process in multilateral diplomatic settings where linguistic and technical constraints play an important role.

2. Linguistic Constraints

Several linguistic constraints are obstacles to the straightforward application of language technologies to translation work. Some are quite obvious, while others are specific to international organizations.

2.1. Word Choice

Translation cannot be reduced to the mechanical substitution of one set of terms in one language by a similar set in another language.

2.1.1. Semantic Adequacy

The sentence starting with (1) should not be translated into French by (2) no matter how common that phrase is but by (3):

(1) the report shows
(2) le rapport montre que
(3) il ressort du rapport que

Also, the correct rendering in French of the English phrase (4) is not (5) but (6):

(4) abusive sexual practices that may affect very young girls

(5) pratiques sexuelles abusives qui peuvent affecter les très jeunes filles

(6) pratiques sexuelles dont peuvent être victimes les très jeunes filles

It is not always clear with CAT whether faulty phrases such as (2) and (5) would be offered by the system, as it may only keep the first instance found and disregard other instances of the same phrases found subsequently, and whether the translator in haste may not accept the phrases in (2) and (5) since both look correct from the grammatical point of view but are incorrect from the semantic point of view[1]. Maybe more accurate information on what CAT systems do is needed. Yet it remains to be seen whether distributed management of translation memories can be efficiently organized on a large scale, with fifty translators having the right to update the translation memory on a permanent basis in each language pair.

2.1.2. Lexical Variety

Translations serve the purpose of a specific communication need and should not be considered as models for translators to replicate across the board. Such is also the case for terminology in any target language. Mere electronic bilingual dictionaries or glossaries cannot

[1]In (2) an inanimate noun is used with an animate verb; in (5) it is as though sexual practices would be divided into two categories: abusive and non-abusive, which is wrong in the case of very young girls.

satisfactorily capture variation, not only in the original language but also in the target language, if based upon the assumption that a notion corresponds to a term in English and one or several terms in French, for instance. Names given to human rights are a case in point. A terminologist would very happily collect the names of all rights, starting with the right to food, to adequate housing, and to education, while a translator would resent it. Such rights are indeed referred to under different names by different speakers, and a too rigid list of rights would miss the needed subtleties while discussions are still under way. Should "adequate housing" be rendered in French by "logement convenable," "logement adéquat," "logement suffisant," or "logement satisfaisant," all four equivalents being found in United Nations legal instruments or resolutions, and not by "bonnes conditions de logement" or "se loger convenablement" when the context allows or requires it? Translators want to preserve flexibility, when present-day translation systems propagate rigidity and, as a lurking consequence, poverty of style and vocabulary. For Fernando Peral (2002), a translator at the International Labour Organization: "The main operational problems of "semi-automatic" translation [i.e., translation with the help of translation memory systems] are linked to the quality of the output and to a process of "de-training" of the translator, who becomes less and less used to the mental process of searching for proper solutions in terms of functional equivalence and relies more and more on the machine"s decisions, which inevitably affects professional development and job satisfaction."

2.2. Linguistic Insecurity

Document originators at the United Nations are nationals from over a hundred and twenty countries. In most cases their native language is not one of the official languages of the Organization, and document drafters erroneously think they have to use English, which may prevent them from using their main language, even when it is an official language, and produce better originals. Documents may also be submitted to the United Nations by officials or experts working for Member States that do not have either any of the official languages of the Organization as their main language. Syntactic, semantic and morphological mistakes are therefore not rare in documents, and in most cases only translators are in a position to detect mistakes and rebuild faulty sentences in the original text. Only they are required to work in their native language that is one of the official languages. Due to lack of resources at the United Nations, only a small portion of all documents is edited prior to being translated (e.g., documents prepared by the Commission on Human Rights). Translators consequently do act as filters for grammatical correctness and language consistency as they work on the texts to be translated. As a result, they often improve original texts whenever the drafters or submitting officers accept their changes in the original documents. A translation memory processing straightforwardly a document to be translated prior to the perusal of a translator may not detect inappropriate use of terms or syntactic errors in the original language. Even when an automatic term-checking system is appended to the translation memory, it may not be as efficient as a human eye either. The fear therefore is that a computer-assisted translation system may add more mistakes to the original ones, which will then be even harder to detect and correct.

2.3. Different Stylistic Rules

Document drafters use a variety of writing rules and styles to convey meaning. For instance, among writing styles one can mention the fact that repetitious words are not considered as poor style in English but are definitely considered poor style in French. The English sentence (7) presents a repetition of the word "aircraft" which the French rendering in (8) would avoid:

(7) the shooting down of civil aircraft by a military aircraft

(8) la destruction d'aéronefs civils par un appareil militaire

2.4. Functional Adequacy

Each Committee or Body has specific ways of expressing an idea in order to reach a consensus within its respective audience or circle. Underlying references to protagonists, former meetings, earlier decisions discussed by Committee members but not explicitly mentioned in the text play an important role in translation. Sometimes the reasoning of a *rapporteur*, a speaker or an author, or an amalgam of lengthy sentences couched in simple terms that are perfectly unintelligible to the outsider, i.e., someone who has not participated from the beginning in the discussions, has to be left untouched in the original. Acceptability of a translated text does not come solely from its grammatical and semantic well-formedness. It must also be appropriate within the United Nations context. A translated text must, like its original, follow a highly standardized path: it must convey the impression of having been written by a long-time member, perfectly familiar with the background in which the text has been drafted, even if it is deliberately vague or obscure. In fact most United Nations texts cannot be interpreted without prior knowledge of the particular political framework in which they appear. The sociopolitical motivation and rationale behind a text are part of the unwritten constraints imposed on communicative competence at the United Nations. Developments in artificial intelligence are not perceived to have reached this level of refinement. As Fernando Peral (2002) puts it: "translation is based on finding "functional equivalences" that require linguistic, intertextual, psychological and narrative competence; only human beings are capable of determining "functional equivalences"; productivity in translation is therefore intrinsically linked to the capacity of the translator to find the adequate functional equivalence, i.e., it is based on the quality of the translator."

These constraints conflict with the concept of translation reuse for translation purposes on which most commercially available alignment tools and translation memory systems are based, especially when document traceability (i.e., the capacity of retrieving the complete document from which a sentence is extracted by the translation memory system) is not guaranteed.

3. Technical Constraints

Quality requirements are not always met in translated documents for technical reasons.

3.1. Time Constraints

Non-respect of deadlines for document submission results in not allowing translation to be performed in the required conditions. Feeding translation memories with texts that have not been properly revised for lack of time appears to be useless, even when such texts are considered as basic texts in an area. The underlying assumption is that basic texts can be improved over and over as they are cited in other texts, but no one can guarantee that it will indeed be the case, as translators are more and more required to work under emergency conditions, keeping revision at a very low level.

This explains why most documents are not considered by translators as authoritative sources for official denominations either in the source or in the target languages. Most official names of international and national organizations, bodies and institutions are referred to under several names in various documents and sometimes even within the same document. Alignment tools and translation memories that would provide precedents in two languages to translators might perpetuate the number of variants and confusion rather than helping translators to use the right equivalent, unless quality assessment is performed, which is a rather slow and uneconomical process looked down upon in an era of search for productivity gains. The problem is even more complex when it comes to designating a body whose name may be official in one or two languages but not in other languages. Chances are that transliterated names in English, French or Spanish rarely reappear again under the same denomination unless a rather time-consuming compilation is done to provide the best possible equivalents across official languages that would be used by translators. Yet as George Steiner (1975) rightly puts it: "Languages appear to be much more resistant than originally expected to rationalization, as well as to the benefits of homogeneity and technical formalization." Languages resist because human beings resist.

3.2. Digital Divides

Other technical constraints make the use of CAT systems difficult: 1) non-submission of documents in electronic form: many documents are submitted on paper with last minute written corrections – linguistic insecurity or a changing appreciation of political requirements being the main causes of last minute changes; 2) non-availability of reference corpora: some official references may exist in one or two languages, and have to be translated into other languages – reference documents that are considered as authoritative in one language pair may not be so in another, thus the task of building translation memories is labour-intensive, language pair by language pair; 3) scarcity of digitalized language resources in some languages: translators cannot completely switch to ready-made technological innovations – expertise in conventional research means should be kept.

3.3. Lack of Preparedness

CAT tools are known to be most efficient with repetitive texts. So far, since at the United Nations not all texts are available in electronic form, it is hard to assess the amount of repetition to be able to ascertain whether or not CAT is an efficient tool in this environment.

Proper training also has to be given to translators to make certain they know how to utilize the tools that they are given. The fear is that translators are no longer assessed only for their linguistic and narrative competence and performance, but by their computer skills.

Finally, equipment used in an international organization has to be compatible with the equipment required by a particular CAT software.

4. Tools for Translators

Translators at the United Nations make use of internal glossaries and terminologies developed within the specific institutional constraints.

4.1. In-house Glossaries

A dictionary look-up tool commonly used by translators at the United Nations provides a list of equivalents to remind translators of all possible synonyms as is the case for "significant" in English and its possible renderings into French:

"Significant - Accusé, appréciable, assez grave/long, caractéristique, certain, considérable, de conséquence, d'envergure, de grande/quelque envergure, digne d'intérêt, d'importance, de poids, de premier plan, distinctif, efficace, élevé, éloquent, explicatif, expressif, grand, important, indicatif, instructif, intéressant, large, louable, lourd de sens, manifeste, marquant, marqué, net, non négligeable, notable, palpable, parlant, particulier, pas indifférent, perceptible, plus que symbolique, positif, pour beaucoup, probant, qui compte, qui influe sur, réel, remarquable, représentatif, révélateur, sensible, sérieux, soutenu, significatif, spécial, substantiel, suffisant, symptomatique, tangible, valable, vaste, véritable, vraiment; a significant proportion: une bonne part; in any significant manner: un tant soit peu; not significant: guère; the developments that may be significant for: les événements qui peuvent présenter un intérêt pour; to be significant: ne pas être le fait du hasard." [2]

Access to validated and standardized terminology is considered more important than access to tools for document reuse other than the basic cut and paste function from documents carefully selected by the translator and not automatically provided by the system. Dictating sentences afresh, once proper terminology has been identified, also is considered a less time-consuming process than reading and correcting all or a selection of all possible renderings of a sentence found in previously translated documents by a context-based translation tool. Language resources used by United Nations translators thus are primarily terminology search engines that facilitate the search for adequacy given the specific

[2] Organisation des Nations Unies (2000).

context in which the document has been drafted, rather than any previous context.

4.2. Web resources

Language resources used by translators also include online dictionaries and government and research institutions" websites that translators have learned to identify and query for information extraction and data mining. Portals have been designed to help translators locate best language and document sources on the Internet.

4.3. Alignment Tools

Additional tools are document alignment tools by language pairs. Indexing of large text corpora for retrieval of precedents are felt preferable to tools that provide text segments, be they paragraphs, sentences or sub-units with their respective translations, but without any indication of date, source, context, originator, name of translator and reviser to assess adequacy and reliability in an environment where many translators are involved.

4.4. Knowledge Base

The construction of a knowledge base is envisaged to help translators perform their task in a more efficient manner. Ideally it would capture all knowledge generated by United Nations bodies and organs and various organizations and institutions working in related fields (i.e., any subject from outer space to microbiology tackled by the United Nations), and the knowledge and know-how of an experienced translator well trained in United Nations matters and that of an experienced documentalist knowing which documents are the most referred to. Such knowledge base would, for instance, predict instances where "guidelines" should be translated in French by "directives", as given by most dictionaries, and where "principes directeurs" would be a more appropriate translation. In statistical documents at the United Nations, one finds "recommendations," a term which is translated by "recommandations" in French and refers to rules to be followed, and "guidelines", translated as "principes directeurs," which are mere indications to be taken into consideration. If the term "directives" would be used in such context, it would convey the meaning of a document of a more prescriptive nature than "recommandations" would, which are actually more binding. Such instances of translation are best captured by a knowledge base that refines contexts and provides best reference material on any topic in the text to be translated. The knowledge base would provide not only adequate referencing and documentation of the original, but also the basic understanding of any subject that arise in a United Nations document.

Such knowledge base ideally would reduce the choices offered to the translator rather than list all possibilities. The easier it is for the translator to make the decisions he or she needs the faster he or she delivers.

The knowledge base would offer the translator with past alternatives, too, as in the case of "sexual harassment", translated into French by "harcèlement sexuel". Other French equivalents were tested before this rendering was coined and accepted. They may arise in a French original to be translated into other languages and thus should be retrievable: "assiduités intempestives," "avances (sexuelles) importunes," "privautés malvenues," "tracasseries à connotation sexuelle". The knowledge base would refer, too, to associated terms: "attentat à la pudeur," "outrages."

5. Conclusion

In conclusion, United Nations translators are very cognizant of the limitations of automated tools for translation and are more inclined to rely on easily accessible, structured information concerning the history and main issues in a particular subject matter in order to be completely free to choose the best translation equivalents.

6. References

Organisation des Nations Unies. Division de traduction et d'édition. Service français de traduction. Vade-Mecum du traducteur (anglais-français), *SFTR*/15/Rev.3, septembre 2000.

Peral, F. (2002). The Impact of New Technologies on Language Services : Productivity Issues in Translation. Paper for the *Joint Inter-agency Meeting on Computer-assisted Translation and Terminology (JIAMCATT)*, 24-26 April 2002. World Meteorological Organization. Geneva.

Steiner, G. (1975). *After Babel. Aspects of language and translation.* (first published in 1975, reedited in 1998 by Oxford University Press).

Global Content Management – Challenges and Opportunities for Creating and Using Digital Translation Resources

Gerhard Budin

University of Vienna
Department of Translation and Interpretation
Gymnasiumstraße 50, A-1190 Vienna
gerhard.budin@univie.ac.at

Abstract

In this paper the concepts of content management and cross-cultural communication are combined under the perspective of translation resources. Global content management becomes an integrative paradigm in which specialised translation is taking place.

1. Convergence of content management and cross-cultural communication

Two different paradigms that have previously developed independently of each other have converged into a complex area of practical activities: cross-cultural communication has become an integral part of technical communication and business communication, and content management has become a process that is complementary to communication by focusing on its semantic level, i.e. its content. Specialised translation as a form of cross-cultural communication is a content-driven process, thus digital translation resources become a crucial element in content management that takes places in a globalised marketplace.

Content management has recently emerged as a concept that builds upon information management and knowledge management with an additional focus on content products, such as databases, electronic encyclopedias, learning systems, etc. Due to globalised commerce and trade, such products are increasingly offered on multiple markets, therefore they have to be adapted from a cultural perspective, which also includes the linguistic viewpoint. We will have a closer look at the concept of content, its transcultural dimension, and the role translation resource management plays in this area.

2. Reflections on concurrent trends

Economic globalisation had been a re-current development during several phases in modern history and several industrial revolutions and has been one of the crucial driving forces in the development of modern engineering, in particular computer technology. Together with rapid advances in telecommunications it was the basis for building databases and global information access networks such as the Internet. Visualisation techniques and constantly increasing storage capacities led to multimedia applications.

This increasingly powerful technology base has then been combined with terminology management practices in the form of termbases, with multilingual communication and translation requirements as well as with cultural adaptation strategies in the form of localisation methods. Language engineering applied to translation in the form of computer-assisted translation, translation memory systems, and machine translation have recently been combined with localisation methods and terminology management for creating integrated workbenches.

On the economic level, international trade and commerce have increasingly required cross-cultural management and international marketing strategies tailored towards cultural conventions in local markets. This trend towards customisation of products has now generated personalised products and services that are based on specific user profiles, customer satisfaction and quality management schemes. The emergence of information and knowledge management systems has been another key development in recent years. Computerisation and economic globalisation are the key drivers in a complex context of the information society, leading to interactive processes between linguistic and cultural diversity, professional communication needs in economic and industrial processes and technological developments. As a result, cross-cultural specialised communication and content management have emerged, both complex process themselves, as a dynamic and integrative action space in society.

3. What is Content?

While terms such as *data*, *information*, *knowledge* have been defined many times so that we can compare and ideally synthesize these definitions, the term content has not been defined so often. But since this term is essential for our discussion here, and since it is used so often in terms such as *content management*, *eContent*, *content industry*, etc., we have to take a closer look at what this term actually means.

In a modest attempt at distinguishing the different conceptual levels, an iterative and recursive value-adding chain emerges:

data + interpretation = information + cognitive appropriation = knowledge + collective representation and utilization = content

Each higher level of complexity integrates diverse elements of the lower level. Usability aspects are most important on the content level. All lower levels remain crucial on the higher levels, e.g. data management is still an important part of content management.

Looking at the generic concept behind the word content, we would say: *Content* is what is *contained* in a written document or an electronic medium (or other

containers of such types). We would expect, that any content has been created by humans with certain intentions, with goals or interests in their minds. So we can say that content is usually created for specific *purposes* (such as information, instruction, education, entertainment, arts, etc.).

Content is often created in specific *domains* (arts, sciences, business/industry, government, social area, education, etc.). When specific content that was originally created in a science context, for instance, it will have to be adapted and re-organised, in order to be able to re-use this content in other contexts, e.g. in secondary education or in industry.

Discussing the term content, we cannot avoid dealing with related terms such as data, information, and knowledge. In fact it is essential to have a clear distinction between the meanings of (the concepts behind) these terms. From an economic or business perspective, 'data is a set of particular and objective facts about an event or simply the structured record of a transaction' (Tiwana 2000: 59f). We derive information by condensing (summarising, eliminating noise), calculating (analysing), contextualising (relating data to concrete environments, adding historical contexts), correcting (revision of data collections on the basis of experience) and categorising data (Davenport/Prusak 1998).

Data management has always been a fundamental activity that is as important as ever. Data repositories and data sharing networks are the basic infrastructure above the technical level in order to facilitate any activity on the levels above, i.e. information management and knowledgement. The transition from information to knowledge can also be described from a systems theory point of view: a certain level of activities has to be reached, so that knowledge 'emerges' from information flows. Many knowledge management specialists warn companies not to erroneously equate information flows to knowledge flows. In order to legitimately talk about knowledge, a number of conditions have to be met:

- *Cognitive appropriation*: knowledge is always the result of cognitive operations, of thinking processes. Yet knowledge is not limited to the personal, individual, subjective level. When people consciously share knowledge on the basis of directed communication processes, it is still knowledge, either referred to as collective or shared knowledge, or as interpersonal, intersubjective, or objective knowledge. In theories of scientific knowledge, the term 'objective knowledge' was mainly explicated by Karl Popper (1972) and is the result of regulated research processes such as hypothesis testing, verification, proof, etc., and that is written down in science communication processes. This is the justification for libraries to talk about their knowledge repositories in the form of books that contain this type of knowledge, i.e. objective knowledge – but as mentioned above, this knowledge has once been subjective knowledge in some persons, in this case scientists, that had thought and communicated about it before.
- *Complexity*: the level of complexity is another factor in the transition from information to knowledge. The same processes as on the previous emergence level, from data to information, are relevant: condensation of information (summarising), analysis and interpretation of information gathered, contextualisation (relating information to concrete problem solving situations, embedding and situating information in historical contexts and drawing conclusions from that, correcting (revision of data collections on the basis of experience) and categorising knowledge accordingly.
- *Life span:* the validity of knowledge has to be checked all the time. Again we are reminded by Karl Popper that all knowledge is unavoidably hypothetical in nature and that no knowledge is certain for eternity. Therefore we constantly have to redefine the criteria by which we evaluate our current knowledge for its validity. Another metaphor from nuclear physics is used for knowledge, especially in scientometrics: the 'half life'of knowledge is constantly decreasing, due to the increase in knowledge *dynamics*, not only in science and technology, also in industry, commerce and trade, even in culture, the arts, government and public sectors, the social sector, etc.

In knowledge management, three basic steps in dealing with knowledge are distinguished (Nonaka/Takeuchi 1998, Tiwana 2000: 71ff, etc.):

- *Knowledge acquisition*: learning is the key for any knowledge management activity
- *Knowledge sharing*: the collaborative nature of knowledge is the focus
- *Knowledge utilization*: knowledge management systems have to allow also informal knowledge to be dealt with, not only formalized knowledge (this is a crucial factor in evaluating knowledge technologies for their suitability in knowledge management environments.

The focus and the real goal of knowledge management is actually on *content*, i.e. not on the formal aspects of computing, but on what is behind the strings and codes: the concepts and the messages. When knowledge is then packaged as a product for a certain audience, presented in certain media presentation forms, then we can speak about *content*, which also has to be managed in specific repositories and to be processed for publishing purposes, for instance.

As soon as we introduce another dimension, that of culture and cultures, communicating content across cultural boundaries becomes a crucial issue. Since we talk about *localization* as the process of culturally adapting any product to a market belonging to another culture than that of the original market of a product, content also needs to be localized when it should be presented to other cultures. Translation, as a part of the complex process of localization, is one crucial step in this process, but not the only one. Content localization may very well involve more than translation in the traditional sense, i.e. we might have to re-create part of that content for another culture, or at least change fundamentally the way this content is presented to a certain culture.

Since 'content' is a relational concept, we have to ask ourselves, what contains something, i.e. what is the container, and what is in this container. A book (with its table of *contents*), for instance, is such a container, or a database with the information entered in the records as the content. A text or a term can also be containers, with the semantics of sentences and the meaning of the term as the

content. But this distinction between container and content cannot be made in a very clear-cut way. We are faced with a semiotic dilemma. Form and content always interact. The medium we choose to present certain information will have some impact on this information, the structure of the information will also lead us in the choice of an adequate medium. Usually we cannot completely separate the container from the content, the form from the content, the term from the concept, the semantics from the text, the medium from the message, etc. Despite the heuristic validity and necessity of an analytical separation, we need a synthesis in the sense of a dynamic interaction, an interactive complementarity. At the same time we also might want to transform one form of knowledge representation into another one, for certain purposes and tasks, and then have to be sure that the content of each knowledge representation does not change – a difficult task.

Similar to typologies of data, information, and knowledge, we also need a content typology. There are different criteria for distinguishing types of content:

the domain where specific content is created in: any field of scientific knowledge, a business branch, a profession, a form of art, a type of social activity, etc. For this type of distinction, we may also differentiate different degrees of specialisation (highly technical and scientific, monodisciplinary or multidisciplinary, popularised, etc., depending on the audience targeted);

the form of representation: text, picture, personal action, etc. or the medial manifestation: web site content, the 'story' of a film, of a video, a piece of music recorded, a digitized scroll, etc.

Here we see again that the form of representing content and the medium chosen to do this is constitutive for distinguishing types of content.

First of all, the purpose of the content: instruction, education, research, aesthetic and artistic purposes, etc. Secondly, the kind of content product that is designed for a particular target audience (e.g. a multimedia CD-ROM for 6-year old children to learn a foreign language, e.g. English). In addition to a content typology, we also have to look at the structures of content. In this respect, and regardless of the content type, we can make use of terminology engineering, and, more recently, also ontology engineering. Terminologies and ontologies are the intellectual (conceptual) infrastructures of content, both

- implicitly (in the form of personal or subjective knowledge of the content generator), or
- explicitly (as objective knowledge laid down in a specific presentation form).

So we can conclude that concepts are content units (conceptual chunks) and that conceptual structures (the links among concepts) are the structures of concept. Again we have to remember that the multi-dimensional content typology will determine the concrete structures of content that users will encounter in specific products.

4. Global Content Management

After having investigated a little bit into the concept of content, we can now look at content management and how cultural diversity determines this practice. Since the target audience of any content product is always culture-bound, i.e. belonging to one or more cultures, with we can simply state that content management always has to take into account cultural factors in content design and all other processes and tasks of content management. The language(s) spoken by the target audience, social and historical factors, among many others, are examples of criteria for concrete manifestations of content management. Also the meta-level of content management, i.e. those who are content managers, are also culture-bound. Those who have designed and created content products, such as multimedia encyclopedias on CD-ROM, have to be aware that they themselves are belonging to at least one culture (in most cases, there will be one pre-dominant culture in such content management teams), and that this very fact will unavoidably determine the way the content of the product is designed.

Now we look at a list of key processes of content management:

- Design and creation of content
- Processing of content, such as

Analysis of existing content structures, segmentation of content into units, aggregation of content units into structures, condensation of content (summarization, abstracting, etc.), expansion of content into more detailed forms, transformation of content, etc.

- Presentation of content in different media and knowledge representation forms (see above)
- Dissemination of content on intranets or other web structures, on CD-ROMs, but also more traditionally in the form of books, etc.
- Sharing content in collaborative workspaces
- Using content for various purposes

Taking into consideration the differentiation between data, information, knowledge, and content (see above), we can make a parallel distinction between data management, information management, knowledge management, and content management. It is important to note that each management level is based on the one underneath, i.e. information management is impossible without data management, knowledge management needs both, data management and information management, and content management relies on all three levels below. The following figure shows different levels of complexity and levels of integration. As a result of combining these two dimensions, degrees of usability can be differentiated: data management is usually not user-oriented, since it is an internal process at an infrastructural level. Content management, on the other end, is most user-oriented.

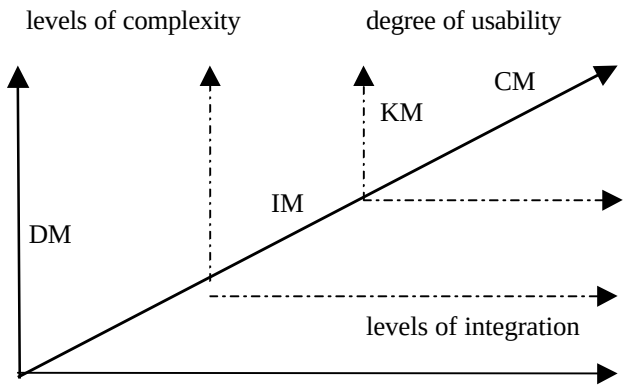


Figure 1: Levels of complexity and levels of integration, and degrees of usability as emergence levels of data management (DM), information management (IM), knowledge management (KM), and content management (CM)

Now we should return to the aspect of cultural diversity and the way it determines content management. Global content design, accordingly, is an activity of designing content for different cultures as target groups and is cognizant of the fact that content design itself is a culture-bound process, as shown above.

From the field of cultural studies we can benefit when looking at definitions of what culture is: a specific mind set, collective thinking and discourse patterns, assumptions, world models, etc.

Examples for types of culture are corporate cultures, professional, scientific cultures, notably going well beyond the national level of distinguishing cultures.

Cultural diversity is both a barrier and at the same time an asset and certainly the raison d'être for translation, localization, etc.

The following model shows the various dimensions of Global Content Management discussed above. The term element 'global' stands for all the cross-cultural activities such as translation, localization, but also customization, etc. 'Content' includes terminologies and ontologies as its infrastructures, products and their design, user documentation, but also pieces of art, etc. And the management component includes all the processes such as markup and modelling, processing, but also quality management, communication at the meta level, etc. Usability engineering is crucial for all these components:

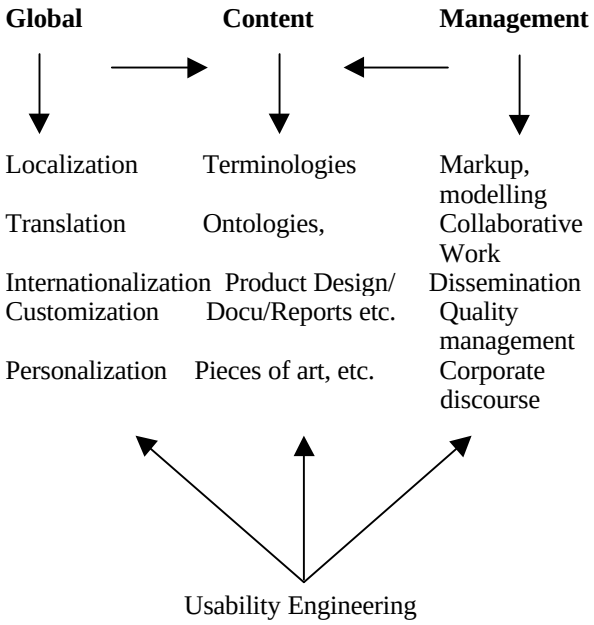


Figure 2: the three components of global content management with individual processes and components, all three nowadays determined by usability engineering imperatives

5. Pragmatic Issues in Global Content Management

Content management processes cannot do without appropriate knowledge organization and content organization. Terminological concept systems are organized into Knowledge Organization Systems (KOS) that can be used for this purpose of content organization:

- Thesauri, Classification Systems, and other KOSs, also conceptualized as (extrinsic) ontologies
- (Intrinsic) Ontologies (language-related, e.g. WordNet), domain-specific (medicine, etc.)

In order to establish and maintain the interoperability among heterogeneous content management systems, federation and networking of different content organization systems are necessary in order to facilitate topic-based content retrieval and exchange of content in B2B interactions.

Global Content Management may have very different manifestations. In the area of Cultural Content Management, for instance, cultural heritage technologies have developed in order to build up digital libraries, digital archives and digital museums.

Other applications of Global Content Management systems are:

- ePublishing (single source methodologies)
- eLearning (managing teaching content
- Cyber Science (Collaborative Content Creation)
- Digital Cities and other Virtual Communities projects.

On the pragmatic level of maintaining content management systems we observe similar problems as on the level of knowledge management, that a corporate

culture of knowledge sharing has to be developed and nurtured, that special communicative and informational skills are needed to share knowledge across cultures and that the dynamic changes in content require a management philosophy that is fully cognizant of the daily implications of these constant changes.

Translation resources such as translation memories and other aligned corpora, multilingual terminological resources, reference resources, etc. are typical examples of content that needs to be managed in such global action spaces.

6. Outlook

On the technological level a number of enabling technologies for global content management have emerged that are converging into Semantic Web technologies. Intelligent information agents are integrated into such systems. They are combined with knowledge organization systems (in particular multilingual ontologies). Semantic interoperability has also become a major field of research and development in this respect.

In the field of the so-called content industry different business models have developed that could not be more diverse: on the one hand open source and open content approaches are rapidly gaining momentum, also facilitated by maturing Linux-based applications. On the other hand national, regional and international legislation concerning intellectual property rights is becoming more and more strict and global players are buying substantial portions of cultural heritage for digitisation and commercial exploitation that might eventually endanger the public nature of cultural heritage.

Epistemological issues of global content management will have to be addressed, as well as best practices to be studied in detail in order to develop advanced methods for these complex management tasks. Managing cultural diversity in a dynamic market with rapidly changing consumer interests and preferences, with new technologies to be integrated, also requires a strategy for sustainable teaching and training initiatives (based on knowledge management teaching and training initiatives) in this fascinating field.

7. References

Davenport, Thomas H./Prusak, Laurence (1998). *Working Knowledge. How Organizations Manage What They Know*. Boston: Harvard Business School Press

Hoffmann, Cornelia/Mehnert, Thorsten (2000). „Multilingual Information Management at Schneider Automation“ Robert C. Sprung (Hrsg.). *Translating Into Success. Cutting-edge strategies for going multilingual in a global age*. (pp. 59-79) Amsterdam/Philadelphia: John Benjamins

Holden, Nigel J. (2002). *Cross-cultural Management. A Knowledge Management Perspective*. Harlow: Pearson

Nonaka, Ikujiro/Takeuchi, Hirotaka (1995). *The Knowledge-Creating Company*. Oxford University Press

Popper, Karl (1972). *Objective Knowledge. An Evolutionary Approach*. London: Routledge

Tiwana, Amrit (2000). *The Knowledge Management Toolkit. Practical Techniques for Building a Knowledge Management System*. Upper Saddle River: Prentice Hall

TFPL (1999). *Skills for Knowledge Management: building a knowledge economy*. London: TFPL

Trompenaars, Fons/Hampden-Turner, Charles (1993/2001). *Riding the Waves of Culture. Understanding Cultural Diversity in Business. 2nd edition.* London: Nicholas Brealey Publishing

Wright, Sue Ellen/Budin, Gerhard (comp.) (1997, 2001). *Handbook of Terminology Management*. 2 volumes. Amsterdam/Philadelphia: John Benjamins

Supports

Korterm, KAIST

Co-operating Organisations

ISO/TC37, Korterm, Infoterm, EAFTerm,
ISO/TC37/SC4

The Workshop Programme

14:30-14:45	Opening Address, Introduction and Summarization ----- *Christian Galinski, Key-Sun Choi*
14:45-15:00	General View of TC37/SC4 ----- *Laurent Romary*
15:00-15:20	General Methodology for TC37/SC4 ----- *Nancy Ide*
15:20-15:40	Terminology of Language Resources ----- *Klaus-Dirk Schmitz*
15:40-15:55	OpenNetTerminologyManager - a Web and Standards based OpenSource Terminology Management Tool ----- *Klemens Waldhör*
15:55-16:05	An XML Markup Language Framework for Lexical Databases Environments: the Dictionary Markup Language ----- *Mathieu Mangeot-Lerebours, Frédéric Andres*
16:05-16:10	Towards a generic architecture for Lexicon Management ----- *Cristina Vertan, Walther Von Hahn*
16:10-16:25	Management of Language Resources with Metadata ----- *Peter Wittenburg, Daan*

16:25-16:45	***Discussion***
16:45-17:00	***Coffee Break***
17:00-17:15	Towards Multimodal Content Representation ----- *Harry Bunt, Laurent Romary*
17:15-17:30	Where will the Standards for Intelligent Computer-Assisted Language Learning Come From? ----- *Lars Borin*
17:30-17:50	Standards for the Localization Industry ----- *Alan Melby*
17:50-18:05	Personal Names in Unrestricted Chinese Texts: Nature and Identification ----- *Benjamin K.TSOU, Lawrence Y.L.Cheung*
18:05-18:10	Changes in the Etymological Type of New Terminology in Japanese - The Decrease of Sino-Japanese and Increase of Alphabetical Terms- ----- *Takehiro Shioda*
18:10-18:15	A Corpus-based Approach to Term Bank Construction ----- *Bai Xiaojing, Hu Junfeng, Chen Yuzhong, Yu Shiwen*
18:15-18:35	***Discussion***

Workshop Organisers

- Laurent Romary, Laboratoire Loria, France
- Christian Galinski, Infoterm, Austria
- Nancy Ide, Vassar College, USA
- Key-Sun Choi, KAIST, Korterm, Korea

Workshop Programme Committee

- Gerhard Budin, University of Vienna, Austria
- Nicoletta Calzolari, CNRS, Pisa, Italy
- Key-Sun Choi, KORTERM, KAIST, Korea
- Yuzuru Fujiwara, National Center for Industrial Information, Tokyo, Japan
- Christian Galinski, Infoterm, Austria
- Koiti Hasida, Cyber Assist Research Center, Tokyo, Japan
- Gerhard Heyer, Leipzig University, Germany
- Isahara Hitoshi, CRL, Japan
- Junfeng Hu, Peking University, China
- Churen Huang, Academia Sinica, Taiwan
- Nancy Ide, Vassar College, USA
- Yeon-Bae Kim, Human Science Division, NHK, Japan
- Jong-Hyeok Lee, Postech, Korea
- Fang Qing, CNIS, China
- Laurent Romary, Laboratoire Loria, France
- Klaus-Dirk Schmitz, Fachhochschule Koeln, Germany
- Takehiro Sioda, NHK Broadcasting Culture Research Institute, Japan
- Virach Sornlertlamvanich, NECTEC, Thailand
- Tokunaga Takenobu, TIT, Japan
- Benjamin Tsou, City University of Hong Kong
- Sue-Ellen Wright, Kent State University, USA
- Shiwen Yu, Peking University, China
- Antonio Zampolli, CNRS, Pisa, Italy

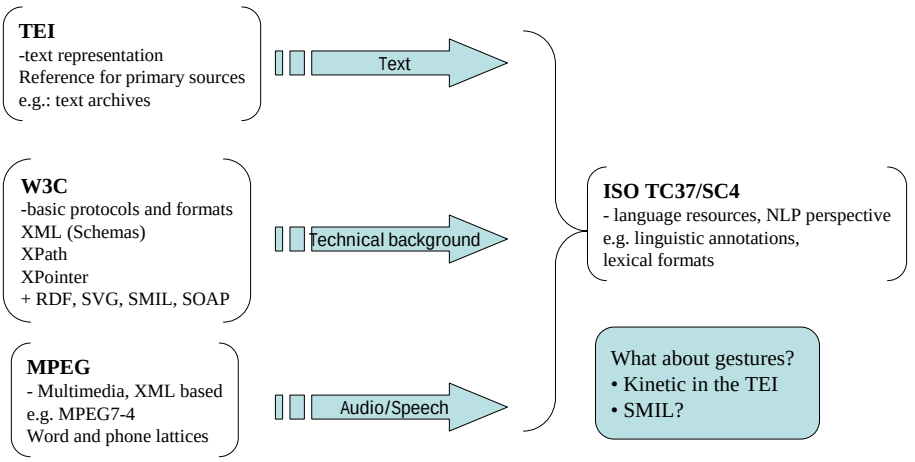
Table of Contents

Organization

- Secretary: Key-Sun Choi
- Provisional Chair: Laurent Romary
- International Advisory Committee
  - Permanent Chair: Prof. Antonio Zampolli

SC4 - environment

SC4 and other standardizing bodies



Possible sources for SC4

- Eagles, Mate
- Isle
  - Meta-data
  - Multilingual lexica
- OLIF
- OLAC
- MMA?

TC37/SC4 Work Items

- WI-1: Linguistic annotation framework
- WI-2: Linguistic resource documentation
- WI-3: Structural content representation scheme
- WI-4: Multimodal content representation sheme
- WI-5: Discourse level representation scheme
- WI-6: Multilingual text representation scheme

WI-1

- Linguistic annotation framework
  - Basic mechanisms and data structures for linguistic annotation and representation [data architecture]
    - Structural nodes and information units
    - Data category specification
    - Methods and principles for the design of an annotation scheme
    - Linking mechanisms
    - Feature Structures
  - Possible sources:
    - TMF, iso12620-revised, Mate (general methodology)
    - TEI (Linking mechanisms, feature structures)

WI-2

- Multimodal and multilingual information documentation
  - Description of a meta-data representation scheme to document linguistic information structures
    - General content description
    - Local content description
  - Possible sources:
    - Mile, OLAC
    - Data category specifications…

WI-3

- Structural content representation scheme
  - Definition of two annotation/representation schemes for morpho-syntax and syntax, to be used for annotation and interchange puposes
    - Meta-model for morpho-syntactic annotation
    - Meta-model for syntactic annotation (lexicalized grammar, elementary trees, dependancy structures)
    - Data category registry for morpho-syntactic annotation
    - Data category registry for syntactic annotation
  - Possible sources:
    - Eagles
    - TAGML
    - Working group with representatives from existing TreeBanks initiatives

WI-4

- Multimodal content representation sheme
  - Representation scheme for the integration of the semantic content of multimodal information (spoken, graphical and gestural)
    - Meta-modal for contant representation (Events, participants)
    - Data category registry for multimodal content
  - Possible sources:
    - SIGSEM working group on semantic content

WI-5

- Discourse level representation scheme
    - Meta-model for discourse and dialogue representation
    - Meta-model for discourse level annotation (e.g. reference annotation)
  - Possible sources:
    - Mate

WI-6

- Multilingual text representation scheme
  - Framework for representing bi- or multi-lingual textual information
    - Translation Memory
    - Alignment – Parallel Corpora
  - Possible sources:
    - TMX for translation memories
    - TEI based linking mechanism (or see WI-1) for Parallel texts

LREC Thematic session

- Special session on linguistic resource representation (chair K-S Choi)
  - Submitted papers - in concertation with LREC program committee
  - 30-45 minutes open discussion on main priorities for linguistic resource standardization

LREC Workshop

- Standardizing Linguistic Resources - Past activities & new prospects
  - Submitted papers
  - Round table + discussion on the definition of the work item, possible sources, etc.

Contacts

- DE: Alexander Geyken (Annotation schemes), Günter Neumann
- SP: Nuria Bel (POS/Syntax)
- NL: Harry Bunt (Semantics, SIGSEM)
- JP: Hashida Koichi

Terminology and Data Categories of Language Resources

Klaus-Dirk Schmitz
University of Applied Sciences, Cologne, Germany

Terminology Standards

Two meanings of "Terminology Standard"

- "Vocabularies" are terminology standards that contain subject-field-specific concepts and terms produced by terminology sub-committees on national, regional and international level
- Terminology-principles-and-methods standards produced by specific committees on national and international level (ISO/TC37, DIN-NAT, ...)

TC 37
Terminology and other language resources

Example 1:

Vocabularies

Principles and Methods

04/2002 Klaus-Dirk Schmitz 3

TC 37
Terminology and other language resources

- **ISO 1087-1**: Terminology - Vocabulary - Part 1
- **ISO 1087-2**: Terminology work - Vocabulary - Part 2: Computer applications

04/2002 Klaus-Dirk Schmitz 4

Contents

3 Vocabulary

3.1 Language and reality

3.1.1
object
anything perceivable or conceivable

NOTE Objects may be material (e.g. an engine, a sheet of paper, a diamond), immaterial (e.g. conversion ratio, a project plan) or imagined (e.g. a unicorn).

3.1.2
subject field
domain
field of special knowledge

NOTE The borderlines of a subject field are defined from a purpose-related point of view.

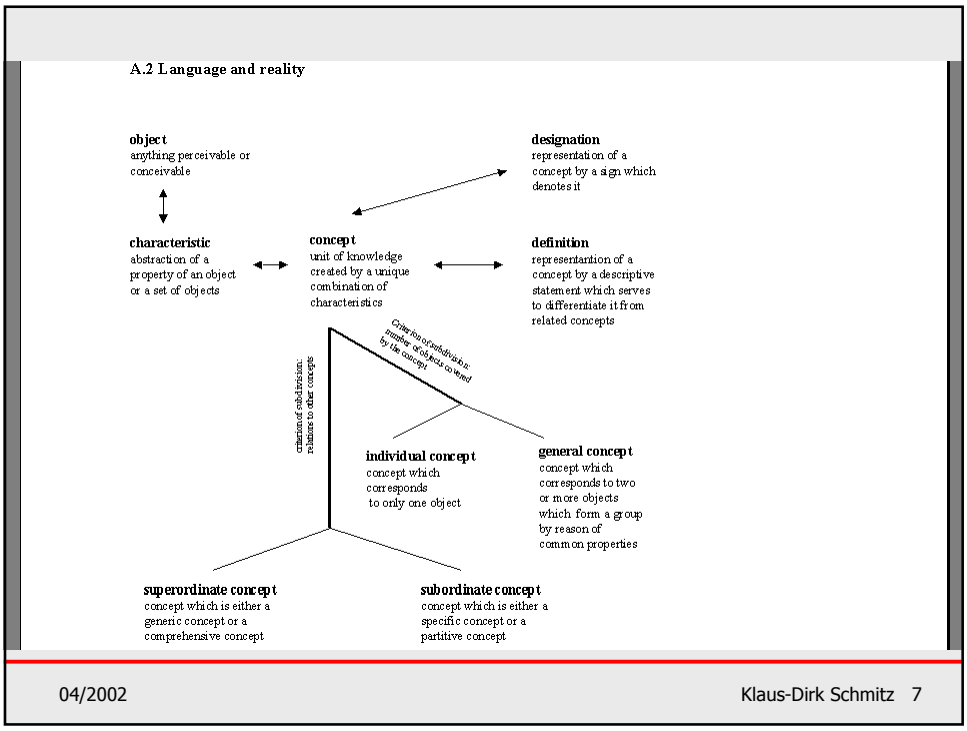
3.1.3
special language
language for special purposes
LSP
language used in a **subject field** (3.1.2) and characterized by the use of specific linguistic means of expression

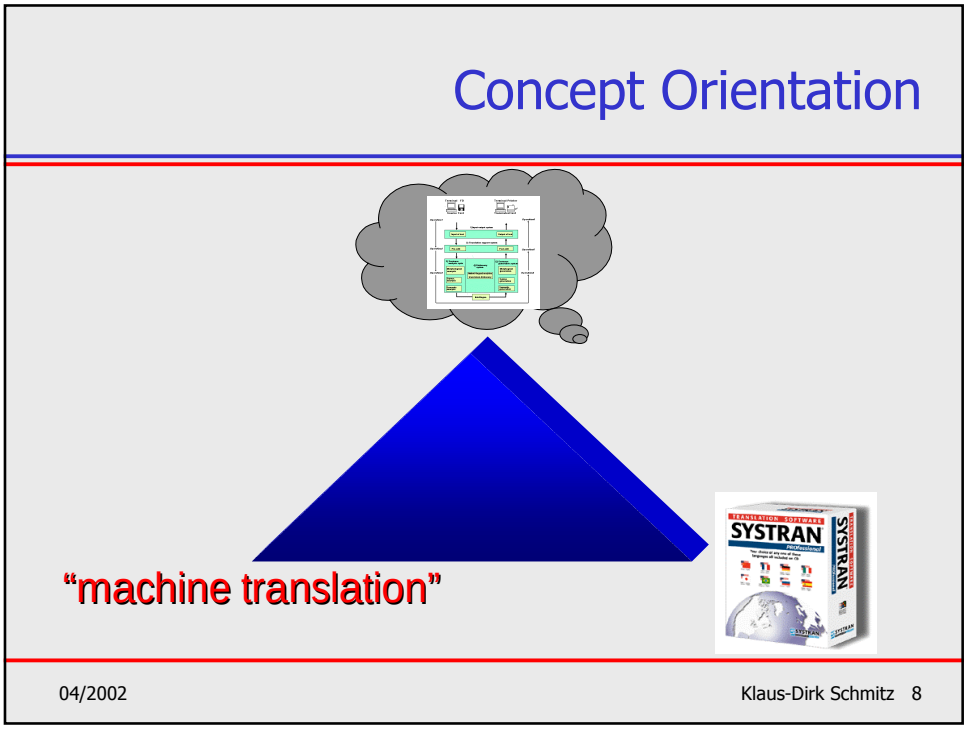
NOTE The specific linguistic means of expression always include subject-specific **terminology** (3.5.1) and phraseology and also may cover stylistic or syntactic features.

3.2 Concepts

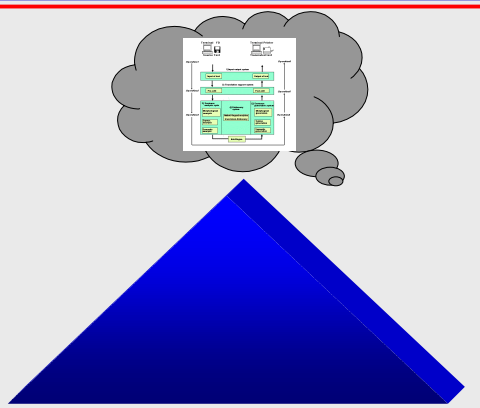
3.2.1
concept
unit of knowledge created by a unique combination of **characteristics** (3.2.4)

NOTE Concepts are not necessarily bound to particular languages. They are, however, influenced by the social or cultural background often leading to different categorizations.


A.2 Language and reality
object
anything perceivable or conceivable
designation
representation of a concept by a sign which denotes it
characteristic
abstraction of a property of an object or a set of objects
concept
unit of knowledge created by a unique combination of characteristics
definition
representation of a concept by a descriptive statement which serves to differentiate it from related concepts
Criterion of subdivision: number of objects covered by the concept
criterion of subdivision: relation to other concepts
individual concept
concept which corresponds to only one object
general concept
concept which corresponds to two or more objects which form a group by reason of common properties
superordinate concept
concept which is either a generic concept or a comprehensive concept
subordinate concept
concept which is either a specific concept or a partitive concept
04/2002
Klaus-Dirk Schmitz 7


Concept Orientation
"machine translation"
SYSTRAN
04/2002
Klaus-Dirk Schmitz 8

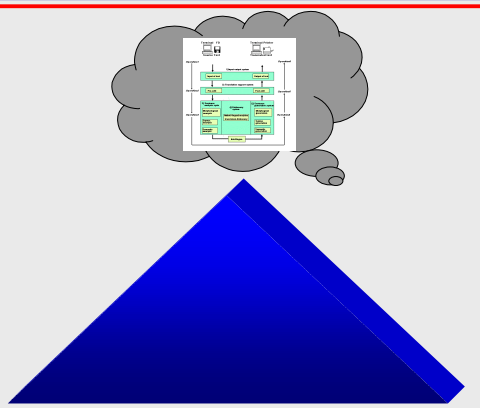
Concept Orientation



"machine translation" object

04/2002 Klaus-Dirk Schmitz 9

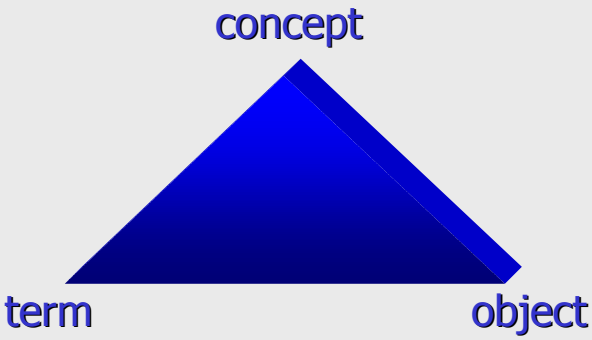
Concept Orientation



term object

04/2002 Klaus-Dirk Schmitz 10

Concept Orientation



Concept Orientation

object — Any part of the perceivable or conceivable world
Objects may be material (e.g. engine) or immaterial (e.g. magnetism)

concept — Unit of thought made up of characteristics that are derived by categorizing objects having a number of identical properties
Concepts are not bound to particular languages. They are, however, influenced by social or cultural background

term — Designation of a defined concept in a special language by a linguistic expression
A term may consist of one or more words

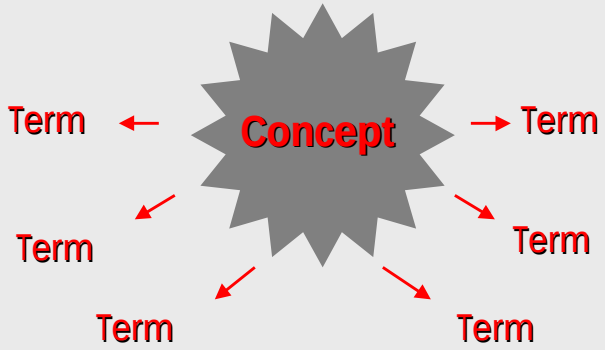
Concept Orientation

- All terminological information pertaining to *one* concept including all terms (designing this concept) in all languages and all descriptive and administrative data must be handled as *one* terminological unit.

04/2002 Klaus-Dirk Schmitz 13

Terminological Entry

Graphic adopted from Sue Ellen Wright

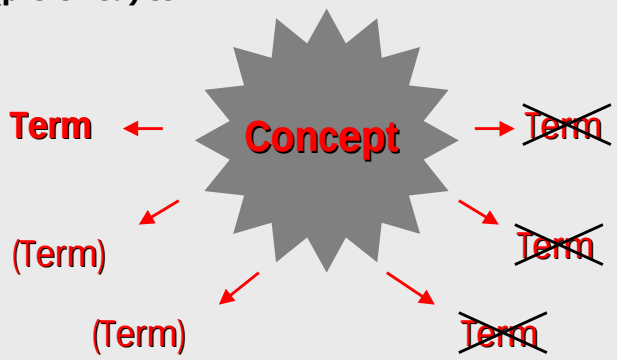


04/2002 Klaus-Dirk Schmitz 14

Terminological Entry

In standardized terminology:
only one (preferred) term !

Graphic adopted from
Sue Ellen Wright

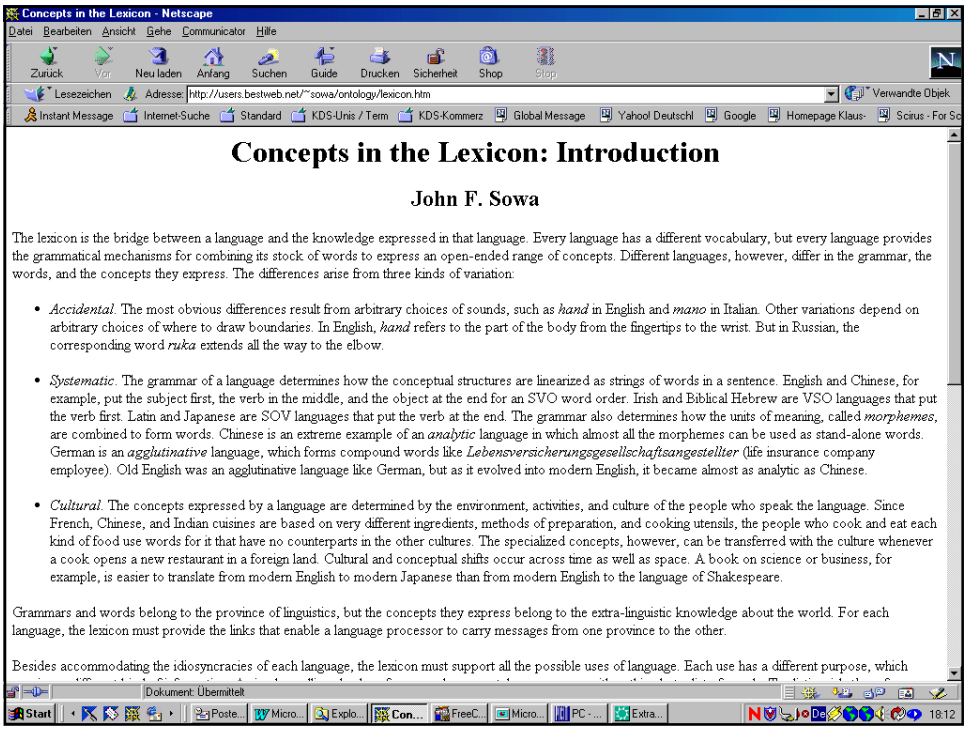


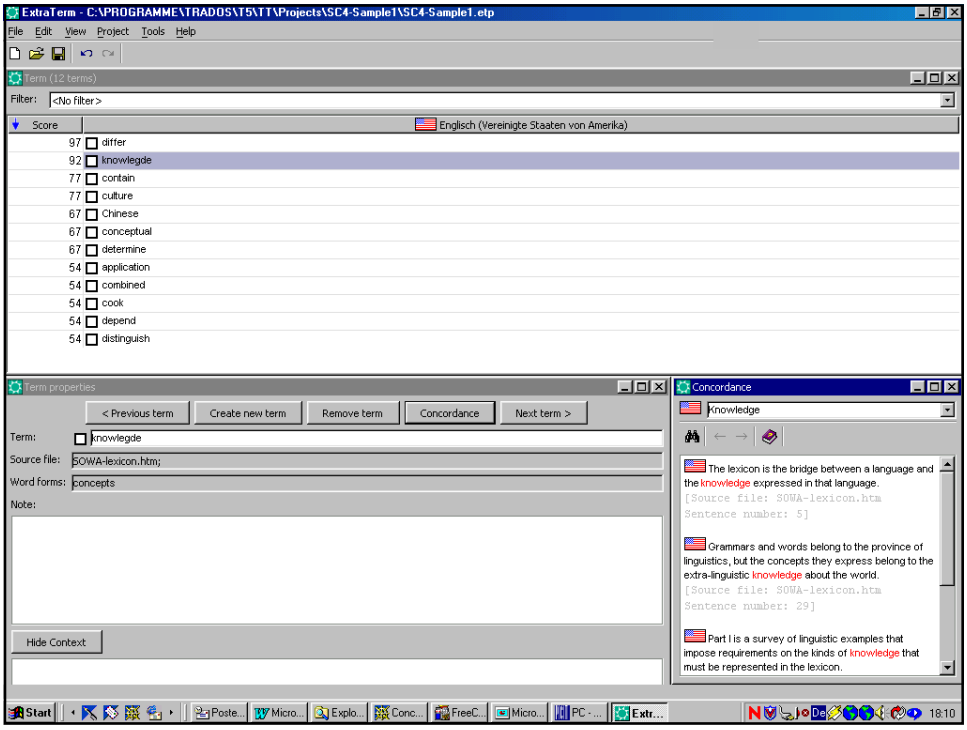
04/2002 Klaus-Dirk Schmitz 15

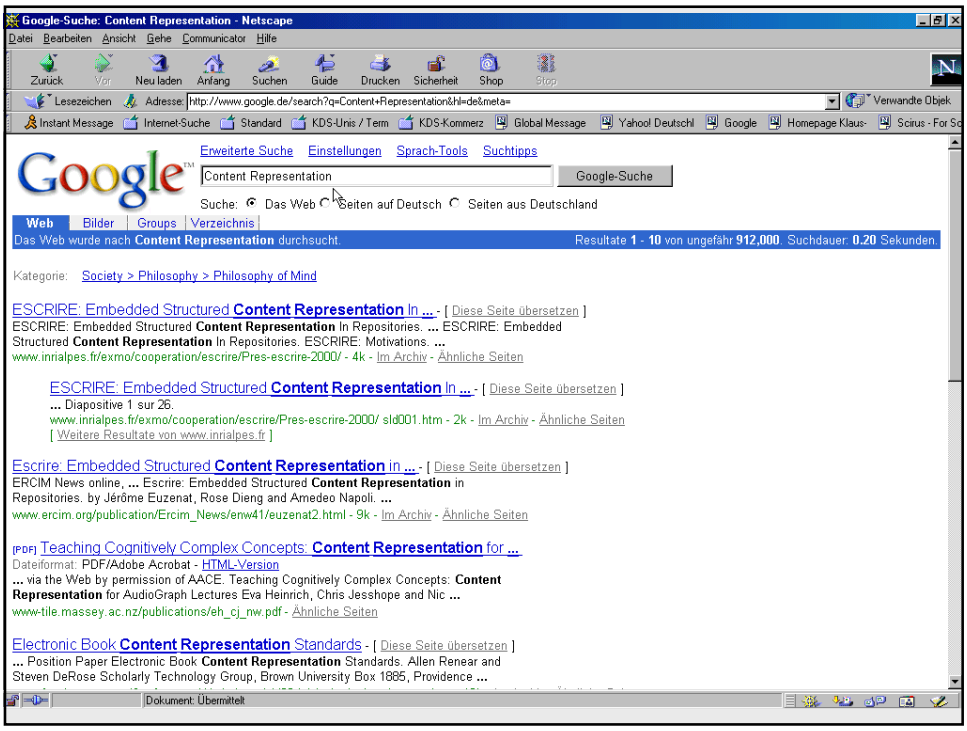
TC 37 / SC 4

- It is important to develop a standard for the terminology used in SC 4, i.e. the **terminology of language resources and language processing !**
- Preferably as the **first** work item in SC 4
- Data analysis material: LSP texts; websites
- Data analysis method: term extraction / term mining

04/2002 Klaus-Dirk Schmitz 16


Concepts in the Lexicon - Netscape
Datei Bearbeiten Ansicht Gehe Communicator Hilfe
Zurück Vor Neu laden Anfang Suchen Guide Drucken Sicherheit Shop Stop
Lesezeichen Adresse: http://users.bestweb.net/~sowa/ontology/lexicon.htm
Concepts in the Lexicon: Introduction
John F. Sowa
The lexicon is the bridge between a language and the knowledge expressed in that language. Every language has a different vocabulary, but every language provides the grammatical mechanisms for combining its stock of words to express an open-ended range of concepts. Different languages, however, differ in the grammar, the words, and the concepts they express. The differences arise from three kinds of variation:
Accidental. The most obvious differences result from arbitrary choices of sounds, such as hand in English and mano in Italian. Other variations depend on arbitrary choices of where to draw boundaries. In English, hand refers to the part of the body from the fingertips to the wrist. But in Russian, the corresponding word ruka extends all the way to the elbow.
Systematic. The grammar of a language determines how the conceptual structures are linearized as strings of words in a sentence. English and Chinese, for example, put the subject first, the verb in the middle, and the object at the end for an SVO word order. Irish and Biblical Hebrew are VSO languages that put the verb first. Latin and Japanese are SOV languages that put the verb at the end. The grammar also determines how the units of meaning, called morphemes, are combined to form words. Chinese is an extreme example of an analytic language in which almost all the morphemes can be used as stand-alone words. German is an agglutinative language, which forms compound words like Lebensversicherungsgesellschaftsangestellter (life insurance company employee). Old English was an agglutinative language like German, but as it evolved into modern English, it became almost as analytic as Chinese.
Cultural. The concepts expressed by a language are determined by the environment, activities, and culture of the people who speak the language. Since French, Chinese, and Indian cuisines are based on very different ingredients, methods of preparation, and cooking utensils, the people who cook and eat each kind of food use words for it that have no counterparts in the other cultures. The specialized concepts, however, can be transferred with the culture whenever a cook opens a new restaurant in a foreign land. Cultural and conceptual shifts occur across time as well as space. A book on science or business, for example, is easier to translate from modern English to modern Japanese than from modern English to the language of Shakespeare.
Grammars and words belong to the province of linguistics, but the concepts they express belong to the extra-linguistic knowledge about the world. For each language, the lexicon must provide the links that enable a language processor to carry messages from one province to the other.
Besides accommodating the idiosyncracies of each language, the lexicon must support all the possible uses of language. Each use has a different purpose, which
Dokument: Übermittelt
Start
18:12


ExtraTerm - C:\PROGRAMME\TRADOS\T5\TT\Projects\SC4-Sample1\SC4-Sample1.etp
File Edit View Project Tools Help
Term (12 terms)
Filter: <No filter>
Score
Englisch (Vereinigte Staaten von Amerika)
97 differ
92 knowlegde
77 contain
77 culture
67 Chinese
67 conceptual
67 determine
54 application
54 combined
54 cook
54 depend
54 distinguish
Term properties
< Previous term
Create new term
Remove term
Concordance
Next term >
Term: knowlegde
Source file: SOWA-lexicon.htm;
Word forms: concepts
Note:
Hide Context
Concordance
Knowledge
The lexicon is the bridge between a language and the knowledge expressed in that language.
[Source file: SOWA-lexicon.htm
Sentence number: 5]
Grammars and words belong to the province of linguistics, but the concepts they express belong to the extra-linguistic knowledge about the world.
[Source file: SOWA-lexicon.htm
Sentence number: 29]
Part I is a survey of linguistic examples that impose requirements on the kinds of knowledge that must be represented in the lexicon.
Start
18:10



TC 37 / SC 4

- It is important to use a web-based tool for managing terminology
- This tool must follow the basic principles of terminology management and the specific needs of ISO technical (sub-)committees standardizing the terminology of a specific domain
- In order to speed up the work between and in meetings

TC 37

Terminology and other language resources

Example 2:

Vocabularies

Principles and Methods

04/2002 Klaus-Dirk Schmitz 21

TC 37

Terminology and other language resources

- **ISO 12620:** Computer Applications in Terminology - Data categories
- **ISO 12200:** Computer Applications in Terminology - Machine-Readable Terminology Interchange Format (MARTIF) - Negotiated Interchange
- **ISO DIS 16642:** Computer applications in terminology – Meta model for representing terminological data collections / Terminology mark-up framework

04/2002 Klaus-Dirk Schmitz 22

ISO 12620 (Data Categories)

- Inventory of more than 200 data categories used in terminological data collections:
  - A.1 term
  - A.2 term-related information
  - A.3 equivalence
  - A.4 subject field
  - A.5 concept-related description
  - A.6 concept relation
  - A.7 conceptual structures
  - A.8 note
  - A.9 documentary language
  - A.10 administrative information
  - Annex B (informative): Bibliographical data categories

ISO 12620 (Data Categories)

A.2.2.1 part of speech

NONADMITTED TERM1**: grammatical category**

NONADMITTED TERM2**: word class**

DESCRIPTION: A category assigned to a word based on its grammatical and semantic properties.

PERMISSIBLE INSTANCES: Examples of parts of speech commonly documented in terminology databases can include:

a) noun
b) verb
c) adjective

- On the basis of a study and analysis of a great variety of practical applications; can be amended

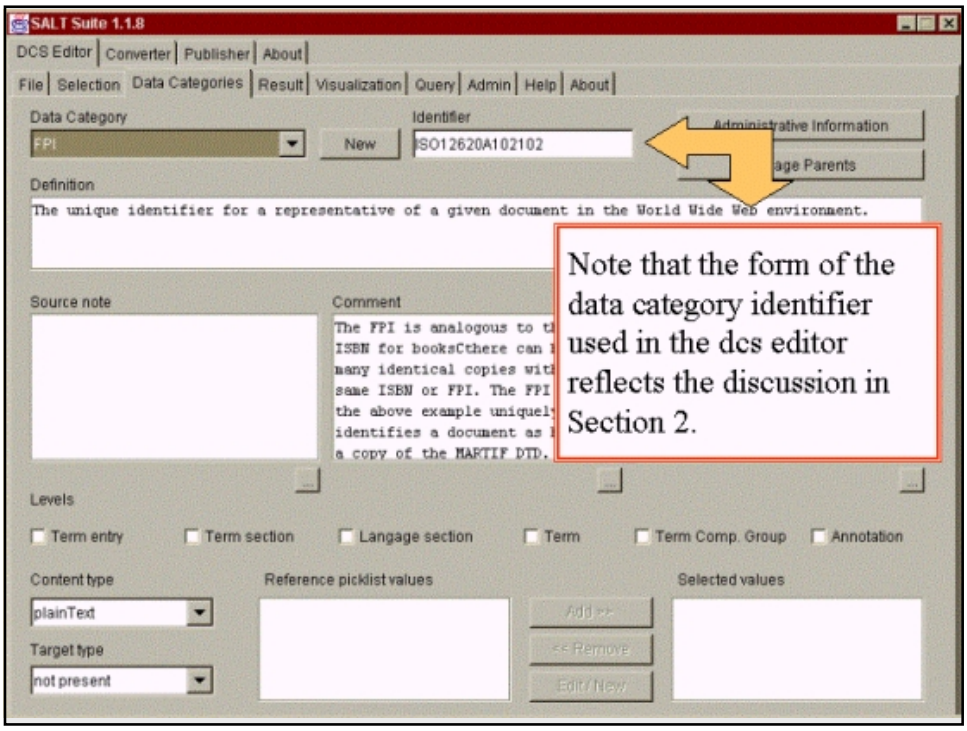
ISO 12620 new

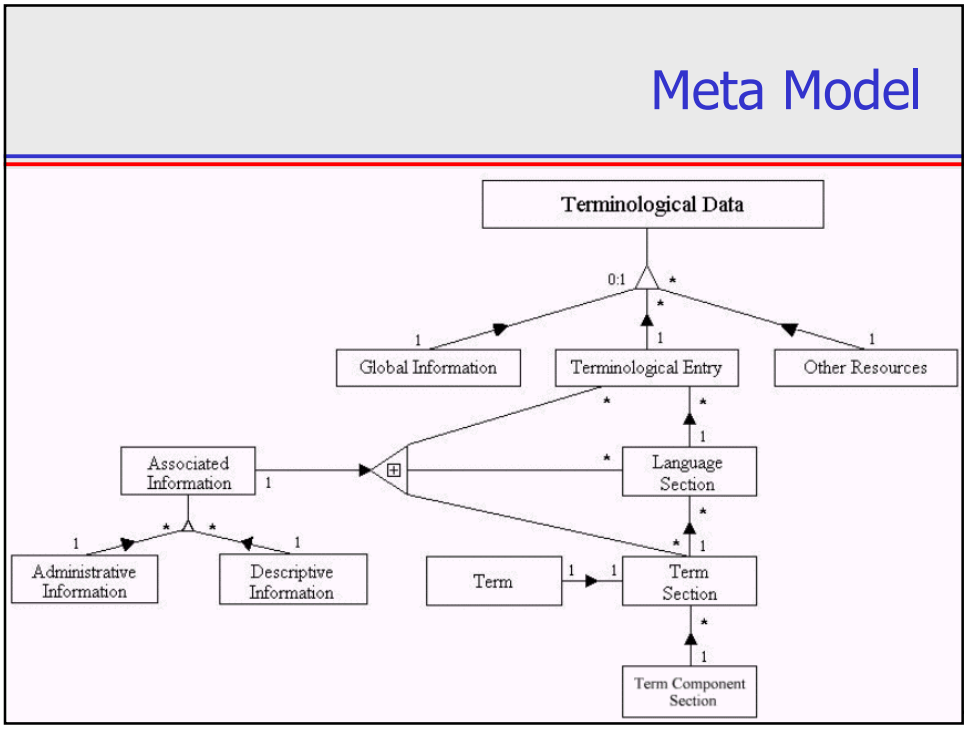
Metadata Registry

- contains terms that describe database fields
- for describing and comparing databases
- for human use
- "concept-oriented" but referring to objects (fields) that are IT representations of (real) objects/concepts
- ISO JTC1/TC32 provides a standard for metadata registries

ISO 12620 new

- Converting ISO 12620:1999 data category description into metadata registry format
- Using the DCS-Editor, developed within the framework of the SALT Project, for the description of the data categories
- Create the list of datCats and the description of datCats directly by the DCS-Editor as a normative annex of the new ISO 12620
- The body defines the (metadata) description format


SALT Suite 1.1.8
DCS Editor
Converter
Publisher
About
File
Selection
Data Categories
Result
Visualization
Query
Admin
Help
About
Data Category
FPI
New
Identifier
ISO12620A102102
Administrative Information
Definition
The unique identifier for a representative of a given document in the World Wide Web environment.
Source note
Comment
Note that the form of the data category identifier used in the dcs editor reflects the discussion in Section 2.
Levels
Term entry
Term section
Langage section
Term
Term Comp. Group
Annotation
Content type
plainText
Target type
not present
Reference picklist values
Selected values
Add >>
<< Remove
Edit / New


Meta Model
Terminological Data
0:1
Global Information
Terminological Entry
Other Resources
Associated Information
Language Section
Administrative Information
Descriptive Information
Term
Term Section
Term Component Section

ISO 12620 new

ISO 12620 Expanded Position	ISO 12620 Name	Target	DataType	Level Restrictions	Sub-set
A.02.02	grammar				DT
A.02.02.01	part of speech		plaintext	TS , TC	DT
A.02.02.02	grammatical gender		picklist 2	TS, TC	DT
A.02.02.03	grammatical number		picklist 3	TS, TC	DT
A.02.02.04	animacy		picklist 4	TS, TC	DT
A.02.02.05	noun class				DT
A.02.02.06	adjective class				DT
A.02.02.07.T	grammatical valency		plainText	TS	DT
A.02.02.08.T	inflection			TS, TC	DT

04/2002 Klaus-Dirk Schmitz 29

part of speech

Identifying and Definitional Attributes

Data Element ID:	ISO12620A020201 **Version No :** 1
Data Element Name:	part of speech
Type :	Data Element
Status :	Current 12-DEC-1999
Admitted Name:	
Non-admitted Name 1:	grammatical category
Non-admitted Name 2:	word class
Definition:	A category assigned to a word based on its grammatical and semantic properties.
Source-related Comment:	
Concept-related Comment:	
Example:	
Dictionary ID :	A.2.2.1

Relational and Representational Attributes

Datatype :	Plain Text (or user-defined picklist)
Representational Form :	??
Representation Layout :	??
Minimum Size :	
Maximum Size :	
Guide for Use :	In a given database, it is wise to configure this category as a user-defined picklist in order to avoid the proliferation of alternate forms, etc. For a global interchange format, however, it is important to specify this item as plainText because it is impossible to predict all the possible options that might occur in all possible language combinations
Validation Rules :	

Data Domain Details

Examples of parts of speech commonly documented in terminology databases can include:

Permissible value	Domain Meaning Definition Text	Example
noun	A word that refers to a person, place, thing, event, substance or quality.	'Doctor', 'tree', 'party', 'coal' and 'beauty' are all nouns.
verb	A word or phrase that describes an action, condition or experience.	The words 'run', 'keep' and 'feel' are all verbs.
adjective	A word that describes a noun or pronoun.	'Big', 'boring', 'purple', 'quick', 'obvious' and 'silvery' are all adjectives.
etc.		

TC 37 / SC 4

- It is important to develop a standard for data categories used in typical "SC 4 applications"
- May-be with different parts for different types of language resources (NLP lexica, texts, speech etc.)
- Data analysis material: language data collections
- Data analysis method/tool: (mod.) SALT's DCS-Editor

Thank you for your attention!

- MARTIF, ISO 12620 Data Categories, MSC, Meta-Model, TBX: www.ttt.org
- SALT Project: www.loria.fr/projets/SALT

klaus.schmitz@fh-koeln.de

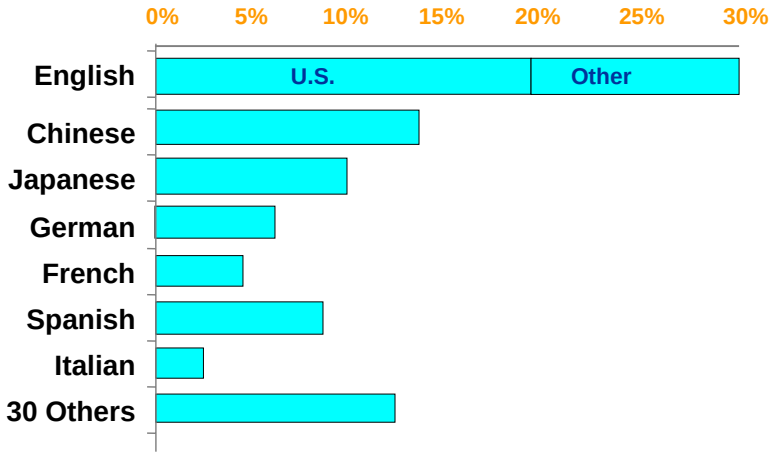
2002 May 28

Language Resource Representation and Management Standards for the Localization Industry

Alan K. Melby
<akm@byu.edu>

ttt.org -translation, theory & technology-

Tidbit: Internet Users by Native Language - 2005



Source: IDC Internet Commerce Market Model, V7.1 (c/o Rose Lockwood)

Overview of Presentation

A. Definition of localization as part of GIL
B. Brief history of LISA and OSCAR
C. Layers of Localization standards
D. XLIFF for text and source code
E. TMX for translation memory exchange
F. TBX and OLIF for terminology
G. Unresolved issue: segmentation

[A] GIL

- Globalization (G11N)
- Internationalization (I18N)
- Localization (L10N)

[A.1] Globalization

- Globalization (G11N)
  - Globalization is the business process of taking products and services into various new markets around the globe
  - A locale is the geographic region and language of a particular market
- Internationalization (I18N)
- Localization (L10N)

[A.2] Internationalization

- Globalization (G11N)
- Internationalization (I18N)
  - Internationalization is the engineering process of generalizing a product or service so that it can handle multiple languages and cultural conventions

Localization (L10N)

[A.3] Localization

- Globalization (G11N)
- Internationalization (I18N)
- Localization (L10N)
  - Localization is the cross-cultural communication process of preparing locale-specific versions of a product or service and consists of translation of textual material and adaptation of non-textual material.

[B] Brief History of LISA and OSCAR

- LISA (Localization Industry Standards Association) – see http://www.lisa.org/info/about.html
- OSCAR (Open Standards for Container/content Allowing Re-use)
- OSCAR is a LISA special interest group for language resource data standards

Localization-related Technologies

- Text Representation (Unicode and XML)
- Translation/Localization Container (TLC)
- Translation Tools (specialized)
  - Segmentation, alignment, encapsulation
  - Termbase setup or enrichment
  - Translation memory and machine translation
  - Terminology lookup
  - Missing segment and markup check
  - Term check (consistency, false friends, and variants)

[C] Layers of Localization Standards

- Unicode
- XML (including language/locale ids)
- XLIFF
- TMX
- TBX and OLIF

[D] XLIFF

- XLIFF is a format to store extracted text and carry the data from one step to another in the localization process
- see http://www.opentag.com/xliff.htm for more information

[E] TMX

- The purpose of TMX is to allow easier exchange of translation memory data between tools and/or translation vendors with little or no loss of critical data during the process.
- See http://www.lisa.org/tmx/ for more information

[F] TBX and OLIF

- TBX and OLIF allow the representation and exchange of terminological data, with a focus on human-oriented and NLP-oriented data, respectively
- See http://www.lisa.org/tbx/ for more on TBX
- See http://www.olif.net/ for more on OLIF

[G] Unresolved Issue

- Segmentation of words, sentences, and other linguistic units has not yet been standardized for the localization industry
- This means that word counts are not standard
- This also means that translation memory lookup may not detect some matches

OpenNetTerminologyManager- a Web and Standards based OpenSource Terminology Management Tool

Klemens Waldhör*

* Friedrichstr. 17, 90574 Roßtal, Germany, dr.klemens.waldhoer@waldhor.com

Abstract

OpenNetTerminologyManager is a privately started Open Source project which aims at developing a freely available pure web based concept terminology management system. It runs with any browser supporting JavaScript. The server side requires MySQL, Apache Web Server and Perl. The system is currently available through sourceforge.net at http://openwebterm.sourceforge.net. OpenNetTerminologyManager supports different terminological models. A version which is based on MARTIF has been implemented.

1. Introduction

Through the years the world has seen the attempt to establish several different terminology standards starting from MARTIF, Geneter to TBX, XLT (SALT), TMF (ISO 16642) and so on. The author himself was part of one of the older efforts which started 1990 where within the MULTILEX project a first try was made to create a standard exchange description for areas like mono- and multilingual dictionaries, machine translation etc. The basic idea there was to use SGML as the description language. This was followed up in projects like EAGLES, Otelo (OLIF) etc. In parallel other attempts have been made like Geneter. Sometimes one is really puzzled how creative the terminology community is in inventing new ideas and standards. Often it is really hard to follow what is going on. This is the one side of the coin. On the other side the industry uses "**quasi standards**" like the export format used in MultiTerm™ from Trados™. Several products of competitors like TermStar™, UniTerm™ and others provide import and export features from and into the MultiTerm™ format, simply because MultiTerm™ is the market leader in this area. Otherwise getting into this application field for new systems is nearly impossible as most customers either use MultiTerm™ or at least provide their data in this format.

Interestingly enough Open Source terminology software was never really part of the terminology game, in contrast to other areas like web servers where open source software like Apache is the dominating software (60 % of the world web server market). If one searches for "open source terminology management" in Yahoo and inspects the returned results in detail there are only two other relevant matches, the ForeignDesk and OpenGALEN match. In the last half year Lionbridge has made its software **ForeignDesk** available through open source. Another notable effort is **RosettaWerks** which deals implementing a set of tools for the localisation process.

But what is really missing is a terminology tool which is available on several operating systems (not just Windows™) and can be used through the web and itself is built on free available software. This is not the place to discuss the advantages of the open source model. A lot of discussion is going on this area, but I just want to add that one clearly has to distinguish the open source model from models which are offered by software suppliers where one can get the executables for free, but has no access to the source code. Several providers of terminology software supply down-graded or full versions of their tools – mainly viewers - e.g. UniLex™ from Acolada GmbH, but this does not bring any advantage to the user as he still relies on the provider to fix bugs etc. In addition it is hard to check if there are any hidden traps in the software. As professional terminology management contains company or customer information security aspects and the ability to check this will be an important aspect of choosing a system in the future. Based on this observations – and being also a fan of the open source community - I started developing a terminology management software which should fill this gap.

2. OpenNetTerminologyManager Terminology Model

The basic idea of the system architecture is the capability to support **different terminology models**. The user should have the option either to create his own model or to adapt an existing model by sub-classing it or adding his own fields. It should also be possible to keep track with on-going changes in the standardisation community. This has been realised in the system in the following way: attributes (elements) of the terminology model are not directly mapped to database tables, but this information is kept in a specific column where the structure can be freely defined. The actual mapping of these content of this column to attributes is defined in **model files**. Each database represents one model. The advantage of this approach is a) that it keeps the number of databases tables to a minimum, b) as a result the system is quite fast in searching and reading entries and c) adaptations of attributes can be made easily.

The basic OpenNetTerminologyManager approach is **concept oriented** as it used in most modern terminology systems. In this approach a concept corresponds to one meaning of a word. The language specific parts of a concept are called "**language terms**" or simply "**terms**". Each concept is tagged with an unique identifier, while each term related to the concept uses the concept identifier plus a language identifier and an internal term counter as identifier.

Example: The German term "**Birne**" (three meanings: Glühbirne, Frucht, Kopf = bulb, pear, nut) will be represented by creating three concepts (Figure 2):

a) one with the meaning of "Frucht = fruit" and

b) one with the meaning of "Glühbirne = bulb" and

c) one with the meaning of "Kopf = head".

The **kernel** of OpenNetTerminologyManager consists of several tables:

a) A **MONOTERM** table which holds all relevant information for a term including term attributes

b) A **MULTITERM** table which links entries in the MONOTERM table to a concept and also also stores concept related attributes.

c) A **DETAILS** table which contains links from attributes to terms and concepts. This table is only used to optimise the speed when searching with attributes.

d) A **LINK** table which establishes links between either concepts or term (e.g. in order to express a relations like "synonym").

Different terminology models are now mapped to the kernel model in a **model file**. This model file defines:

The **names** (e.g. "Gender") to be used for the **attributes** of the terminology model into an internal name. This association differentiates between concept related and term related attributes.

The **values** and **forms** to be associated with a such **names**. As an example associate the attribute "Gender" with three possible values ("male", "female", "neuter") and display them in the browser as a select box.

Table 1 shows a simple section for the MARTIF model. Models can further be differentiated into two classes: "**full models**" and "**sub-models**". A sub-model is defined as a subset of attributes of a full model. This is mainly necessary if for a given model (e.g. MARTIF) only specific attributes should be shown or if specific restrictions may apply for attribute values. The system contains some additional fixed attributes like the owner of the concept, read and write accesses etc.

3. OpenNetTerminologyManager Features

The following functions are currently supported:

- Constraints between attributes can be realised with JavaScript
- New models and sub-models can be created by the user (see Figure 1).
- Attributes can be defined by the user.
- Different types for attributes like option fields, text fields, select etc. are supported.
- Multiple databases; multi-user read/write support (locking at concept level). Different right combinations can be used. Databases are either private (with user and password protection) or public.
- Partial Unicode support. Unicode characters above Ascii 255 are stored as SGML entities in the database. This will be removed once MySQL supports directly UTF8 or a similar Unicode encoding scheme. Languages like Arabic, Chinese, Japanese etc. can be used through this approach. Once a Unicode implementation of MySQL is available this representation will be changed to an internal Unicode character set.

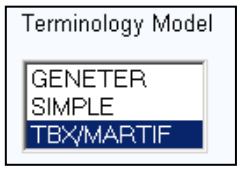


Figure 1: Models

Currently one terminology model based on MARTIF has been (partially) implemented. It normalises the XML definitions into the relational (table based) approach defined above. Others like Geneter are under way.

```
opwdetail40=Grammatical Gender
opwdetail41=Term Type
opwdetail44=Grammatical Number
…
 <tr>
   <td>
     <fieldset>
       <legend>Grammar</legend>
       <table>
         <tr>
           @tdopwdetail43=Part Of Speech?10<select!noun|verb|adjective|other
           @tdopwdetail40=Grammatical Gender?10<select!na|feminine|masculine|neuter|other
         </tr>
         <tr>
           @tdopwdetail41=Term Type?10<select!...|variant
           @tdopwdetail46=Valency?10<input
         </tr>
         <tr>
           @tdopwdetail44=Grammatical Number?10<select!na|dual|mass|other|plural|singular
           @tdopwdetail45=Animacy?10<select!animate|inanimate|other
         </tr>
       </table>
     </fieldset>
   </td>
 </tr>
```

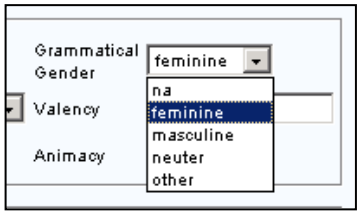


Table 1: OpenNetTerminology Manager GUI description

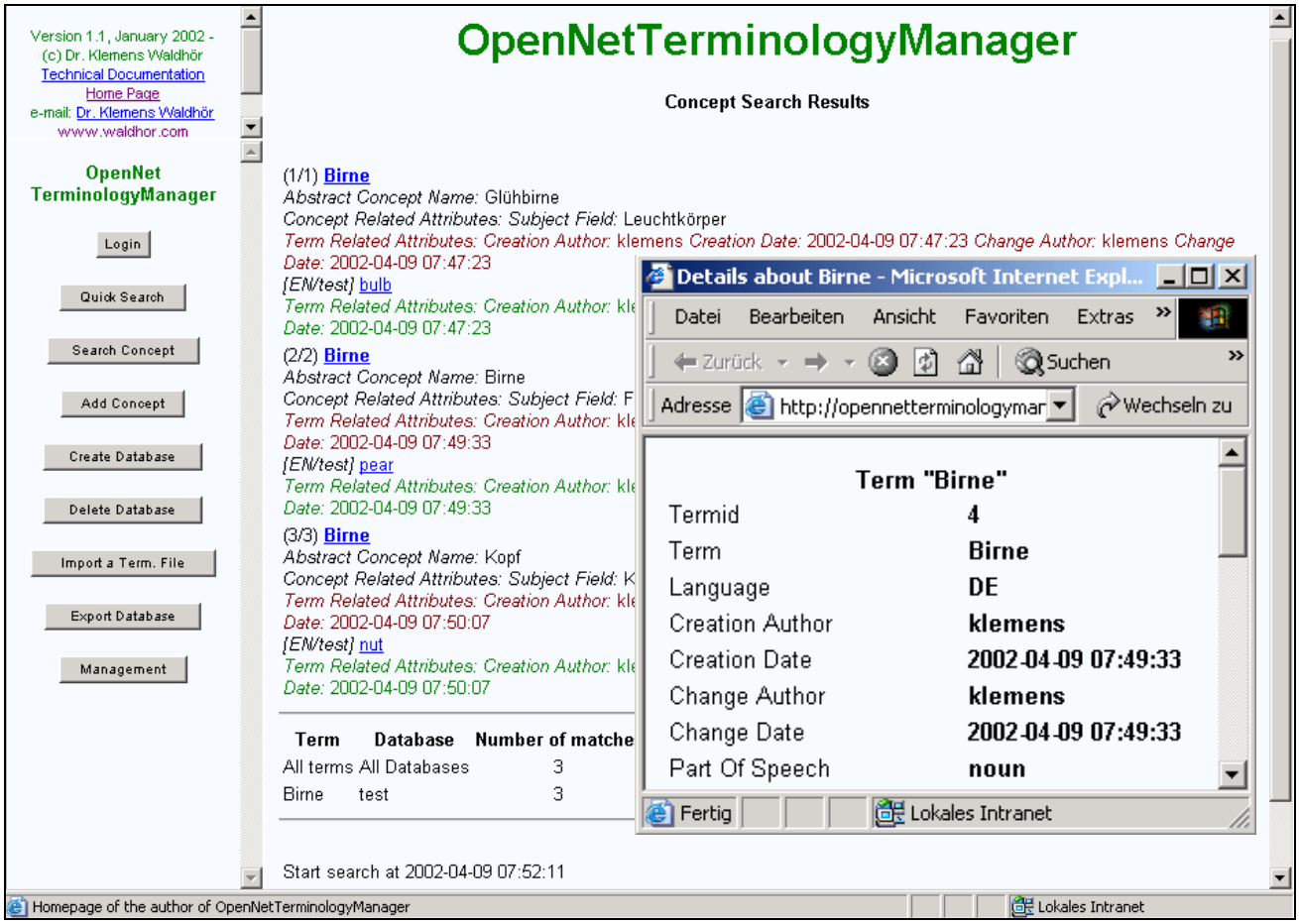
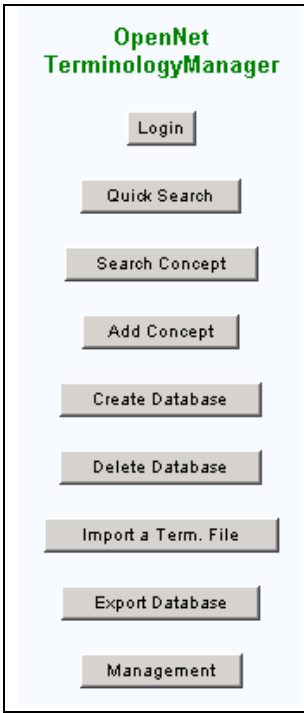


Figure 2: OpenNetTerminology Manager User Interface



Login allows the user to define default values like his preferred databases, languages, attributes to be displayed for a model, how search results should be displayed etc.

Quick Search offers a simplified search mode which simply looks up the database for a specified term independently of the language.

Search Concept supports more detailed search options like languages to be searched and displayed etc. Through this item the user also can edit concepts in the database.

Add Concept adds new concepts to the database.

The user can create new databases using **Create Database** where he also specifies the model and access rights for the database.

Databases can be deleted using **Delete Database**.

Import Term File allows existing terminology files to be imported in various formats supporting double detection during import,

while **Export Database** exports databases into various formats.

Management is mainly intended to give an overview of current system settings and databases. It also supports recreating the database structure.

Figure 3: OpenNetTerminologyManager Commands

4. The Basic User Interface of OpenNetTerminologyManager

Figure 2 shows the basic web based user interface. It consists of a main window where the results of queries etc. are shown and a navigation window (left). Optionally additional concept or term related information can be displayed in a separate browser window. Figure 3 describes the basic functions of the navigation window.

Concepts can be edited by first searching them with the **Search Concept** function and using the "Edit Mode" (not choosing "Dictionary View" option). See figure 4. Results are then displayed in a tabular like format (figure 5). Clicking on "Edit" will then display the full entry (figure 6) in an editable format. Results can also be displayed in a "Dictionary View" mode (figure 7). In this mode concepts found with the same name for a given language may optionally be collapsed into one output entry. This displays the entry in a similar way as they are show in printed dictionaries. Depending on the user search result display settings attributes will be displayed either directly in the main window as part of the entry or the term name is realized as a hyperlink and when clicking on it is displayed later in a separate browser window (figure 2). In addition the user can configure for each database which attributes should be shown. The query itself supports various search options like full text search, regular expressions, the LIKE operator etc.

5. Software requirements

OpenNetTerminologyManager requires the following software components: Perl > 5.0 (with some additional modules installed), Apache Web server or a compatible server, MySQL and a JavaScript enabled Web Browser. Tests have been done with Internet Explorer 5.0, 6.0®, Netscape® and Opera® . The system has been tested both on Windows (NT® and 2000®) and LINUX.

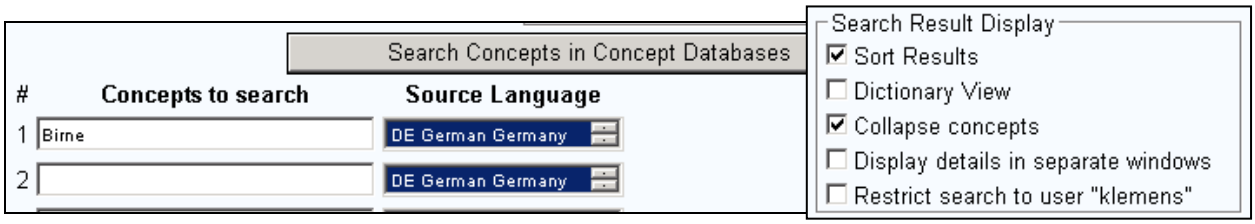


Figure 4: Searching concepts

No	ID / Concept	Source Language DE	Translation Term	Language	Database	Operation
1/1	316071404	Birne	nut	EN	meine	Edit Delete Copy Concept
2/2	284021368 Glühbirne	Birne	bulb	EN	meine	Edit Delete Copy Concept
3/3	312237593	Birne	pear	EN	meine	Edit Delete Copy Concept

Figure 5: Searching result display

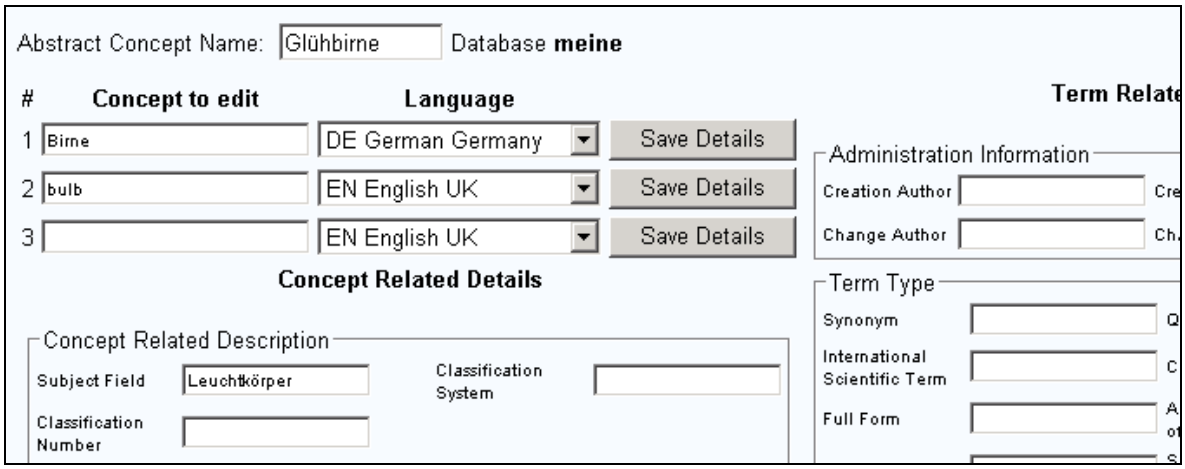


Figure 6: Editing concepts

Figure 7: Dictionary View display with no attributes displayed searching for "Planet%"

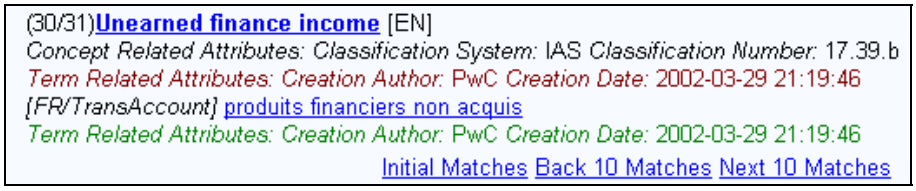


Figure 8: Result of a TransAccount terminology database full text query (searching for the term "finance") with attributes displayed.

6. Application Scenario

The TransAccount project (MLIS 5016) deals with the need for a multilingual translation system allowing the translation and interpretation between the annual accounts of a member state of the European Union (France) and IAS (International Accounting Standards) statements. Within this project the XBRL (eXtensible Business Reporting Language) IASCF taxonomy has been translated from English to French by one of the partners. The resulting 2000 concepts have been imported into a TransAccount terminology database. In addition about 2000 other general financial terms have been converted from a Geneter based format which have been produced by another partner at the start of the project. An example of the results of a query is shown in figure 7.

7. Next Steps

An important feature which is currently under development is an advanced link concept. This link concept will not only support links in the way as TBX defines them but will allow to create complex typed links between concepts and terms and databases. This will allow the user to search the databases not only as a simple term-lookup tool but to browse through it in a kind of semantic net and to find related concepts.

A concept is also developed which supports "similarity queries". It is intended to introduce a "stemming based index" by applying the Porter stemming algorithm to terms for some languages automatically (Porter, 1980). Other developments concern additional import / export formats and simplified form handling for attributes.

As there are several opens source project on mapping xml to relation databases on the way (e.g. XML-DBMS) I am currently also looking into replacing the internal structure of the database by a full xml database approach. This will heavily depend on the access speed compared to the current implementation.

8. References

Acolada. http://www.acolada.de

ForeignDesk. http://sourceforge.net/projects/foreigndesk/

OLIF. http://www.olig.net

OpenGALEN. http://www.opengalen.org/

OpenNetTerminologyManager. http://openwebterm.sourceforge.net

Porter, M.F., 1980. An algorithm for suffix stripping, *Program, 14 no. 3, pp 130-137, July 1980*

Rosettawerks. http://rosettawerks.sourceforge.net/Default.php

Sourceforge. http://sourceforge.net/

Star AG. http://www.star-ag.ch/eng/software.html

Trados. http://www.trados.com

TransAccount: http://www.transaccount.org

XML-DBMS. http://www.rpbourret.com/xmldbms/index.htm

Waldhör, K., Tesniere, B., 2002. Multilingual Terminology Database, *MLIS 5016 TransAccount Report*.

XBRL. http://www.xbrl.org

9. Acknowledgements

Thanks has to be given at this place to SourceForge which provides an excellent – and free – way to make open source projects available through the web.

An XML Markup Language Framework for Lexical Databases Environments: the Dictionary Markup Language.

Mathieu MANGEOT-LEREBOURS **Frédéric ANDRÈS**

Software Research Division, NII
Hitotsubashi, 2-1-2 Chiyoda-ku
101-8430 Tokyo, Japan
mangeot@nii.ac.jp

Introduction

Lexical data resources are growing rapidely thanks to the Internet. Unfortunately, despite numerous existing standards like TEI, MARTIF, GENELEX, EAGLES/PAROLE, etc. each resource has its own format and own structure. Furthermore, the existing lexical data is generally developed for a specific purpose and can't be reused easily in other applications.

In this paper, we intend to define a complete framework for developing multilingual lexical database for multipurpose. The framework is generic enough in order to accept a wide range of dictionary structures and proposes for manipulating heterogeneous dictionaries a set of common pointers into these structures.

We will first present the organisation of Dictionary Markup Language (DML) framework.

Then we will describe more precisely the DML language based on XML schemata.

Next, we explain how to describe dictionary macro and microstructures with the DML.

Lastly, we will explain our concept of common pointers defined in a Common Dictionary Markup (CDM) set.

1. Presentation of the DML Framework

The DML Framework described first by Mangeot-Lerebours (2001) is a complete framework for the consultation of heterogeneous dictionaries, cooperative construction of new dictionaries and communication with other lexical databases or lexical data client and supplier applications. The framework is completely generic in order to manage heterogeneous dictionaries with their own proper structures.

The consultation of heterogeneous dictionaries is possible as soon as they are encoded in XML, consultation of other resources via remote servers through API, possibility of adding pre-consultation help modules such as spell checking and morphological analysis before consultation or post-consultation modules like syntethisers, conjugation of verbs, learning drills, etc. Possibility of automatic consultation of the database via client API.

The construction of new dictionaries can be done by a community of contributors and validated by a group of head lexicographers specialists.

The management of user profiles, preferences and weights for consultation, annotation and edition of lexical data with inheritance and sharing possibilities among groups of users is also handled by the framework.

The `<database>` element describes a lexical database and lists the dictionaries that are stored in it.

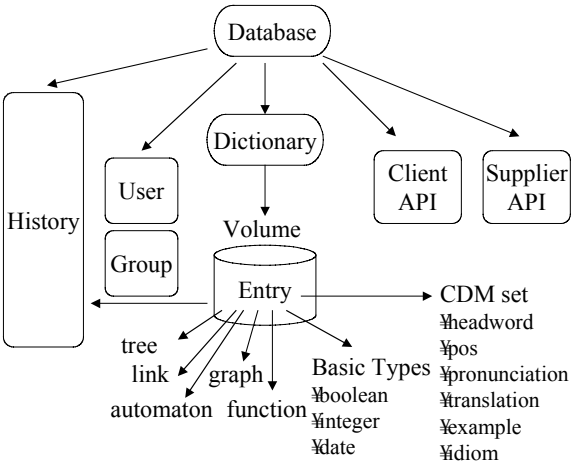


Figure 1. Logical Organisation of a Lexical Database

The `<dictionary>` element describes the metadata linked to ther dictionary. It links all the volumes of the dictionary.

The `<volume>` element describes a dictionary part. The content is principally a list of dictionary entries. For example, a bilingual bidirectional French-English dictionary will be described by only one `<dictionary>` element. The French->English entries will be in one `<volume>` element and the English->French entries in another `<volume>` element

2. The DML Language

2.1. The DML Namespace

To describe the structure of all the documents, elements, attributes and XML types, we use an XML namespace [XML Namespaces]. Our namespace is called DML for Dictionary Markup Language. The

namespace URI points to an XML schema [XML Schemas] describing the contents of the namespace. It is available online[1] to allow users to edit and validate their files online with an XML schema validator.

```
<MyElement
xmlns:dml="http://www-clips.imag.fr/
geta/services/dml">
  ...
  <dml:MyDescendant/>
  ...
</myElement>
```

Figure 2: Usage Example of the DML Namespace

2.2. DML Common Types and Attributes

For some information, we define type and attributes common to all DML elements. It allows to standardize the data. The XML schemata have originally simple predefined types. We selected and reused some in our definitions.

2.2.1. Dates and Time

Dates are represented by the `date` DML attribute of the XML schema type dateType taken from the extended format of the ISO 8601 standard.

2.2.2. Response Delay

The `delay` DML attribute of an element indicate the response delay when a request has been launched on this element.
This delay is a duration of the XML schema `durationType` type. For example, 5 seconds and 10 cents will be indicated : "5.10S".

2.2.3. Unique ID

The `id` DML attribute of an element is a unique ID in all the lexical database. It allows to create links between elements. It redefines the XML schema ID simple type.

2.2.4. Modifications History

The modifications history of an element has a unique ID. The element links to its history thanks to the DML attribute `history` that gives the value of the history ID. The type redefines the XML schema ID simple type.

2.2.5. Languages Notation

To note the various languages, we use the ISO-639-2/T (T for Terminology) [ISO98] standard that defines a 3 letter code for each language (French->fra; English->eng, Malay->msa, etc.). It is far more complete that the two letters code standard ISO-639-1. We also add our proper codes like "unl" for the UNL language. This codes list represents the lang DML type. The `lang` DML attribute is from this type.

[1] http://www-clips.imag.fr/geta/services/dml/

2.2.6. Documents Encoding

To note the encodings of the various documents in the database, we define the `encodingType`. DML type. The values are those described by the IANA (Internet Assigned Number Authority) for the encodings. These are also the values used for MIME types (Multipurpose Internet Mail Extension). Among the most used, we find ASCII on 7 bits, ISO-8859-1 on 8 bits for latin languages, Shift-Jis on 8 or 16 bits for the Japanese, UTF-8 on 8 bits for UNICODE characters, etc.

2.2.7. Status of an Element

The `status` DML attribute is used to indicate its status. The values can be among others `auto` if the element has been obtained automatically, `rough` if the element has not been revised and `revised` if so, etc.

3 DML Architecture

3.1. Macrostructure Definitions

To describe the macrostructure of our dictionaries as well as our lexical database, we use XML elements. We principally based our definitions on the LEXARD language defined by Serasset (1994) and added some information

3.1.1. Description of a Lexical Database

To describe a lexical database, we use the `<database>` element formally described inthe DML schema.
The modifications of the `<database>` element and its descendants are stored in a document linked with the `history-ref` atttribute.
We add to LEXARD the possibility to define various users and groups in the database. At the beginning three groups are predefined : `universe` contains all the users of the database, `administrators` contains the administrators of the database and `lexicologists` contains the users in charge of the control of the data.
The information relative to each user are stored in another element referenced by the `<user-ref>` element.
All the dictionaries of the database are referenced by pointers on XML documents that describe them. The pointers are the `href` attributes of the `<dict-ref>` elements grouped in the `<dictionaries>` element.

3.1.2. Description of a Dictionary

To describe a dictionary, we use the `<dictionary>` element. The modifications information is stored in a document pointed by the `history-ref` attribute.
We indicate meta-information on the resources.
The elements `<category>`, `<type>` and `<links>` describe the dictionary macrostructure.

The <category> element indicates the dictionary type (monolingual, bilingual, multilingual, interlingual). The <type> element indicates if the dictionaries are unidirectional, bidirectional or pivot based.
The <links> element indicates the links between the volumes of the dictionary. For example, if a dictionary is pivot based with 3 languages English, French and Malay, it contains 4 volumes Interlingual, English, French and Malay linked as follows:

```
<links>
  <link from="English"
to="Interlingual"/>
  <link from="French"
to="Interlingual"/>
  <link from="Malay"
to="Interlingual"/>
</links>
```

The dictionary volumes are referenced by their unque name. The <volumes> element gathers all the reference to the volumes files noted with the <volume-ref> element.
The source and target languages are indicated with the 3 letter code DML lang type.
The <content> element describes the content of the dictionary. The <domain> element indicates the domain covered by the dictionary (general, medecine, computer, etc.)
We indicate also the size of the dictionary in bytes by <bytes>, and the headword number by <hw-number>.
For the version management, we indicate the version number (<version>), the creation-date of the dictionary (<creation-date>) and the date of the integration of the dictionary into the database (<installation-date>).
For the non-DML resources, we need to indicate the file format (<format>) and the encoding (<encoding>). The encoding values are determined by the DML type encodingType.
We also indicate meta-information on the dictionary like the resource supplier (<source>), the owner (<owner>), the responsible at the database level (<responsible>), the rights attached to the dictionary (<legal>) and miscellaneous comments (<comments>).
The CDM (see chapter 4) elements list (<cdm-elements>) is stored with for each element, its real name in the resource and the maximal response delay. The (<corpus>) element is special, it allows to indicate that we search a string anywhere in the dictionary.

3.1.3. Description of a Volume

The <volume> elements gathers dictionary entries with the same source language. The modificaitons history is referenced with the history-ref attribute.

3.2. Microstructure Definitions

To represent dictionary microstructures, we propose to redefine in XML the structures defined with LINGARD (see serasset (1994).

3.2.1. Trees

To represent a dependance tree associated to the sentence "Le chat mange une souris.", for example, we can use a “decorated node” <dn> with attributes corresponding to the grammatical variables.

```
<dn ul="manger" time="present"
aspect="imperfectif">
  <dn ul="chat" determ="defini"
gnr="masc" pos="-1"/>
  <dn ul="souris" determ="indefini"
gnr="fem" pos="+1"/>
</dn>
```

3.2.2. Links

The definition of a link is done with the xlink standard [XLink 1.0]. We alslo add our attributes:

- The attribute type="bidirectionnal" or type="oriented" indicates if the link is bilingual or not;
- The attribute id is of the DML id type. It allows to attribute a unique id for each link;
- The content text of the element allows to tag the links.

Here is a link example:

```
<link type="oriented" id="l001"
href="example.xml#xpointer(//node[xl
:label='n002'])"/>
```

The reference to the external element is done with the href attribute. The reference is noted as a URI. If the object does not have a unique id (id), the link is described with the [XPointer] standard. Otherwise, it is pointed as follows:

```
<link type="oriented" id="l001"
href="example.xml#n002"/>
```

3.2.3. Graphs and Automatons

The xlink standard [XLink 1.0] is used to describe arcs. The arcs type is oriented type="oriented" or bijective type="bijective". The source and the target of the arc are noted with the node identifiers from="n001" and to="n002".
The definition of an automaton follows the definition of a graph. The starting node is noted with the xl:title="starting-node" attribute. The ending nodes are noted with the xl:title="ending-node" attribute.

3.2.4. Functions

The following example represents the lexical function [lambda]x1 ($CausOper_1$x0x1). The results of its application to the French lexie DÉSESPOIR are the following: pousser, réduire quelqu'un au désespoir,

jeter quelqu'un dans le désespoir, frapper quelqu'un de désespoir. The function is noted in XML as follows:

```
<function name="CausOper1">
  <arguments>
    <first value="desespoir"/>
  </arguments>
  <valgroup>
    <value>pousser</value>
    <value>réduire [qqun au
désespoir]</value>
    <value>jeter [qqun dans le
désespoir]</value>
    <value>frapper [qqun de
désespoir]</value>
  </valgroup>
</function>
```

3.2.5. Feature Structures

If the features are typed, the type is noted with an attribute. If the feature has several values, the element is duplicated.

```
<feature1
type="type1">valeur1</feature1>
<feature1
type="type2">valeur2</feature1>
```

3.2.6. Sets and Disjonction

Sets and disjunctions are defined directly at the XML schema level with the two elements <xsd:choice> and <xsd:sequence>

3.2.7. Basic Types

The basic type of an XML document is the character string. Thanks to XML schemata, we can use many other basic types like boolean, entity, decimal, float,etc.

4. The Common Dictionary Markup Subset

We defined a subset of DML element and attributes that are used to identify which part of the different structures represent the same lexical information. This subset is called Common Dictionary Markup (CDM).

4.1. Definition of the Subset

The DML framework may be used to encode many different dictionary structures. Indeed, two dictionary structures can be radically different. So, in order to handle such heterogeneous structures with the same tools, we need a common formalism. Standards like TEI [Ide95], MARTIF [Melby94], [ISO99]; GENELEX/EAGLES [GENELEX93] and [GENETER] aim to be universal but very few resources implement them.

We made a more pragmatic work with identifying the information in the existing resources as well as their meaning and naming them ina unique way in the DML namespace

This hierarchized subset is called Common Dictionary Markup and comes principally from the detailed examination of the FeM, DEC, OHD, OUPES, NODE, EDict, ELRA-MÉMODATA dictionaries and the 12th chapter of the TEI about dictionaries. It contains the most frequent elements found in these resources like the headword, the pronunciation, the part-of-speech, the examples, the idioms, etc. These elements have always the same semantics. For example, <dml:entry> always refer to a dictionary entry and <dml:headword> to the headword.

For some elements with closed lists of values, we define a list representing the intersection of the values and conversion rules for each resource. An example is the list of parts-of-speech for each language.

This set is in constant evolution. If the same kind of information is found in several dictionaries then a new element representing this piece of information is added to the CDM set. It allows tools to have access to common information in heterogeneous dictionaries by way of pointers into the structures of the dictionaries. The table 1 lists a first version of the CDM subset.

<CDM tag>	**(TEI equivalent)**
<entry>	(entry)
<headword hn="">	(hom)(orth)
<headword-var>	(oVar)
<pronunciation>	(pron)
<etymology>	(etym)
<syntactic-cat>	(sense level="1")
<pos>	(pos)(subc)
<lexie>	(sense level="2")
<indicator>	(usg)
<label>	(lbl)
<definition>	(def)
<example>	(eg)
<translation>	(trans)(tr)
<collocate>	(colloc)
<link href="">	(xr)
<note>	(note)

Table 1: CDM Elements Subset

4.2. CDM Correspondance Examples

When a resource is recuperated, a correspondance table is established between the original element names and CDM elements. The table 2 has been used for the FeM, OHD and NODE dictionaries.

CDM	FeM	OHD	NODE
<entry>	<fem-entry>	<se>	<se>
<headword>	<entry>	<hw>	<hw>

<pronunciation>	<french_pron>	<pr><ph>	<pr><ph>
<etymology>			<etym>
<syntactic-sense>		<sense n=1>	<s1>
<pos>	<french_cat>	<pos>	<ps>
<lexie>		<sense n=2>	<s2>
<indicator>	<gloss>	<id>	
<label>	<label>		<la>
<example>	<french_ sentence>	<ex>	<ex>
<definition>			<df>
<translation>	<english_equ> <malay_equ>		<tr>
<collocate>		<co>	
<link>	<cross_ref _entry>	<xr>	<xg> <vg>
<note>		<ann>	

Table 2: Equivalents of the CDM elements in the FeM, OHD and NODE

Conclusion

This framework has been extensively used for the Papillon project (see Serasset & Mangeot-Lerebours (2001)) of mutualized construction and consultation of a pivot multilingual lexical database. This experiments allowed us to correct and adapt some parts of the DML.
Nevertheless, the framework need to be opened to the public in order to receive feedback and comments. We plan to open a web site dedicated to the DML soon.

References

GENELEX (1993) Projet Eureka Genelex, modèle sémantique. Rapport Technique, Projet Eureka, Genelex, mars 1994, 185 p.

Nancy Ide & Jean Veronis (1995) Text Encoding Initiative, background and context. Kluwer Academic Publishers, 242 p.

ISO (1998) ISO 639-1 & 2 Code for the representation of names of languages Part 1 & 2 Alpha-3 code. Geneva, Part 1: 17 p., Part 2: 90 p.

ISO (1999) ISO DIS 12200 (MARTIF) Computer applications in terminology - Machine-readable terminology interchange format - Negotiated interchange.ISO TC 37/SC 3/WG I, Geneva, 118 p.

Mathieu Mangeot-Lerebours (2001) *Environnements centralisés et distribués pour lexicographes et lexicologues en contexte multilingue.* Thèse de nouveau doctorat, Spécialité Informatique, Université Joseph Fourier Grenoble I, 27 September 2001, 280 p.

Allan Melby et al. (1996) The Machine Readable Terminology Interchange Format (MARTIF), Putting Complexity in Perspective. Termnet News, vol.54/55, pp. 11-21.

Gilles Sérasset (1994) *Interlingual Lexical Organisation for Multilingual Lexical Databases in NADIA*. In Proc. COLING-94, Kyoto, 5-9 August 1994, M. Nagao ed. vol. 1/2 : pp. 278-282.

Gilles Serasset & Mathieu Mangeot-Lerebours (2001) *Papillon Lexical Database Project: Monolingual Dictionaries & Interlingual Links.* Proc. NLPRS'2001 The 6th Natural Language Processing Pacific Rim Symposium, Hitotsubashi Memorial Hall, National Center of Sciences, Tokyo, Japan, 27-30 November 2001, vol 1/1, pp. 119-125.

Bookmarks

GENETER modèle GENErique pour la TERminologie.

http://www.uhb.fr/Langues/Craie/balneo/demo_geneter.pl?langue=1

XLink 1.0 W3C Recommendation.

http://www.w3.org/TR/NOTE-xlink-req/

XML 1.0 eXtended Markup Language 1.0. W3C Recommendation.

http://www.w3.org/TR/REC-xml

XML Namespaces XML Namespaces. W3C Recommendation.

http://www.w3.org/TR/REC-xml-names

XML Schemas XML Schemas. W3C Recommendation.

http://www.w3.org/TR/xmlschema-0

XPath XPath Language. W3C Recommendation.

http://www.w3.org/TR/xpath

XPointer XML Pointer Language W3C Recommendation.

http://www.w3.org/TR/xpt

Annexs

Annex 1: XML Document Describing a Database

```
<database xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://clips.imag.fr/geta/services/dml/dml.xsd"
name="GETA Lexical Database"
creation-date="22/10/99"
 owner="GETA">
  <partner-servers>
    <user-ref name="XRCE Analyser" href="xrce.xml"/>
  </partner-servers>
  <users>
    <user-ref name="Mathieu.Mangeot" href="mangeot.xml"/>
    <user-ref name="Mutsuko.Tomokiyo" href="tomokiyo.xml"/>
 </users>
  <groups>
    <group name="universe">
      <user-ref name="Mathieu.Mangeot"/>
      <user-ref name="Mutsuko.Tomokiyo"/>
    </group>
   <group name="lexicologists"><user-ref name="Mutsuko.Tomokiyo"/></group>
    <group name="administrators"><user-ref name="Mathieu.Mangeot"/></group>
  </groups>
  <dictionaries>
    <dict-ref name="FeM" href="FeM.xml"/>
    <dict-ref name="Papillon" href="papillon.xml"/>
  </dictionaries>
</database>
```

Annex 2: XML Document Describing a Dictionary

```
<dictionary
xsi:schemaLocation="http://clips.imag.fr/geta/services/dml
http://clips.imag.fr/geta/services/dml/dml.xsd"
category="multilingual"
creation-date="21/1/97 00:00:00"
encoding="ISO-8859-1"
format="rtf"
hw-number="192460"
installation-date="23/06/99 15:04:00"
fullname="dictionnaire français-anglais-malais"
name="FeM"
owner="GETA"
type="unidirectional"
version="1">
  <languages>
    <source-language lang="fra"/>
    <target-language lang="eng"/>
    <target-language lang="msa"/>
  </languages>
  <contents>general vocabulary in 3 languages</contents>
  <domain>general</domain>
  <bytes>9106261</bytes>
  <source>ML, YG, PL, Puteri, Kiki, CB, MA, Kim</source>
  <legal>all rights belong to ass. Champollion</legal>
  <cdm-elements>
    <headword delay="1s"/>
    <pronunciation delay="5s"/>
```

```
    <part-of-speech delay="5s"/>
    <translation lang="eng" delay="5s"/>
    <translation lang="msa" delay="5s"/>
    <corpus delay="10s"/>
  </cdm-elements>
  <administrators><user-ref name="Kim, ML"/></administrators>
  <volumes><volume-ref name="FeM" href="fem_fr_en_ms.xml"/></volumes>
</dictionary>
```

Annex 3: XML Document Describing a Volume

```
<volume
xsi:schemaLocation="http://clips.imag.fr/geta/services/dml
http://clips.imag.fr/geta/services/dml/dml.xsd"
 name="FeM_fr_en_ms"
source-language="fra">
  <entry>…</entry>
    ...
</volume>
```

Annex 4: XML Document Describing a User

```
<user
xsi:schemaLocation="http://www-clips.imag.fr/geta/services/dml
http://www-clips.imag.fr/geta/services/dml/dml.xsd"
name="Mathieu MANGEOT"
creation-date="22/10/2001">
  <login>Mathieu.Mangeot</login>
  <password>toto</password>
  <email>Mathieu.Mangeot@imag.fr</email>
  <profiles>
   <competences>
   <eng level="good">translation</eng>
      <fra level="mother tongue">phonetic, collocations, examples, grammar</fra>
      <jpn level="beginner"/>
      <spa level="good">translation</spa>
    </competences>
  <interests><interest lang="hun,jpn"/></interests>
  <activities>
   <activity dictionary="FeM">interface</activity>
   <activity dictionary="Papillon">administration</activity>
  </activities>
 </profiles>
 <credits>10</credits>
  <annotations href="mangeot-ann.xml"/>
 <contributions>
  <contribution source="French.xml" href="mangeot-cnt1.xsl"/>
  </contributions>
  <requests href="mangeot-req.xml"/>
  <xml-stylesheet type="text/css" href ="mangeot-sty.css"/>
  <groups>
    <group-ref name="universe"/>
    <group-ref name="administrators"/>
  </groups>
</user>
```

Annex 5: XML Document Describing a supplier API

```
<api type="supplier" category="consultation" name="JMDict_en-ja">
  <info>Dictionnaire japonais-anglais de Jim Breen</info>
  <url href="http://www.csse.monash.edu.au/cgi-bin/cgiwrap/jwb/wwwjdic"/>
```

```
  <protocol type="get"/>
 <delay min="1s" average="1s" max="2s" timeout="10s"/>
 <encoding input="UTF-8" output="EUC-JP"/>
  <format input="txt" output="html"/>
  <arguments>
    <element name="source-language">
      <complexType>
        <restriction base="string">
          <enumeration value="jpn"/>
          <enumeration value="eng"/>
        </restriction>
      </complexType>
    </element>
    <element name="headword" type="string"/>
    <element name="regex" type="boolean"/>
  </arguments>
  <result><element name="output" type="string"/></result>
</api>
```

Annex 6: XML Document Describing a client API

```
<api type="client" category="consultation" name="getabase">
  <info>API de consultation de la base lexicale du GETA</info>
  <url href="http://www-clips.imag.fr/cgi-bin/geta/dicoweb
mailto:dicoweb@imag.fr
telnet://www-clips.imag.fr:2628"/>
  <protocol type="post get mailto DICT" login="anonymous"/>
 <encoding input="ASCII ISO-8859-1 UTF-8" output="UTF-8"/>
 <format input="txt xml" output="xml html txt"/>
  <arguments>
    <element name="name" type="string"/>
    <element name="source-language" type="lang"/>
    <element name="word-order" type="string"/>
    <element name="cdm-elements" type="string"/>
    <element name="context" type="positiveInteger"/>
    <element name="input" type="string"/>
  </arguments>
  <result>
    <element name="output">
      <complexType>
        <sequence><element name="article" type="articleType"/></sequence>
      </complexType>
    </element>
  </result>
</api>
```

TOWARDS A GENERIC ARCHITECTURE FOR LEXICON MANAGEMENT

Cristina Vertan
Walther von Hahn,
University of Hamburg, Natural Language Systems Department
Vogt-Kölln-Straße 30, 22527 Hamburg, Germany
cri@nats.informatik.uni-hamburg.de, vhahn@nats.informatik.uni-hamburg.de

Abstract

In this paper we propose an architecture for a lexicon management tool MANAGELEX. This tool aims at a general environment for reading, updating and combining lexicons in different formats. The starting point is the already existing lexicon models MULTILEX and GENELEX. Each functionality (reading, updating and combining) is based on a corresponding model, which can be configured and maintained coherently.

1. INTRODUCTION

A large amount of lexical resources was developed during the last 15 years. Unfortunately, in the absence of a standard each application produced and used its own lexicon in a specific format and a specific model, according to particularities of language, system functionality and available physical resources. Reusable lexical resources, however, could noticeably reduce the cost of development of NLP applications. Moreover, during research projects, lexicon requirements may change over the run time of the project, and maintaining a suitable lexicon is expensive and time-intensive work.

The problem of standardization appeared as an absolutely and urgent necessity, and several projects were carried out in this sense (v.Hahn 2000). The task is quite difficult because it implies at least two components : standardization of the format and standardization of the model. Moreover, these two components are not completely independent. For the former it is general agreed today, that the starting point is a SGML –based format. Several SGML-lexicon standard formats were already proposed (EAGLES, OLIF, SALT) (Lieske & al. 2001, Melby 1999). It is, however, necessary that we have not only a standard set of tags but also a standard model of a lexicon representation. As a result of this insights, several projects tried to develop a standard and general model for lexicons. The most well-known formalisms after this phase are GeneLex and Multilex.

2. STANDARD LEXICON MODELS. STATE OF ART

Although having many architectural features in common, Genelex is abstracted basically from a French monolingual lexical model, whereas the Multilex architecture is genuinely designed as a multilingual language-independent general structure, trying to include all language specific models (EAGLES 1996). At least, as quoted in one of the final reports (Praprotté & al. 1993), Multilex *"is based on a consideration of the following languages: English, German, French, Spanish and Italian, and to lesser degrees Dutch and Greek"*. Compared to the multitude of (at least) European languages we observe that the Slavonic family was not taken into consideration, and also a lot of other languages which bring in new linguistic features (for example Romanian, although it belongs to the Latin languages, it has several important characteristics, due to the Slavonic influence).

The MULTILEX architecture presented a generic model for a lexical entry, which can be used as a starting point for further developments. However MULTILEX, as other similar projects *"imposes constraints on the linguistic level. Each of these projects imposes its own notion of 'lexical unit' (lemma, word-sense, concept) and its own logical structure (Typed Feature Structures, Entity-relationship model, automata, trees,...)"* (Sérasset 1996).

With these constraints, a user at the moment cannot use the same system to manipulate two lexicons coming from different places. Some steps in this direction were done in MULTILEX, which originally proposed the development of tools to convert lexicons into MULTILEX format. The proposal was not further developed because, quoting the same final report (Praprotté & al. 1993) *"copyright problems, problems in converting and correcting dictionary data, a lack of consistency in the data"* made this proposal unreachable.

Much lexical work from completed projects cannot be used in follow-up projects because of one of the following reasons:

- The lexicons were produced with the help of systems that are not any longer maintained; thus nobody can provide an export facility.
- In some cases, lexicon definitions contain procedural elements, which cannot be used without the hosting system,
- Lexicons may contain too rich features, which are too expensive to remove from the files.
- Experimental lexicons may be inconsistent or contain entries with different granularity,
- Lexicons may be stored in a data base, whereas others are plain files and the export formats do not match,
- Lexicons differ in their linguistic classes, i.e., there is a more-to-more mapping between feature classes.

From another point of view the use of a specific format (for example MULTILEX) means to adapt a posteriori other systems' processes to read and work which such external formats. This is usually quite cost-expensive.

The situation is much more critical for small languages, and languages from Central and East Europe, for which

lexical resources were developed quite ad hoc as they were needed for a certain project.

Although a lot of resources after a few years may be linguistically and technically outdated, about 60% of a dictionary with approx. 80 000 entries comprises the lexical core of very high and rather high frequency words, which remain stable in their syntactic and semantic properties over a long period of time. The other part (especially terminology) from time to time must undergo revision, updating or even replacements.

3. MANAGELEX A GENERIC LEXICON MANAGEMENT MODEL

Following the above considerations, we assume that for a rather long time from now, NLP applications will still have to deal with manipulations of non-standard lexical resources.

However, this is only possible with rather general lexical management tools for acquisition, comparison, manipulation and validation of lexicons, based on several abstract models.

In this section we propose a new architecture for a lexicon management tool (MANAGELEX), a tool, which is able to read, convert and combine lexicons, independent of their format, language or system requirements.
The general architecture of such a system includes (as shown in figure 1) 3 levels of abstraction (which follow the ANSI(1999) data modeling specifications): the meta model level, the model level and the real world level.

- The real world level identifies real (present), distinct objects, their concrete features, and the actual relation among them. In figure 1 this corresponds to the encoded lexicons (DocA, DocB) and their structure (StructA, StructB)

- The model level groups real world objects and present features into object and attribute classes and recognizes possible relationships among object classes. On this level our architecture has 3 tools:
  - A tool for reading and updating a lexicon (acquisition and editing tool),
  - a tool for encoding and decoding (encoding / decoding tool) and
  - a tool for mapping two lexicons, possibly with different structure (mapping tool)

- The meta model level, classifies types of elements appearing on the model level and the abstract relations among them, situation independent. Accordingly, we propose

  - A generic lexicon model (LexMod) which provides a rather rich model of possible lexical information. Here, every linguistic feature, with their possible values which may occur in a set of languages (at least European) are specified (MULTILEX together with the MILE (Calzolari & al. 2001) model (defined in the frame of the ISLE project) are a good starting point). A flexible formal specification will be provided for this model. The model will also allow for new categories, joining as well as splitting of existing categories.

  - A generic encoding model (Encod), which specifies the way of combining the linguistic information in a specific entry and lexicon structure. The model should also include options for encoding files in the new generally agreed SGML-standards as OLIF or SALT (Lieske & al. 2001; Melby 1999).

  - A mapping model (MAP), that specifies modalities of combining two lexicons and takes into account problems like mutual gaps and complex categories.

Given this architecture, we now explain the functionality of the envisaged system in three situations:

1. Building / updating a lexicon.

Input: Lexicon definition from LexMod, Encoding Model Encod,
Output: Lexicon interface, lexicon file

The operation is mainly performed by the acquisition/editing tool. The interface of this tool is built automatically according to the characteristics selected from LexMod for this particular lexicon. The output of this tool is a data structure recording the structure of the lexicon LexA. The encoding / Decoding Tool uses this data structure and the Encoding module and produces and encoded lexicon DocA.

2. Reading a lexicon.
Input: Lexicon file, Encoding Model Encod,
Output: -

This operation requires first the identification of the encoding and the generation of the corresponding linguistic structure (StructB). Responsible for all these is the encoding tool

3. Join of two lexicons (LexA and LexB)
Input: General Lexicon definitions from LexMod, lexicon definitions from StructA and StructB, mapping models MAP
Mapping models MAP
Output: Lexicon file

This is the most challenging operation. The mapping tool has to use not only the structure of the two lexicons (StructA and StructB) and the mapping model (MAP) but also the generic lexicon model (LexMod). This is required for example in case of different names for the same linguistic feature. The resulting structure contains data consistent with both lexicons. Furthermore a new lexicon can be encoded as described above.

4. CONCLUSIONS

In this paper we described a model of a possible lexicon management tool, which can deal with frequent problems in lexicon acquisition / maintenance. The presented architecture is still in prototyping phase. We envisage to develop it in the frame of an European project. How ever for the moment we will take into account the European languages. Extensions to other language should be possible one the system reaches a stable version. The system is not intended to replace the actual already defined standards, but to supply the use and reuse of the already developed non-standard lexical resources

REFERENCES

ANSI-American National Standard Institute(1999), Standard X3. 138-1988, *Information Resource Dictionary System (IRDS)*

Calzolari, N. and A. Lenci and A. Zampolli and N. Bel and M. Villegas M. and G. Thurmair G., (2001) "The ISLE in the Ocean – standards for Multilingual Lexicons (with an Eye to Machine Translation*)", Proceedings of MT Summit VIII, Santiago de Compostella, 2001*

EAGLES (1996) "Input to the EAGLES architecture work: survey of MULTILEX", *http://www.ilc.pi.cnr.it/EAGLES96/lexarch/node4.html*

v.Hahn, W (2000), "Standards in Natural Language Processing – New Steps in Language Engineering", in *Standards in Information technology S. Nedevschi and K. Pusztai (Eds.*), Casa Cartii de Stiinta, Cluj.

v.Hahn, W. (1999), "Metamodelling of Lexical Acquisition Tools", *Proceedings of EUROLAN '99*, Iasi.

Lieske Ch. and S. McCormick and G. Thurmair (2001), "The Open Lexicon Interchange Format (OLIF) comes of Age", *Proceedings of MT Summit VIII, Santiago de Compostella*

Melby, A. K. (1999), "SALT: Standards-based Access service to multilingual Lexicons and Terminologies", *http:// www.ttt.org*

Paprotté, W. and F. Schumacher(1993), "MULTILEX – final Report WP 9: MLEXd", *Report MWP 8 – MS*

Sérasset G.(1996), "Recent Trends of Electronic Dictionary. Research and Development in Europe", *Report GETA-IMAG, CNRS, Grenoble,*

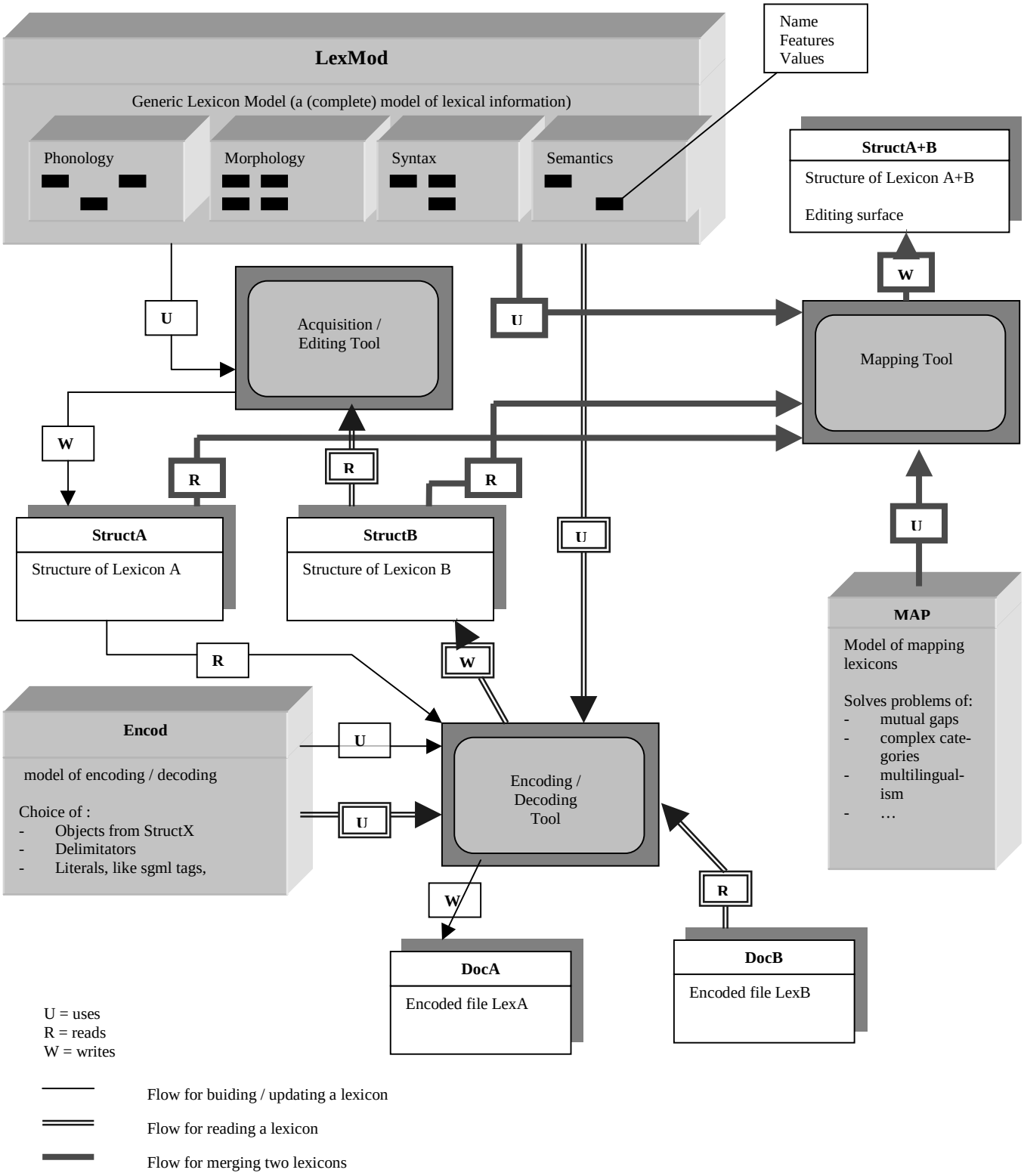


Figure 1: MANAGELEX: components and Workflow

Management of Language Resources using Metadata

P. Wittenburg, Daan Broeder

Max-Planck-Institute for Psycholinguistics
peter.wittenburg@mpi.nl

Abstract

Technology development allows many more researchers than before to create language resources especially with multimedia extensions. This creates a resource management problem that exceeds the boundaries of established resource centers. Metadata environments such as the one proposed by IMDI that offer a metadata set and also tools to operate on them have a strong potential to help the individual researcher to carry out his resource management tasks. In addition, it allows him to easily integrate his resources into a large distributed domain of resources. The work at the Max-Planck-Institute for Psycholinguistics to establish a large multimedia language corpus helped to understand the needs and requirements. Due to this experience the IMDI environment has reached a state of maturity, but still some important features have to be added.

1. Introduction

Researchers and developers in the area of language resources are faced with four very dominant trends in the recent years: (1) The number and complexity of language resources stored in digital archives is growing fast, (2) there is an increasing acceptance of the need to improve the availability of the resources, (3) the Internet now connects many archives storing such resources and this asks for interoperability and (4) for many language resources need to be stored in archives for a large period of time due to economical and ethical reasons.

An impression about this explosion of resources can be given by the example of the multimedia/multimodal corpus at the Max-Planck-Institute for Psycholinguistics where every year around 40 researchers carry out field trips, do extensive recording of communicative acts and later annotate the digitized audio and video material on many interrelated tiers. The institute now has already more than 7000 annotated sessions - the basic linguistic unit of analysis - and we foresee a continuous increase. It was usual that researchers managing their resources with individually designed Excel-Sheets eventually were not able to keep control of them and that the institute effectively lost all access to resources when a researcher left. Thus the individual researcher as well as the institute was both faced with a resource management problem. It is known that in other research centers, universities and also in industry similar situations occur.

The increase of the amount of resources was paralleled by an increase in the variety and complexity of formats and description methods. Moving from purely textual to multimedia resources with multimodal annotations caused this. Media can include not only several audio and video tracks, but also increasingly often other information such as for example from eye trackers, data gloves and brain image recorders.

In many areas resources were seen as the private capital of a researcher or a specific project that served only to investigate a limited number of research questions. Therefore, the need to make resources available for other research was not seen. However, researchers now understand the potential of modern technology to immediately access the raw material, which enables for example re-coding, or incremental annotation procedures that can be part of collaborations. These opportunities increase the individual researchers willingness to share his resources and to invest time to create publicly available descriptions. We clearly recognize a trend towards making the resources themselves available via the Internet or at least indicating what resources exist by creating structured descriptions available on the Internet.

The usage of the Internet demands for interoperability on various levels. Therefore new technologies devoted to the special requirements of the Internet such as RDF (Resource Description Framework), XML and UNICODE are have been developed to improve the exchange and re-usage of data. The usage of open standards is even more important when repositories of language resources have to support long archive periods. The Internet also adds another dimension of complexity since people want to create distributed repositories where the resources of a corpus can be scattered over different locations, nevertheless requiring transparent access to them.

Summarizing we can say that a much broader group of researchers besides the experts who have always handled expensive resources are now involved. They are managing larger amounts of more complex structured resources, making them available in standardized formats and descriptions via the Internet. Now that resource creation has become much more easy many individual researchers are also coping with resource management problems pushing the management task beyond the experts at large data centers.

2. Resource Management

The increased relevance of resource management can best be seen in the document domain by the emergence of various sorts of commercial Content Management Systems. It is widely understood that only improved management concepts will allow us to prevent a chaotic situation where we will have an increasing amount of data on our storage devices, but don't know about them nor know how to access them.

We can identify at least four different groups of people involved in resource management each one with their own views: (1) the computer system specialists have to be able to manage data on a physical level. They allocate physical resources, define structures in file systems and take care of redundant copies for secure data storage. (2) The producer of resources wants to integrate his resources into the repository in an easy way and describe them easy and correctly to facilitate retrieval. (3) The user wants to deal with data on a domain-oriented level, i.e. a level where the

well-established concepts and terminology of a domain are used. He is not interested in file system details. This view includes distributed scenarios where the user wants to combine resources from different institutions without having to know where exactly the resources reside. Often the producer is himself a user. (4) The archive manager acts as an interface between system specialists and producers and also prefers to manage data at the level of domain concepts. At least he has to know how the system managers handle the resources since he has to draw the links between logical and physical structure and influence for example the policies for protecting the data. In many cases the producer/user is also the archive manager, since there is no support stuff. Management has to consider all views.

The following is a non exhaustive list of points to be addressed by modern resource management (resource discovery is in general seen as being a component of resource management, but in this paper we will mention it, but not focus on it).

- How to store resources such that they can survive for many years independent from technology changes.
- How to protect resources against unauthorised access
- How to create personalized views on resource repositories to facilitate easy and optimised navigation
- How to offer easy and immediate access to resources after access is approved?
- How can descriptions of sets of resources be modified easily?
- How to easily integrate new resources into the distributed resource repository?
- How to keep track of old versions?
- How to make such a management scheme available to interested parties.
- How to easily move groups of resources to other locations transparent to the user/producer?
- How to achieve hardware and operating system independent operation within the resource domain?
- How to easily integrate different data types that belong together and allow access while hiding the complexity?
- How to inform people about the existence of a resource and its major characteristics?
- How to easily discover resources in a distributed scenario from a conceptual perspective?

In this paper we will focus on the resource manager and user views. This although many important problems such as for example the problems of long-term archiving of digital media are not at all solved.

3. Pillars of Management

As already indicated, industry delivers a wide range of software solutions that are meant to cover documents of all sorts. In this paper we will not discuss Document Management Systems although they may deliver much functionality, but focus on the key pillars of open distributed solutions aimed at our specific environment and data types.

3.1. Standards

Open standards are very important to achieve interoperability, to build up long-term archives and to produce long-term available tools. Especially in the domain of computer-based language resources, however, we are faced with an extremely dynamical situation. This means we are confronted with a multitude of standards making many people turn over to use the word “best practice guidelines” instead. For multimedia resources for example we are confronted with a long list of media compression methods (MPEG1/2/4, Cinepak, Sorensen, MP3, ATRAC etc) all emerging within the last decade. Each having its advantages and disadvantages dependent on the field of application. For an archive one has to decide about major backend standards (such as MPEG2) which allows creating other representations for specific applications on the fly.

Referring to the earlier questions we need a couple of standards. We claim that many of the management problems can be solved with the help of establishing a suitable metadata environment existing of a metadata element set and appropriate tools. Tools themselves are not subject of standardization per se, since it is good to have competing solutions. With respect to the metadata set, however, we need agreements on various levels. The metadata elements are the dimensions of how to characterize a resource and it is clear that each choice for a set of dimensions limit the expressiveness for other groups of users. Therefore, we can expect that there will be different sets of dimension to describe multimedia/multimodal language resources. Important for the community is that we have open accessible definitions of the elements such that schemes can refer to them. They should be described as Data Categories if this will be the common practice for terminology repositories.

In addition, in the case of non-orthogonal spaces as the one we need to describe, these dimensions can only be defined appropriately by specifying suitable controlled vocabularies. They are the values that a specific dimension can take. Also these controlled vocabularies have to be openly accessible and should be defined in the same way. Both elements and their controlled vocabularies, have to be known exactly to achieve interoperability. Of course, it makes sense to use just one controlled vocabulary for example for language codes, but also here we are faced with different (quasi) standards such as ISO 639-2, the Ethnologue list from SIL[1] [1,2] and the various lists handled by specific projects. Also here we must accept that different vocabularies will exist.

Consequently, we are faced with mapping problems on different levels. RDF will be the primary language to try and bring all the different pieces of the mosaic together. This problem has not been tackled yet with the exception of a few cases such as in the Harmony project and in the mapping proposal from IMDI[2] to DC[3]/OLAC[4]. MPEG7[5] categories were mapped on Dublin Core categories in a very restricted way and the element relations are described

[1] Summer Institute of Linguistics
[2] ISLE Metadata Initiative
[3] Dublin Core Metadata Initiative
[4] Open Language Archives Community
[5] MPEG7 is the standard for media annotation within the family of MPEG standards in the film and media industry

with the help of the RDF formalism. Such a formal framework has not yet described the IMDI to OLAC mapping. At the moment we don't know which expressional power the community will need to accomplish the big task to create such a mapping for the language resource domain. The emergence of DAML/OIL [3] indicates, however, that RDF itself will probably not be sufficient.

It is assumed here without further comment that XML is our common language, i.e. all definitions and frameworks to be used should be based on XML.

3.2. Metadata Descriptions

The usage of metadata descriptions for improving the management of documents is not a new concept. Librarians are used to describe their documents with cards since many years. Linguists and speech engineers were used to describe characteristics of their resources and put these in file headers - mostly project specific formats. The community learned a lot from the TEI [6] work about standards for resource headers (later adopted by the CES[7]) and it is still used as a reference to look at. Also in some projects such as CGN[8] the TEI recommendations were followed to a certain extent.

TEI is a comparatively exhaustive descriptor set meant to describe the characteristics and structure of a resource. Newly developed metadata sets do not want to describe the resource in a too great detail, but address the problem of easy discovery primarily, i.e. a resource would be described sufficiently well, if a user manages to find it. Metadata sets such as DC, OLAC and IMDI follow this approach. DC tries to address the discovery problem with 15 sloppily defined categories ordered in a flat structure. In doing so DC allows the user to describe resources about steam engines as well as resources about Sign Language both on a very general level. For many DC categories it is not clear how they can be applied to different domains, therefore refinements are defined as was done by the OLAC initiative. The "DC:Type" element that defines the resource type is refined by the characteristic "CPU" to describe the type of CPU a NLP tool can run on. The semantics of such an element are stretched extremely.

MPEG7 and IMDI followed another approach since they started with studying the domain specific requirements. For MPEG7 it is essentially the production process of movies that has to be covered to later be able to retrieve relevant segments that are covered by the metadata set in addition to the ordinary elements such as "Creator". The basis of IMDI was an extensive survey of the different ways in which linguistic resources in all their variety have been described. Often this was done in the form of a proprietary "file-header" that contained metadata information about the annotation as a whole such as for instance the CHAT file format [4]. CES (being TEI compliant with respect to corpora) suggestions were applied were useful for discovery, however, we have not found sufficient support for other types of linguistic data than text. TEI/CES also mixes metadata and content in the same way as MPEG7. IMDI has favored a physical separation of metadata and content allowing uncomplicated protection schemes which is important for some groups of users. It also allows separate management of resources and metadata, usefull because the integration of legacy data formats has to be supported.

[6] Text Encoding Initiative
[7] Corpus Encoding Standard
[8] Spoken Dutch Corpus Project

3.3. IMDI

3.3.1. Session Concept

The IMDI set was especially targeted at multimodal/multimedia resources and their inherent complexity, i.e. basis is in general the existence of media recordings. This led to the development of the "Session" concept. For linguists a session is defined as the basic unit of linguistic analysis and covers a coherent type of linguistic action or performance. From a corpus organization point a session is the leave in the tree. A session is in general associated with a bundle of tightly related resources: a video recording of a native speaker, a set of pictures of that persons house, some field notes about this scene and afterwards some multimodal annotations. The IMDI definition of the term "session" covers this bundling from an access and management point of view.

In DC one would have to use the "DC:Related" element to describe the relation between these resources that is associated with much overhead. This was described in more detail in the IMDI-OLAC mapping document [5].

From a management point of view the session concept makes sense since accessing or extracting subcorpora implies accessing resp. copying of complete sets of related information.

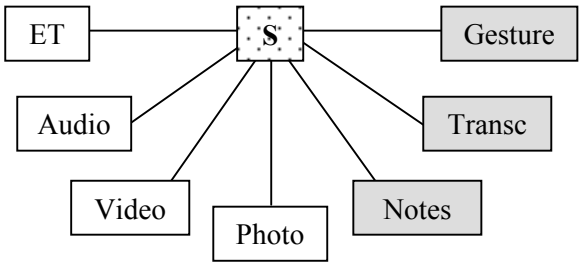


Figure 1 shows a typical session with its related resources all referring to the same linguistic event. It covers different types of recordings and different annotations.

In IMDI its the structured metadata set which describes this relation, i.e. there is only one metadata description (if the user decides to do it that way) with different sub-blocks describing the characteristics of the individual components. This way allows a user to ask questions such as "give me all resources which have eye movement recordings and a phonetic transcription of what was spoken"

3.3.2. Browsable Domain

Next to the "Session" concept, IMDI introduced the idea of structuring corpora in a conceptual space by having hierarchies of (sub-) corpora where description nodes representing a certain level of abstraction with respect to other (sub-) corpus nodes culminating eventually in pointers to session nodes (see figure 2). Each level represents a certain abstraction layer that is meaningful to the resource manager or user.

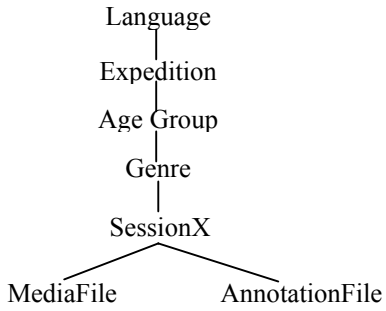


Figure 2 shows a typical hierarchy from field linguistics

Since corpus nodes create logical structures several parallel hierarchies can be created to structure the same (sub-)corpus and to express different interests of users. This allows each user to establish his own preferred view on the distributed resource domain and by also using bookmarks to create his own conceptual space (see figure 3). These parallel hierarchies can also be used to support versioning. Of course, there is no reason for the user to not create cross-references. For management purposes such cross-references are of course difficult to handle, i.e. the resource managers preferably would work with just the canonical tree.

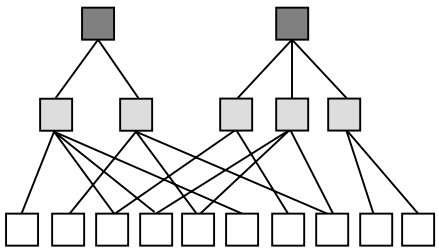


Figure 3 shows two user defined hierarchies referring to the same set of session nodes that are at the bottom level. One view could make a sex distinction, another one by age groups.

The mechanism by which the (sub-) corpora nodes refer to each other is to use URL's. This has the advantage to support distributed corpora frameworks and create a unique namespace for all resources.

3.3.3. Data Type Integration

Such a browsable domain as indicated is of course very useful for integrating various data types that we find in complete corpora. We already described the integration on the session level. For many data types however it only makes sense to associate them with higher nodes in a corpus tree. Such a node represents an abstraction with respect to a number of metadata elements (for example sharing the same language). Lexica can be related to a sub-corpus associated with a language or a set of recordings for a language (lexicon of a 3 year old child). Field notes and comments about dialect variants in general can appear on all levels of a corpus. In general many of these data types do not have any definite structure, but are just prose texts in some general format such as DOC, HTML or PDF. Corpus management has to provide mechanisms to include such descriptions in a flexible way. IMDI allows the resource manager to do so, but of course, will exclude proprietary formats such as DOC.

3.3.4. Practical Considerations

A strong concern was and still is how one can enforce creators and managers to adhere to standards with all its consequences as described above. The stricter the rules are such as full adherence to the chosen controlled vocabulary of a certain element, the more sensitive these procedures will become. Although the IMDI type of operations are now in operation for 3 years we cannot claim that a "standard" such as IMDI for describing language resources will not undergo changes. In IMDI for example we expect changes with respect to the dimensions and vocabularies that describe the resource content.

It was found - and this experience is nothing new - that it is very important to support the creators and managers with professional tools. Within IMDI it was always tried to have a balance between the development of the metadata set and an editor that supports the creation of IMDI descriptions. The IMDI editor now supports

- All metadata elements including their controlled vocabularies in a dynamic way, i.e. if the definition in the repositories change the editor will adapt its representations
- Sub-blocks which allow the user to save and reuse reoccurring information such as participant or project information

Version changes in the metadata set can of course lead to severe problems for corpus management and metadata usage. There efficient tools are of the greatest importance to modify all whole sets of existing metadata descriptions. Currently, a script allows the resource managers to change the values of the elements for a whole set of metadata descriptions. Of course, such operations are very sensitive and such a script may not be given to the general user. The intention is to include such an option in the editor such that all changes are conforming to the actual IMDI definitions.

The browser offers the same feature as the editor in so far that it also uses the actual vocabulary definitions from the repository. Further, the browser offers the following management relevant features:

- A user can create new (private) nodes and therefore define his own view on a sub-corpus
- It is possible to start the editor from the browser environment to modify metadata descriptions
- It is possible for the users (managers) to associate tools with individual or bundles of resources such that when a (set of) useful resources was found immediately a tool can be started to operate on the resources.

Both tools will have to provide for version conversion in case they find metadata descriptions in an older format. They should not however work with old versions without forcing (if possible) an update.

In the future the editor has to be extended to be able to create formatted lists (Spreadsheet type) of the content of a range of metadata descriptions for easy check and input to for example statistic programs. This is a favorite view on metadata of many users. The user has to be able to select the elements he wants to see. One complication is

given through the fact that some elements can occur several times such as participants, i.e. the number of entries for the spreadsheet can only be computed by first reading all selected metadata descriptions.

3.3.5. Difference to Normal HTML Domains

Of course, the basic organization principles sound very familiar, since we use the same for designing web pages. Instead of creating XML based descriptions one could create HTML pages and include all information and data types as hyperlinks in the usual way. Some archives are operating this way. Metadata descriptions could be included in the headers of the HTML files to support element-based search.

The IMDI team did not choose for this way for the following major reasons:

- HTML is basically a way to describe how documents should be displayed and not to describe data structures.
- Using HTML would not have made sense without also using HTTPD servers and browsers. Otherwise HTML is just a much less powerful version of XML. The current HTML browsers however are not suited to perform all computation tasks required of a metadata browser such as making intelligent choices for tools to work on resources.
- We needed a format to transfer information. Tools should be able to interpret this information either to display parts of it or to offer the user a choice of tools to work on referenced resources.

4. Conclusions

Based on 3 years of experience with a multimedia/multimodal corpus which covers already more than 7000 metadata descriptions and a showcase application including sample corpora from 6 European institutions we can draw some conclusions.

1. All questions raised in chapter two are addressed by the IMDI environment with two exceptions: (1) Version handling of resources and metadata description schemes are not yet supported by the tools by the tools. (2) The tool for extracting complete sub-trees of a corpus is not yet available.
2. The need to apply the definitions and tools to such a big and heterogeneous corpus as for example the MPI corpus was a useful and necessary enterprise. It made us understand the underlying processes and requirements to establish an environment such as IMDI.
3. Corpus management was performed during the development phase of the IMDI environment. This meant that frequent updates of the metadata schema took place that required frequent transformation of the metadata files.
4. We now have an environment where it is comparatively easy to integrate or build up IMDI based archives that supports the creator, the user and especially the resource manager with suitable mechanisms and tools.
5. Since all definitions are open everyone can create his own set of tools to work on the metadata descriptions, i.e. improve the search engine or write another browser.
6. Using a file oriented framework for storing metadata only appears as an advantage when distributing or integrating small (personal) archives or making extractions of sub corpora on portable media for off-line use. It does however create confidence of the linguists that they can take their metadata descriptions with them on a floppy and are not dependent on server bound DBMS's.
7. Using metadata in a uniform, controlled and structured way is a new experience for our linguists. It did and still costs a large persuasion effort to have them input their metadata. It has only be since a short time that they themselves can reap the benefits by using for instance metadata search, since a critical mass is necessary and since the improvements for resource management had to become apparent.
8. The introduction of a complete and operational metadata environment was the first experience for the development team of this sort. Often the practical experience guided us in designing and improving the tools, since we did not foresee all aspects of efficient resource management beforehand.

Finally, it seems to be appropriate to add a statement about future perspectives. We see metadata for language resources still in its beginning phase, since there are not so many resource repositories which already created the appropriate files. Especially there are only few attempts to do resource management with the help of metadata environments. We have shown their great potential but also the difficulties involved. Especially the inclusion of metadata element and vocabulary definitions in open repositories and the formulation of their relations with the help of Semantic Web compliant mechanisms such as RDF will motivate more groups to contribute and participate. Interoperability between different metadata sets will also be facilitated by applying these agreed standards.

The soon to be started INTERA project is aiming to realise and work at the above mentioned points.

[1] [ISO639-2]
Codes for the representation of names of languages - part 2: alpha-3 code, International Organization for Standardization (ISO), 1998. http://lcweb.loc.gov/standards/iso639-2/langhome.html

[2] Ethnologue language name index http://www.sil.org/ethnologue/names/

[3] DAML/OIL: http://www.daml.org

[4] Childes: http://childes.psy.cmu.edu

[5] IMDI-OLAC-Mapping: http://www.mpi.nl/ISLE

Towards Multimodal Content Representation

Harry Bunt*, Laurent Romary&

Computational Linguistics and AI, Tilburg University
P.O. Box 90153, 5000 LE Tilburg, The Netherlands
Harry.Bunt@uvt.nl
&LORIA, University de Nancy
B.P. 239, 54506 Vandoeuvre-les-Nancy, France
Laurent.Romary@loria,fr

1. Introduction

Multimodal interfaces, combining the use of speech, graphics, gestures, and facial expressions in input and output, promise to provide new possibilities to deal with information in more effective and efficient ways, supporting for instance:

- the understanding of possibly imprecise, partial or ambiguous multimodal input;
- the generation of coordinated, cohesive, and coherent multimodal presentations;
- the management of multimodal interaction (e.g., task completion, adapting the interface, error prevention) by representing and exploiting models of the user, the domain, the task, the interactive context, and the media (e.g. text, audio, video).

An intelligent multimodal interface requires a number of functionalities concerning media input processing and output rendering, deeper analysis and synthesis drawing at least upon underlying models of media and modalities (language, gesture, facial expression of user or animated agent), fusion and coordination of multimodal input and output at a semantic level, interpretation of multimodal input within the current state of the interaction and the context, and reasoning about and planning of multimodal messages. This implies an architecture with many components and interfaces; a reference architecture of an intelligent multimodal dialogue system was established at the workshop `Coordination and Fusion in Multimodal Interaction' in Dagstuhl, Germany, November 2001 (see Bunt, Kipp, Maybury and Wahlster, forthcoming, and http://www.dfki.de/~wahlster/Dagstuhl_Multi_Modality). The communication between many of the components in a multimodal interactive system rely upon an enabling syntax, semantics and pragmatics. A multimodal meaning representation plays central stage in such a system, supporting both interpretation and generation. Such a representation should support any kind of multimodal input and output, and should, in order to be useful in a field which is still developing, be sufficiently open to support a range of theories and approaches to multimodal communication.

The present document is intended to support the discussion on multimodal content representation, its possible objectives and basic constraints, and how the definition of a generic representation framework for multimodal content representation may be approached. It takes into account the results of the Dagstuhl workshop, in particular those of the informal working group on multimodal meaning representation that was active during the workshop (see http://www.dfki.de/~wahlster/Dagstuhl_Multi_Modality, Working Group 4).

2. Scope

To delineate the task of formulating objectives, constraints and components of multimodal meaning representation, we must first have a shared understanding of what is meant by *meaning* in multimodal interaction. We propose to define the meaning of a multimodal `utterance' as the specification of how the interpretation of the `utterance' by an understanding system should change the system's information state (taken in a broad sense of the term, including domain model, discourse model, user model, task model, and maybe more - see e.g. Bunt, 2000). While formulated with reference to input interpretation only, this definition can also be related to the generation of multimodal outputs, by assuming that an output is generated by the system in order to have an effect on the user through the interpretation of that output by the user. (The generation of appropriate outputs thus depends on the system having an adequate model of what its outputs may mean to the user – which is exactly as it should be.)

A multimodal meaning representation should support the fusion of multimodal inputs and the fission of multimodal outputs at a semantic level, representing the combined and integrated semantic contributions of the different modalities. The interpretation of a multimodal input, such as a spoken utterance combined with a gesture and a certain facial expression, will often have stages of modality-specific processing, resulting in representations of the semantic content of the interactive behavior in each of the separate modalities involved. Other stages of interpretation combine and integrate these representations, and take contextual information into account, such as information from the domain model, the discourse model and the user model. A multimodal meaning representation language should support each of these stages of interpretation, as well as the various stages of multimodal

output generation. Since we are considering inputs and outputs from a semantic point of view, the representation of lower-level modality-specific aspects of interactive behavior, like syntactic linguistic information or morphological properties of gestures is not a primary aim, but some such information may percolate as features associated with a meaning representation, especially at intermediate stages of interpretation, where their relevance for semantic interpretation may not have been fully exploited. At the other end of interpretation, where understanding is rooted in information structures like domain models and ontologies, a multimodal meaning representation language should support the connection with frameworks for defining ontologies and specifying domain models, such as DAML + OIL.

While supporting the linking up of meaning epresentations with ontologies and `low-level' modality-specific information, the design of multimodal meaning representations is to be clearly distinguished from the design of domain model representations, linguistic morphosyntactic representations, representations of facial expressions, etc., which do not fall within this scope. Also, meaning representations should not represent the underlying processes by which they are constructed and manipulated, although it may be important that they are `annotated' with administrative information relating to their processing, such as time stamps.

3. Objectives

The main objective of defining multimodal meaning representations is to provide a fundamental interface format to represent a system's understanding of multimodal user inputs, and to represent meanings that the system will express as multimodal outputs to the user. This interface format should thus be adequate for representing the end result of multimodal input interpretation, and for representing the semantic content that the system will present to the user in multimodal form. It should therefore allow dialogue management, planning and reasoning modules to operate on these representations. In order to be useful for this purpose, this interface format should support the interfaces of these as well as other modules that form part of the system, and thus be adequate not only for representing the end result of semantic interpretation but also intermediate results. Something similar holds for generation. This is a second objective that follows almost immediately from the first.

Another objective in defining a well-defined representational framework for multimodal communicative acts is to allow the specification and comparison of existing application-specific representations (e.g. the M3L representation used in the SmartKom project) and the definition of new ones, while ensuring a level of interoperability between these.

Finally, the specification of a multimodal meaning representation should also be useful for the definition of annotation schemes of multimodal semantic content.

4. Basic Contrains

Given the main objective of defining meaning representations, the first and foremost basic requirements of a semantic representation framework are those that we may call `expressive' and `semantic' adequacy:

- *Expressive adequacy:* the framework should be expressive enough to correctly represent the meanings of multimodal communicative acts;
- *Semantic adequacy:* the representation structures should themselves have a formal semantics, i.e., their definition should provide a rigorous basis for reasoning (whether deductive, statistical, in the form of plan operators, or otherwise).

The second objective, of providing interface formats within a multimodal dialogue system architecture, means that `incremental' construction should be supported of intermediate and partial representations, leading up to a final representation or, if the construction of a final representation does not succeed, leading to negative feedback or another appropriate system action. This implies three further basic constraints:

- *Incrementality,* in the sense of supporting various stages of multimodal input interpretation, as well as of multimodal output generation, allowing both early and late fusion and fission;
- *Uniformity:* to make incremental processing feasible, where possible the representation of various types of input and output should be uniform in the sense of using the same kinds of building blocks and the same ways in which complex structures can be composed of these building blocks.
- *Underspecification and Partiality*: to support the representation of partial and intermediate results of semantic interpretation, the framework should allow meaning representations which are underspecified in various ways, and which capture unresolved ambiguities.

Finally, the representational framework should take into account that the design of multimodal human-computer dialogue systems is a developing area in which new research results and new technologies may bring new challenges and new approaches for the representation of multimodal meanings. This means that the representational framework should satisfy the following two constraints:

- *Openness:* the framework should not depend on a single, particular theory of meaning or meaning representation, but should invite contributions from different semantic theories and approaches to meaning representation;
- *Extensibilty*. The framework should be compatible with alternative methods for designing representation schemas (like XML), rather than support only a single specific schema.

5. Methodology

As a first step in the direction of defining a generic multimodal semantic representation form, we have to establish some basic concepts and corresponding terminology.

First, the action-based concept of meaning mentioned above, applicable to multimodal inputs in an interactive situation, means that the meaning of a multimodal `utterance' has two components: one that is often called `propositional' or `referential', and that is concerned with the entities that the utterance refers to and with their properties and relations that may be expressed in propositions, and a `functional' component that expresses a speaker's intention in producing the utterance: what effects does he want to achieve (using `speaker' in a broad, multimodal sense here)? This distinction is familiar from speech act theory, where the two components are called `propositional content' and `illocutionary force', and is also prevalent in other theories of language-based communication (see Bunt, 2000); it is often viewed as drawing a border line between semantics and pragmatics. In the analysis of multimodal interaction it is especially important to pay attention to both these aspects of meaning, since different modalities often contribute to each aspect in different ways; for instance, in spoken interaction the referential and propositional aspects of meaning are often expressed verbally, while gestures and facial expression contribute primarily to the functional aspects. The term `multimodal content' should not be confused with `propositional content', and should not make us forget that multimodal messages have meanings with functional aspects that are equally important as their propositional and referential aspects. In this document we use `multimodal content' as synonymous with `multimodal meaning', including functional aspects, and we use `semantic representation' as synonymous with `representation of meaning'.

A convenient term that has become popular in the literature on human-computer dialogue is *`dialogue act'*. This term is mostly used in an informal, intuitive way, or as a variant of `speech act; it has a formal definition in terms of the effects that a `speaker' intends to achieve through its understanding by the addressee (see Bunt, 2000), which makes it suitably precise for use in the analysis of the meaning of multimodal inputs and outputs. Without further going into definitions here, we will use the term `dialogue act' in the rest of this document. Definitions of other useful concepts can be found in Romary (2002).

As a second methodological step, we propose to distinguish the following three basic types of ingredients that would seem to go into any multimodal meaning representation framework. Each of these ingredients is discussed further in subsequent sections

1. *Basic components*: the basic constructs for building representations of the meaning of multimodal dialogue acts: types of building blocks and ways to connect them.
2. *General mechanisms*: representation techniques like substructure labeling and linking, that make the representations more compact and flexible.
3. *Contextual data categories*: types of administrative (meta-)data that do not, strictly speaking, contribute to the meanings of semantic representations, but that may nonetheless be relevant for their processing.

5.1 Basic Components

Initially, the following basic components can be identified to represent the general organization of any semantic structure:

1. temporal structures (*`events'*), to represent, for instance:
   - spoken utterances (input or output dialogue acts);
   - gestures (same);
   - noncommunicative action (like searching for information, making a calculation);
   - events, states, processes,.. in the discourse domain, representing meanings of verbs and possibly other linguistic expressions;
2. referential structures (*`participants'*), to represent, for instance:
   - the speaker of an input utterance, or the person performing a gesture;
   - the addressee of a system output dialogue act;
   - individuals and objects participating in a semantic event
3. *restrictions* on temporal and referential structures, to represent, for instance:
   - the type(s) of dialogue, act associated with an utterance;
   - a gesture type, assigned to a gesture token
4. dependency structures, representing *semantic relations* between temporal and/or referential structures, for instance:
   - participant roles (like SPEAKER, ADDRESSEE, AGENT, THEME, SOURCE, GOAL,..)
   - discourse/rhetorical relations
   - temporal relations.

It may be noted that linguistic semantic phenomena that have been studied extensively in relation to the needs of underspecific representation, such as quantification and modification, can also be represented with these basic components. For instance, a quantified statement like `Three men moved the piano' can be represented as a move-event involving a group of three men and a piano, where the collectiveness and the group size of the set of men that form the agent of the event are represented by means of restrictions on the event.

5.2 General Mechanisms

In addition to these basic components, certain general mechanisms are important to make meaning representations suitable for representing partial and underspecified meanings, to give the representations a more manageable form, and to relate them to external sources of information. Examples of such mechanisms are:

1. *substructure labeling*: assigning labels to subexpressions and allowing the use of these labels, instead of the substructures that they label, as arguments in other subexpressions;
2. *argument underspecification*: partial or underspecified representations can be constructed using labels in argument positions; restrictions on labels can represent limitations on the ways in which such variables can be instantiated by labels of substruct - ures elsewhere in the representation;
3. *restrictions on label values*: see previous mechanism. Alternatively, *disjunctions*, or *lists* of labels can be used to represent ambiguity or partiality;
4. *structure sharing*, as in typed feature structures, makes it possible to represent that a certain part of the representation plays more than one role, e.g. a participant may be both agent and theme in a semantic event, or may be the speaker of an utterance and the performer of a gesture, as well as the agent in a semantic event expressed by the multimodal dialogue act;
5. *linking to domain models* (types and instances) to anchor meaning representations in the domain of discourse;
6. *linking to lower levels*, such as syntactic structure, prosodic cues, gestural trajectories,.. is useful for tying a purely semantic representation to lower-level information that has given rise to it, and that may not yet have been fully interpreted.

5.3 Contextual Data Categories

Finally, meaning representations will need to be annotated with general categories of administrative information, both globally and also at the level of subexpressions, to capture certain information which is not found inside the elements of interactive behaviour, but which is potentially relevant for their interpretation and generation, such as:

1. Environment data, for instance:
- time stamps and spatial information (when and where was this input received, etc.)
2. Processing information, such as:
- which module has produced this representation; what is its level of confidence, etc.
3. Interactional information:

6. Technical Backgound: XML

At this stage, we should say a word about what appear to be the unavoidable technical choices for the definition of a multimodal content representation format that would be used, among other possibilities, to exchange information between processing modules within a man-machine dialogue system. As a matter of fact, XML, as defined by the World Wide Web Consortium, appears to be the best candidate so far (and probably for quite a long time) to represent information structures intended to be transmitted across a network. In the following section, we give a very brief overview of XML, which we will then use to illustrate some of the principles mentioned above by means of a concrete example.

XML (eXtended Markup Language) is a simplified (but also in some respects enhanced) version of SGML. It provides a syntax for document markup as well as for the description of the set of tags to be used in classes of documents (a so-called DTD, Document Type Definition). An XML document is made of three parts:

- An XML declaration, which, beyond identifying that the current document is an XML one, allows one to declare the character encoding scheme used in the document (e.g. iso-8859-1, utf-8, etc.);
- A document type declaration, which can point to a DTD. This section can be omitted;
- An XML instance corresponding to the actual data represented by the document.

XML makes an important distinction between a *well-formed* document, which only contains the XML declaration and a syntactically conformant instance, and a *valid* one, where the instance is also checked against the associated DTD.

Among other characteristics, we mention the following important properties of XML:

- XML is both Unicode and ISO 10646 compatible[1]
- XML comes along with a specific mechanism, called *namespaces*, allowing one to combine, within the same document, markup taken from multiple sources. This very powerful mechanism, which is in particular the basis for XSLT and XML schemas, allows more modularity in the definition of an XML structure and also to reuse components defined in another context;
- XML provides a general attribute 'xml:lang' to indicate the language used in a given element (see above).

The W3C also provides three very important recommendations for traversing XML documents, namely:

- XPath, which describes a syntax and associated mechanisms to move within a document instance;
- XPointer, which allows one to indicate a location within a document and is based upon the XPath recommendation;
- XLink, which allows one to combine and qualify a set of pointers to describe a link between them.

[1] The W3C has put pressure on both ISO and the Unicode consortium to make sure that they would not diverge in their parallel work on the definition of a universal character encoding scheme.

These three recommendations are important for instance when one wants to relate some information produced by a given processing level and the information that has been used as input for those processes.

Still, it should be noticed that the existence of such a widely recognized *metalanguage* as XML does not solve our problems for representing multimodal content. First, XML by itself does not come with a formal semantics for its tags, and thus does not satisfy the requirement of semantic adequancy. Second, the requirements of flexibility and extensibility forbid us to try to standardize once and for all a precise XML format, but rather think of providing concepts and tools for anyone to be able to design his or her own format, while preserving interoperability conditions with someone else's choices. This is the spirit in which work has already been done within TC37/SC4 for the definition of TMF (Terminological Markup Framework; ISO 16642, under DIS ballot) and which has recently been taken over to deal with morphosyntactic and syntactic annotation (see (Ide & Romary, 2001a and Ide & Romary, 2001b, respectively). The basic assumption that we make is that there exists an entire class of document formats that can be modelled by combining a *metamodel,* that is an abstract structure shared by all documents of a given type (e.g. syntactic annotation document), with a choice of the data categories that may be associated with the various levels of the metamodel. Such a description can be seen as a specification of the document format, which can be instantiated by providing XML representations for the metamodel and the data categories. In such a view, if a community of researchers and implementers agrees on the definition of a reduced set of metamodels for language resources, the actual choice of data categories is left to the responsibility of a specific application. In this framework, the interoperability between formats is ensured by providing a data category registry which gathers, together with precise reference and definition, the various data categories needed for a particular field.

In the case of multimodal content representation we thus advocate that, beyond agreement on the basic components and mechanisms for instance as described in this paper, which could go into the definition of an actual metamodel for content representation, one should not try to standardize a particular XML format more precisely (though we need to make specific choices to illustrate our approach with concrete examples, see below).

7. A simple example

In the following, we illustrate the possible combination of basic components, general mechanisms, and contextual data categories into a multimodal meaning representation. This representation exemplifies the general methodology that we suggested here, by taking up a sample semantic representation derived from an initial example expressed in the ULF+ format (ULF+ is a slightly updated version of a semantic representation language that was developed successively in the PLUS dialogue project, see Geurts and Rentier, 1993, and in the multimodal DENK project; see Bunt et al., 1998; Kievit, 1998).

In the XML excerpt below (corresponding to the sentence "I want to go from Paris to Stuttgart" uttered by a speaker named Peter), we have extended the original ULF+ representation to introduce the notion of dialogue act, whose participants are the speaker and the system. This example is intended to show how we can differentiate between three types of information in such a representation:

- The instantiation of the semantic content representation metamodel as an XML outline (shown in underlined characters), which organizes the general information layout of the data to be represented;
- The actual information units describing the various levels in the XML outline (shown in gray characters);
- The generic mechanisms used to combine events, participants, restrictions and relations (indicated in **bold characters**).

The specific choices made in this example to represent the metamodel or the data categories as XML objects are only one possibility among many, and this does not affect the formal semantics of the underlying information structure. More precisely, the following explanation may help to clarify the example:

- The <semRep> element corresponds to the semantic representation of one elementary utterance or dialogue act. It is identified uniquely by an id attribute;
- The <event> element is used in this example to represent both the dialogue act proper ("e1") and the event expressed by the corresponding linguistic content ("e2");
- The <participant> element is used to represent the various entities involved in the events. Events and participants being related to one another by means of <relation> elements (with source and target attributes pointing to the corresponding arguments of the relation.

The various levels are then further described by a number of data categories, chosen here to illustrate the wide variety of possible cases. Notice the use of an <alt> structure to illustrate the case where an ambiguity would remain at a given step of analysis, each possibility being associated with a certainty evaluation ('cert' attribute). In accordance with the methodology developed in TMF, the name of the corresponding XML elements and attributes should not be the object of standardization, data categories being defined by abstract properties.

```
<semRep id="rep1">
    <event id="e0">
      <evtCat>utterance</evtCat>
      <speaker target="Peter"/>
      <adressee target="System"/>
```

```
    <alt>
        <dialAct cert="0.8">Order</dialAct>
        <dialAct cert="0.3">Inform</dialAct>
    </alt>
</event>

<participant id="Peter">
    <!-- A description of the speaker that can be referendum elsewhere in the document -->
</participant>

<event id="e1">
    <tense>present</tense>
    <voice>active</voice>
    <wh>none</wh>
    <evtType>wanttogo</evtType>
    …
</event>

<participant id="x">
    <lex>I</lex>
    <synCat>Pronoun</synCat>
    <num>sing</num>
    <pers>first</num>
    …
</participant>

<participant id="y">
    <lex>Nancy</lex>
    <synCat>ProperNoun</synCat>
    <pers>third</num>
    …
</participant>

<participant id="z">
    <lex>Stuttgart</lex>
    <synCat>ProperNoun</synCat>
    <pers>third</num>
    …
</participant>

<relation source="x" target="e1">
    <role>agent</role>
</relation>

<relation source="y" target="e1">
    <role>source</role>
</relation>

<relation source="y" target="e1">
    <role>goal</role>
</relation>
</semRep>
```

8. Action Plan

The variety of existing theoretical approaches, as well as the wide number of factors to be considered makes it very difficult to devise from scratch a truly generic framework for multimodal content representation. As a consequence it is necessary to involve, beyond the possibilities offered by the definition of a working group on this topic in TC37/SC4, as large a community of experts as possible in the development of such a framework. This is why we suggest that the work shall be initially conducted within a dedicated working group of SIGSEM (Special Interest Group on Computational Semantics of the Association of Computational Linguistics), which would be, right from the beginning, a liaison with TC37/SC4. This group would prepare a working draft, which would then be submitted to ISO.

Doing so, it would also be easier to ensure a proper interaction with other interested communities, in particular the people working on multimedia representation (SIGMedia, in complement to the existing liaison between MPEG and TC37/SC4) and on discourse and dialogue (SIGDial).

The agenda would thus be the following:

- Refining the workplan on the basis of the present paper at the TC37/SC4 Preliminary Meeting in Jeju (Korea) in February 2002.
- Presenting a position paper at the LREC workshop on "International Standards of Terminology and Language Resources Management" in May 2002.
- First working group meeting in conjunction to IWCS-5 (5th International Workshop on Computational Semantics) in Tilburg, the Netherlands, in January 2003.

9. References

Bunt, H.C., 2000. Dialogue pragmatics and context specification. In H. Bunt and W. Black, editors, *Abduction, Belief and Context in Dialogue.* John Benjamins Publishing Company, Amsterdam.

Bunt, H.C., R. Ahn, R.J. Beun, T. Borghuis and C. van Overveld, 1998. Multimodal cooperation with the DENK system. In: H.C. Bunt, R.J. Beun and T. Borghuis, editors *Multimodal Human-Computer Communication*. Springer, Berlin.

Bunt, H.C., M. Kipp, M.T. Maybury and W. Walster, forthcoming. *Fusion and Coordination for Multimodal Interaction. Roadmap, Arcitecture, Tools, Semantics.*

Geurts, B. and G. Rentier, 1993. Quasi-logical form in PLUS. Internal Report, Esprit Project P5254, *A Pragmatics-based Language Understanding System*. Tilburg University.

Ide, N., A. Kilgariff and L. Romary 2000, A Formal Model of Dictionary Structure and Content. *Proceedings Euralex 2000,* Stuttgart.

Ide, N. and L. Romary, 2001a, Standards for Language Resources, *IRCS Workshop on Linguistic Databases*, 11-13 December 2001, University of Pennsylvania, Philadelphia, USA.

Ide, N. and L. Romary, 2001b, A Common Framework for Syntactic Annotation, *Proceedings of ACL'2001*, Toulouse. Morgan Kaufman, Menlo Park.

Kievit, L.A., 1998. *Context-driven Natural Language Interpretation*. Ph.D. Thesis, Tilburg University.

Maybury, M.T. editor, 1993. *Intelligent Multimedia Interfaces*. AAAI/MIT Press. 405 pp. ISBN 0-262-63150-4 (www.aaai.org:80/Press/Books/Maybury1, mitpress.mit.edu/book-home.tcl?isbn=0262631504)

Maybury, M.T. and W. Wahlster, editors, 1998. *Readings in Intelligent User Interfaces*. Morgan Kaufmann Press. (www.mkp.com/books_catalog/catalog.asp?ISBN=1-55860-444-8)

Romary, L., 2002. MMIL requirements specification. Project MIAMM – *Multidimensional Information Access using Multiple Modalities.* EU project IST-20000-29487, Deliverable D6.1. LORIA, Nancy.

Where will the Standards for Intelligent Computer-Assisted Language Learning Come from?

Lars Borin

Computational Linguistics, Department of Linguistics,
Stockholm University, SE-106 91 Stockholm, Sweden
and
Department of Linguistics, Uppsala University,
Box 527, SE-751 20 Uppsala, Sweden

lars.borin@ling.su.se, lars.borin@ling.uu.se

Abstract

Intelligent computer-assisted language learning—Intelligent CALL, or ICALL—can be defined in a number of ways, but one understanding of the term is that of CALL incorporating language technology (LT) for e.g. analyzing language learners' language production, in order to provide the learners with more flexible—indeed, more 'intelligent'—feedback and guidance in their language learning process. However, CALL, ICALL and LT have been three largely unrelated research areas, at least until recently. In the world of education, 'e-learning' and 'ICT-based learning' are the new buzzwords. Generally, what is meant is some kind of web-based setup, where course materials are delivered via the Internet or/and learners are collaborating using computer-mediated communication (CMC). An important trend in ICT-based learning is that of standardization for reusability. Standard formats for all aspects of so-called 'instructional management systems' are rapidly gaining acceptance in the e-learning industry. Thus, learning applications will need to support them in order to be commercially viable. This in turn means that the proposed standards should be general enough to support all conceivable kinds of educational content and learning systems. In this paper, we will discuss how ICALL applications can be related to the various standards proposals, basing our discussion on concrete experiences from a number of (I)CALL projects, where these standards are used or where their use has been contemplated.

1. Introduction

For some years, I have been actively involved in trying to combine computer-assisted language learning (CALL) with language technology (LT) (a.k.a. computational linguistics (CL), language engineering (LE), or natural language processing (NLP)) into what is often referred to as "Intelligent CALL" (ICALL), both as a teacher of CALL to LT students at the university, and as a researcher involved in a number of research efforts dealing with CALL/ICALL (see below), and also with neighboring areas, such as computer support for lesser used and lesser taught languages (Borin, 2000a; Allwood and Borin, 2001; Nilsson and Borin, 2002), and contrastive linguistic studies using computational methods (Borin, 1999; Borin, 2000b; Borin and Prütz, 2001; Borin and Prütz, 2002).

The present paper flows from a desire to make ICALL benefit from, as well as inform, ongoing standardization efforts in the computational linguistics and e-learning communities.

The rest of the paper is organized in the following way. First, I will try to sort out the relationships between CALL, LT, artificial intelligence (AI), and ICALL. Then I will describe briefly ongoing standardization work in the e-learning and CL communities, and some of the standards proposals that this work has produced. Following that, I will turn to a description of some (I)CALL projects in which I have been or am currently involved, where these standards are used or where their use has been contemplated, namely the SweLL Didax project, the LingoNet project, 'Corpus based language technology for computer-assisted learning of Nordic languages', the SVANTE learner corpus project, and 'IT-based collaborative learning in Grammar'. Finally, I will discuss the situation of ICALL with regard to this standardization work, in order to form an understanding of where we stand at the moment, but more importantly, of where we would like to go from here.

2. CALL, LT and ICALL

Intelligent computer-assisted language learning—Intelligent CALL, or ICALL—has been defined in a number of ways, but one understanding of the term relevant here is that of CALL incorporating LT techniques for e.g. analyzing language learners' language production or modeling their knowledge of a second/foreign language in order to provide them with more flexible—indeed, more 'intelligent'—feedback and guidance in their language learning process.

CALL, ICALL and LT have been three largely unrelated research areas, at least until recently:

1. The CALL 'killer apps' have been e-mail, chat and multimedia programs, developed and used by language teaching professionals with very little input from LT research (Pennington, 1996; Chapelle, 1997; Chapelle, 1999; Chapelle, 2001; Levy, 1997; Salaberry, 1999). The only kind of LT which has had any kind of impact on the CALL field is corpus linguistics, and even in this case it has been the Humanities Computing 'low-tech' kind of corpus linguistics,

rather than the kind pursued in LT (the latter is sometimes referred to as "empirical natural language processing").

2. ICALL has often been placed by its practitioners in the field of artificial intelligence (AI), rather than in LT (e.g. Swartz and Yazdani (1992); Holland et al. (1995)), more specifically in the subfield of AI known as *intelligent tutoring systems* (ITS) (e.g. Frasson et al. (1996); Goettl et al. (1998)). Partly for this reason, work on ICALL has proceeded, by and large, without feedback into the LT community.

3. But on the other hand, in LT in general, (human) language learning has not been seen as an application area worth pursuing. In the recent broad *State of the art of human language technology* overview edited by Cole et al. (1996), 'language learning' does not appear even once in the index, and there is no section on CALL. Certainly there are some exceptions to this general trend; there have been occasional COLING (*International Conference on Computational Linguistics*) papers on ICALL, although few and far between (e.g. Borissova (1988); Zock (1996); Schneider and McCoy (1998)), and there is a research group in Groningen which has been working very actively on LT-based CALL applications for quite some time (Nerbonne and Smit, 1996; Dokter, 1997; Dokter, 1998; Dokter and Nerbonne, 1997; Dokter et al., 1997; Jager et al., 1998). The situation has been changing somewhat only in the last few years, however, with dedicated workshops on language learning applications of CL being arranged in connection with LT conferences and the like (e.g. Olsen (1999); Schulze et al. (1999); Efthimiou (2000)).

3. Standardization in e-Learning and Language Technology

3.1. E-learning standardization efforts

In the world of education, 'e-learning' and 'ICT-based learning'[1] are the new buzzwords (see, e.g., European Commission (2000)). Generally, what is meant is some kind of web-based setup, where course materials are delivered via the Internet or/and learners are collaborating using computer-mediated communication (CMC) methods.

An important trend in ICT-based learning is that of standardization for reusability. Standard formats are defined for all aspects of so-called 'instructional management systems'. Thus, not only educational content formats are agreed upon, but also course structure formats, test formats, as well as how their interaction with recordkeeping systems used in education should take place. There is a number of organizations working on standards in the e-learning area, the most important ones being IMS (Instructional Management System Inc. `http://www.imsproject.org/`), IEEE's LTSC (Learning Technology Standards Committee; `http://ltsc.ieee.org/`), the American Department of Defence ADL (Advanced Distributed Learning; `http://www.adlnet.org/`) initiative, and the European ARIADNE project. Standards being developed by these and other bodies include educational metadata (Learning Objects Metadata – LOM; Anderson and Wason (2000)), test formats (IMS Question and Test Interoperability – QTI; Smythe and Shepherd (2000)), content packaging formats (IMS Content Packaging; Anderson (2000)), modular courseware (ADL SCORM; Dodds (2001)), and others (see, e.g. the IMS and LTSC websites referred to above). At least some of these standards are rapidly gaining acceptance in the e-learning industry. Thus, learning applications will need to support them in order to be commercially viable. This in turn means that the proposed standards should be general enough to support all conceivable kinds of educational content and learning systems.

The general idea is to create standards which are

> "pedagogically neutral, content-neutral, culturally neutral and platform-neutral"
> (Farance and Tonkel, 1999, 9),

and which support. . .

> "common, interoperable tools used for developing learning systems [. . .]
>
> a rich, searchable library of interoperable, "plug-compatible" learning content [. . .]
>
> common methods for locating, accessing and retrieving learning content"
> (Farance and Tonkel, 1999, 14)

One may certainly entertain doubts as to the general attainability of these goals, but one cannot afford to ignore the huge amount of time and labor invested in pursuit of their fulfillment by the organizations mentioned above and others. This being so, it is of course not unimportant if learning and teaching within a particular field—such as language learning—is adequately covered by the proposed standards or not.

3.2. Standardization in Language Technology/Computational Linguistics

In the LT world, too, standardization efforts are legion, and a recurring theme at the LREC (Language Resources and Evaluation Conference) series of conferences.

There is LT standardization work going on at least in the areas of

- resource storage and exchange: TIPSTER (Grishman et al., 1997), ATLAS (Bird et al., 2000), XCES (Ide et al., 2000);
- resource annotation: XCES (Ide et al., 2000), EAGLES (e.g., tagsets: see Monachini and Calzolari (1996));
- resource metadata: OLAC, ISLE (Wittenburg et al., 2000);
- resource presentation and manipulation, and software integration: THISTLE, GATE (Cunningham, 2001), KABA (Olsson, 2002).

[1] ICT is to be read out "Information and Communication Technologies".

To the best of my knowledge, however, the work within LT on resource markup and annotation has not been informed by language learning applications or by the work done on compiling and investigating so-called learner corpora by applied linguistics researchers (see, e.g., Granger (1998)).

4. (I)CALL Case Studies

In this section, we will look at some CALL research projects, where the issue of combining (I)CALL applications with e-learning standards has arisen in various ways.

4.1. Didax

Didax – the Digital Interactive Diagnostic Administering and Correction System, is a project in the framework of the *Swedish Learning Lab* (SweLL), a research effort funded by the Knut & Alice Wallenberg Foundation as part of the larger *Wallenberg Global Learning Network* endeavor, where a number of centers—or "nodes"—worldwide receive funding for exploring the use of ICT and other new technologies in higher education.

At present, there are three nodes in the WGLN: (1) SweLL, with three participating institutions of higher education, (1a) the Royal Institute of Technology and (1b) Karolinska Institutet in Stockholm, and (1c) Uppsala University, (2) the Stanford Learning Lab (SLL), at Stanford University, California, USA, and (3) Learning Lab Lower Saxony (L3S), at the University of Hannover, Germany. SweLL research is currently organized into a multi-tiered structure, with two top-level 'projects' subdivided into a number of 'experiments´. Each experiment is further subdivided into 'tracks', where each track in turn typically is made up of several research teams cooperating on related research issues. Our work on Didax is thus carried out in the *Digital Resources in the Humanities* (DRHum) track of the *Archives – Portfolios – Environments* (APE) experiment of the SweLL project *New meeting places for learning – New learning environments*.

The Didax research team currently consists of three computational linguists and one SLA researcher, but we also cooperate closely with the other DRHum research teams, drawing on the other kinds of competence found there, especially the teams working with digital archives for humanities teaching, as well as with the Uppsala Learning Lab e-folio project group.

The end result of the Didax project is supposed to be a web-based language testing environment, which will provide both students and teachers with a more flexible format for taking, marking, constructing and setting diagnostic language tests in higher education. In Figure 1, the overall architecture of Didax is shown. The three Didax clients (*teacher – setting test*, *teacher – marking test*, and *student*) run in ordinary web browsers. There is nothing out of the ordinary to be seen in any of the client interfaces. This is quite deliberate. Most of the innovation is hidden under the surface, and the interface is a familiar one from many web applications. Didax is described in more detail by Borin et al. (2001).

4.2. LingoNet

LingoNet is a one-year R&D project funded by the Swedish Agency for Distance Education. The project is a cooperation between the Divison of IT Services and the Department of Humanities, Mid Sweden University, and the Department of Linguistics, Uppsala University (see `http://www.mitt.mh.se/lingonet/`).

The aim of the LingoNet project is to build a 'language lab on the Internet', i.e. a web site with a collection of language training resources to be used in higher education, both locally and in distance education. Even though the point of departure for the LingoNet project is the traditional language lab, we actually envision a more general language training resource than this, i.e. a 'computer language lab', rather than a 'computerized version of the tape recorder-based language lab', as the idea is not only to transfer older techniques into this new technology, but also to exploit the additional possibilities offered by the new technology itself, including the incorporation of LT-based language learning resources in the LingoNet lab.

Specifically, in the LingoNet project, we make systematic use of quality control and metadata. It is a well-known fact that the information to be found on the web on any topic is, not only abundant in almost all cases, but also—to put it mildly—of extremely varying quality. At the same time, web search engines are still fairly primitive, so that finding educational resources, appropriate as to their content and level—regardless of their quality—in itself takes some work (Howard Chen, 1999, 24f.). It is only after they have been found that the real work begins, however, when the chaff—resources which are of low quality or of the wrong kind—is to be separated from the wheat—the resources which we can use for our educational purpose, i.e. educational web resources which are quality controlled and classified as to their content and level. In the LingoNet project, the quality control and metadata markup are done by academic language teachers. For more details about the LingoNet project, see Borin and Gustavsson (2000).

4.3. Corpus based language technology for computer-assisted learning of Nordic languages

'Corpus based language technology for computer-assisted learning of Nordic languages', or in short, the Squirrel project, is funded by the Nordic Council of Ministers, and represents a collaboration between the University of Helsinki in Finland, the research foundation SINTEF in Norway, and Stockholm University in Sweden (see `http://www.informatics.sintef.no/projects/CbLTCallNordicLang/squirrel.html`).

One of the aims of the Squirrel project has been to build a prototype web browser for students and teachers of Nordic languages as a second language, which will help them to find practice texts on the web according to the three parameters *language*, *topic*, and *text difficulty* (Nilsson and Borin, 2002). For more details about the Squirrel project, see Borin et al. (2002)

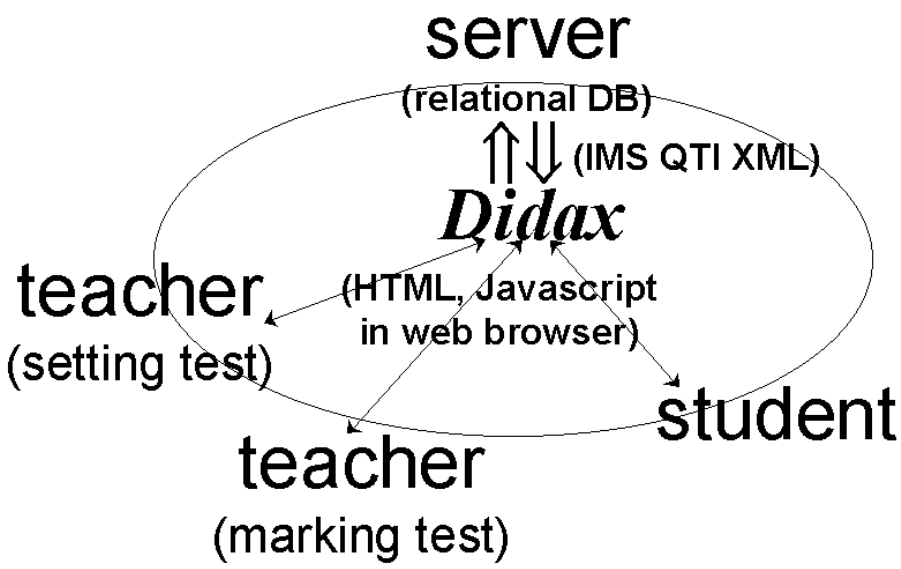


Figure 1: The anatomy of Didax

4.4. SVANTE

SVANTE (SVenska ANdraspråksTexter – Swedish Second Language Texts) is a loose collaboration between linguists, computational linguists, and teachers of Swedish as a second language, with the aim of creating a versatile learner corpus of written Swedish, to complement the learner corpora of spoken Swedish that already exist (see `http://www.ling.uu.se/lars/SVANTE/`). The SVANTE project is partly funded by VINNOVA within the CrossCheck second language Swedish grammar checking project (see `http://www.nada.kth.se/theory/projects/xcheck/`).

4.5. IT-based collaborative learning in Grammar

'IT-based collaborative learning in Grammar' is a collaborative project, funded by the Swedish Agency for Distance Education, with partners in the Linguistics Departments at the universities in Uppsala and Stockholm, and the IT Department and two language departments at Uppsala University. This project revolves around two fundamental assumptions:

1. The use of web-based communication and collaboration technologies will help us make make basic grammar courses better and more effective for students and teachers alike;

2. Language resources originally developed in a research setting, such as tagged and parsed corpora (of Swedish in our case) and grammar writing workbenches, can be (re)used in the context of teaching grammar (Borin and Dahllöf, 1999).

Perhaps I should clarify at this point that this is not primarily an application intended for *language* students, but rather for students of Linguistics and Computational Linguistics, although we believe that it will be useful also as a component in language courses (Saxena and Borin, 2002).

4.6. Relation to e-learning standards and to ICALL

These projects are variously related to ICALL on the one hand and to e-learning standards on the other:

- Didax is not an ICALL project *per se*, but creates an infrastructure which can be used for ICALL applications, and thus must be able to accomodate them. It uses the IMS QTI, and the IEEE, IMS, ARIADNE LOM emerging standards.

- LingoNet is not an ICALL project either, but it goes without saying that among the more exciting possibilities for a web-based language lab are language training applications built on LT methods and resources; hence, we must take this into consideration in designing the underlying language lab format. Like Didax, LingoNet can be considered as an infrastructure project which should be able to accomodate ICALL applications. The standards involved are IMS Content Packaging, and IEEE, IMS, ARIADNE LOM.

- Squirrel is an ICALL project, which does not (yet) utilize any of the proposed e-learning standards, but we see how e.g. the LOM could be used to mark up the located text resources, e.g. for inclusion in something like the LingoNet database.

- SVANTE forms an integral part of an ICALL project, namely the CrossCheck second language grammar checking project, but SVANTE itself is more in the way of a linguistic resource project, where LT standards for basic markup and linguistic annotation of the texts are important.

- 'IT-based collaborative learning in Grammar' is very much an ICALL project. At this initial stage of the project (it started in January 2002), there are still a number of implementational details left to be decided.

However, we would certainly like to make our learning resources as widely useful as possible, meaning, i.a.,

1. that they should be—wholly or in part—easy to integrate into other e-learning environments, but also
2. that it should be easy to use corpus resources for other languages than Swedish in our application.

The first requirement implies the existence and use of general standards for e-learning applications, while the fulfillment of the second requirement certainly would be facilitated by standardization of language resources.

5. So, where will the Standards for ICALL Come from?

Summing up the foregoing, we may say that there are three communities which would benefit from closer interaction, because of a considerable overlap in their goals, but which thus far have pursued these goals separately:

1. The 'ordinary' CALL community—including those researchers working with learner corpora—has extremely tenuous links to LT (see e.g. Chapelle (2001, 32ff.)), and, as far as I have been able to acertain, none at all to the ongoing e-learning standardization work mentioned in section 3.1. above.

2. Nor is the e-learning community working on any standardization for *language learning* (as opposed to *learning* in general). For example, the IMS Question and Test Interoperability (QTI) proposal specifies five test question response types, which can be rendered in up to three different formats (Smythe and Shepherd, 2000, 17). However, for the 'IT-based collaborative learning in Grammar' application, as well as for many other of the corpus-based CALL applications found in the literature, a response type "select (portion/s of) a text" would certainly be good to have.[2]

3. The LT community is not involved in any standardization effort for *language learning* information (as opposed to *language* information in general). The kinds of standards that come to mind first are those involving linguistic annotation schemes, with regard to both their content and their form:

   So-called *learner interlanguage* is characterized by a number of linguistic features absent from the native-speaker version of the target language (and sometimes absent from the learner's native language as well (Richards and Sampson, 1974, 6)). Interlanguage goes through a number of stages, terminating in a final (hopefully close) approximation of the target language. This has some implications for linguistic annotations of learner language production, whether in learner corpora (longer texts) or in analyzers of free learner language production in ICALL language exercises. Thus, part-of-speech (POS) tagging or parsing of learners' interlanguage may have to deal with categories absent from the canonical target language grammar as reflected in an LT standard, etc., but which can be related either to categories in the learner's native language, to universally unmarked categories, to a conflation of target categories, to the pedagogy used, to some combination of these, etc. (Cook, 1993, 18f.). The status of a given linguistic element can change from one language learning stage to another, e.g. the unmarked form in a morphological paradigm becoming functionally more and more specified, as the learner acquires the marked forms and their functions.[3]

   Hence, multiple linguistic annotations of the kind proposed for XCES (Ide et al., 2000) and ATLAS (Bird et al., 2000; Cotton and Bird, 2002) are a necessity for language learning applications of e.g. language corpora.[4] In addition to providing multiple annotations of the same linguistic object (a word, phrase, etc.), the annotations should also be relatable to each other, making it possible to relate an analysis of a form in learner production to the (inferred) intended interpretation of this form, for providing appropriate feedback to the learner. The linguistic categories provided by annotation standards would need to be different from the ones used by native speaker experts (which is arguably most often the kind of annotation aimed for now) if they are to be used for formulating feedback to language learners. They would also have to be different for different kinds of learners, depending on their level, background, native language, etc.

   Standardization of (formats for) *error typologies* would also be desirable. Again, this desideratum is not exclusive to language learning applications; work on grammar and style checkers for native speakers would also benefit from standardized formats for error typologies.

   In the same way as the learner's language progresses through successively more advanced stages, the authentic language that the learner is exposed to as part of her learning process should be successively more complex, in a linguistic sense. This is the main motivation for the Squirrel web search application described above (Nilsson and Borin, 2002). Here, there is consequently a need for a classification and concomitant annotation scheme which relates linguistic complexity to language learning stages, for applications where corpora are used for e.g. generating lan-

[2] In the QTI specification, there is actually a sixth response type response-extension, intended for proprietary response types, but the predefined types will always determine the 'path of least resistance', at least for many users.

[3] Here I have in mind cases such as when e.g. learners of English initially use the infinitive (or sometimes gerund) as their only—and hence extremely polyfunctional—verb form, and then gradually start using other forms (tensed forms in finite clauses, etc.), which then usurp, as it were, some of the functions of the initial forms.

[4] Multiple annotations actually seem necessary for other reasons as well, see e.g. Sampson (2000).

guage learning exercises.

In language learning applications, the need to cater for *bilingual* and *multilingual* text materials is evident, which raises the issues of how to handle multiple writing systems in a standardized way, e.g. left-to-right and right-to-left writing in the same text corpus (the latter issue is raised by Cotton and Bird (2002) as still not having been determined for ATLAS).

Hopefully, the state of affairs depicted here is really due more to lack of interaction than anything else, and if the present paper can be instrumental in bringing about this interaction, it will have served its purpose.

6. Acknowledgements

The work reported herein was carried out partly within the project 'Corpus based language technology for computer-assisted learning of Nordic languages', in the framework of the Nordic Language Technology Research Program 2000–2004 (Holmboe, 2002), funded by the Nordic Council of Ministers through Nordisk Forskeruddannelsesakademi (NorFA), partly within the project 'Digital resources in the humanities', funded by the Knut & Alice Wallenberg Foundation, as part of the Wallenberg Global Learning Network, and partly within the CrossCheck/SVANTE project, funded by VINNOVA within the Language Technology Program.

7. References

Jens Allwood and Lars Borin. 2001. Datorer och språkteknologi som hjälpmedel i bevarandet av romani • Computers and language technology as an aid in the preservation of Romani. Plenary presentation at the symposium *Romani as a language of education: possibilities and restrictions today*. Göteborg University.

Thor Anderson and Tom Wason. 2000. IMS learning resource meta-data information model. final specification version 1.1. Retrieved from the WWW in August 2000: `http://www.imsproject.org/metadata/mdinfov1p1.html`.

Thor Anderson. 2000. IMS content packaging information model. final specification version 1.0. Retrieved from the WWW in October 2000: `http://www.imsproject.org/content/packaging/cpinfo10.html`.

Steven Bird, David Day, John Garofolo, John Henderson, Christophe Laprun, and Mark Liberman. 2000. ATLAS: a flexible and extensible architecture for linguistic annotation. In *Proceedings of LREC 2000*, pages 1699–1706, Athens. ELRA.

Lars Borin and Mats Dahllöf. 1999. A corpus-based grammar tutor for Education in Language and Speech Technology. In *EACL'99. Computer and Internet Supported Education in Language and Speech Technology. Proceedings of a Workshop Sponsored by ELSNET and The Association for Computational Linguistics*, pages 36–43, Bergen. University of Bergen.

Lars Borin and Sara Gustavsson. 2000. Separating the chaff from the wheat: Creating evaluation standards for web-based language training resources. In Khaldoun Zreik, editor, *Learning's W.W.W. Web Based Learning, Wireless Based Learning, Web Mining. Proceedings of CAPS'3*, pages 127–138, Paris. Europia.

Lars Borin and Klas Prütz. 2001. Through a glass darkly: Part of speech distribution in original and translated text. In Walter Daelemans, Khalil Sima'an, Jorn Veenstra, and Jakub Zavrel, editors, *Computational Linguistics in the Netherlands 2000*, pages 30–44. Rodopi, Amsterdam.

Lars Borin and Klas Prütz. 2002. New wine in old skins? A corpus investigation of L1 syntactic transfer in learner language. To be presented at the *International Conference on Teaching and Language Corpora (TaLC) 2002*, Bertinoro, Italy.

Lars Borin, Karine Åkerman Sarkisian, and Camilla Bengtsson. 2001. A stitch in time: Enhancing university language education with web-based diagnostic testing. In *20th World Conference on Open Learning and Distance Education The Future of Learning – Learning for the Future: Shaping the Transition. Düsseldorf, Germany, 01–05 April 2001. Proceedings*, Oslo. ICDE. (CD-ROM: ISBN 3-934093-01-9).

Lars Borin, Lauri Carlson, and Diana Santos. 2002. Corpus based language technology for computer-assisted learning of Nordic languages: Squirrel. Progress report September 2001. In Henrik Holmboe, editor, *Nordisk sprogteknologi. Nordic Language Technology*. Museum Tusculanums Forlag, Københavns Universitet, Copenhagen.

Lars Borin. 1999. Alignment and tagging. In *Working papers in Computational Linguistics & Language Engineering 20*, pages 1–10. Department of Linguistics, Uppsala University.

Lars Borin. 2000a. A corpus of written Finnish Romani texts. In Donncha Ó Cróinin, editor, *LREC 2000. Second International Conference on Language Resources and Evaluation. Workshop Proceedings. Developing Language Resources for Minority Languages: Reusability and Strategic Priorities*, pages 75–82, Athens. ELRA.

Lars Borin. 2000b. You'll take the high road and I'll take the low road: Using a third language to improve bilingual word alignment. In *Proceedings of the 18th International Conference on Computational Linguistics*, pages 97–103, Saarbrücken. Universität des Saarlandes.

Elena Borissova. 1988. Two-component teaching system that understands and corrects mistakes. In *COLING Budapest. Proceedings of the 12th International Conference on Computational Linguistics. Vol I*, pages 68–70, Budapest. John von Neumann Society for Computing Sciences.

Carol Chapelle. 1997. CALL in the year 2000: Still in search of research paradigms? *Language Learning & Technology*, 1(1):19–43. `http://llt.msu.edu/`.

Carol Chapelle. 1999. Research questions for a CALL research agenda: a reply to Rafael Salaberry. *Language Learning & Technology*, 3(1):108–113. `http://llt.msu.edu/`.

Carol Chapelle. 2001. *Computer Applications in Second Language Acquisition*. Cambridge University Press,

Cambridge.

Ron Cole, Joseph Mariani, Hans Uszkoreit, Annie Zaenen, and Victor Zue, editors. 1996. *Survey of the State of the Art in Human Language Technology*. Cambridge University Press, Cambridge. Also as `http://cslu.cse.ogi.edu/HLTsurvey/`.

Vivian Cook. 1993. *Linguistics and Second Language Acquisition*. Macmillan, London.

Scott Cotton and Steven Bird. 2002. An integrated framework for treebanks and multilayer annotations. In *Proceedings of LREC 2002*, Las Palmas. ELRA. To appear.

Hamish Cunningham. 2001. *Software architecture for language engineering*. Ph.D. thesis, University of Sheffield.

Philip Dodds. 2001. ADL SCORM – Advanced Distributed Learning Sharable Content Object Reference Model. Retrieved from the WWW in February 2001: `http://www.adlnet.org/`.

D.A. Dokter and J. Nerbonne. 1997. A session with Glosser-RuG. Alfa-Informatica, University of Groningen. Retrieved from the WWW in November 1998: `http://odur.let.rug.nl/~glosser/welcome.html`.

D.A. Dokter, J. Nerbonne, L. Schurcks-Grozeva, and P. Smit. 1997. Glosser-RuG; a user study. Alfa-Informatica, University of Groningen. Retrieved from the WWW in November 1998: `http://odur.let.rug.nl/~glosser/welcome.html`.

D.A. Dokter. 1997. Glosser-RuG; Prototype December 1996. Alfa-Informatica, University of Groningen. Retrieved from the WWW in November 1998: `http://odur.let.rug.nl/~glosser/welcome.html`.

D.A. Dokter. 1998. From Glosser-RuG to Glosser-WeB. Alfa-Informatica, University of Groningen. Retrieved from the WWW in November 1998: `http://odur.let.rug.nl/~glosser/welcome.html`.

Eleni Efthimiou, editor. 2000. *LREC 2000. Second International Conference on Language Resources and Evaluation. Workshop Proceedings: Language Resources and Tools for Educational Applications*, Athens. ILSP.

European Commission. 2000. e-Learning – designing tomorrow's education. Commission of the European Communities, Communication from the Commission. COM(2000) 318 final. Brussels, 24.5.2000.

Frank Farance and Joshua Tonkel. 1999. LTSA specification. Learning Technology Systems Architecture, draft 5. Retrieved from the WWW in March 2000: `http://edutool.com/architecture/`.

Claude Frasson, Gilles Gautier, and Alan Lesgold, editors. 1996. *Intelligent Tutoring Systems. Third International Conference, ITS '96. Montréal, Canada, June 12–14, 1996. Proceedings*. Number 1086 in Lecture notes in computer science. Springer, Berlin.

Barry P. Goettl, Henry M. Halff, Carol L. Redfield, and Valerie J. Shute, editors. 1998. *Intelligent Tutoring Systems. 4th International Conference, ITS '98. San Antonio, Texas, USA, August 16–19, 1998. Proceedings*. Number 1452 in Lecture notes in computer science. Springer, Berlin.

Sylviane Granger, editor. 1998. *Learner English on Computer*. Longman, London.

Ralph Grishman, Ted Dunning, Jamie Callan, Bill Caid, Jim Cowie, Louise Guthrie, Jerry Hobbs, Paul Jacobs, Matt Mettler, Bill Ogden, Bev Schwartz, Ira Sider, and Ralph Weischedel. 1997. TIPSTER text phase II architecture design. Version 2.3.

V. Melissa Holland, Jonathan D. Kaplan, and Michelle R. Sams, editors. 1995. *Intelligent Language Tutors: Theory Shaping Technology*. Erlbaum, Mahwah, New Jersey.

Henrik Holmboe, editor. 2002. *Nordisk sprogteknologi. Nordic Language Technology*. Museum Tusculanums Forlag, Københavns Universitet, Copenhagen.

Hao-Jan Howard Chen. 1999. Creating a virtual language lab: an EFL experience at National Taiwan Ocean University. *ReCALL*, 11(2):20–30.

Nancy Ide, Patrice Bonhomme, and Laurent Romary. 2000. XCES: an XML-based encoding standard for linguistic corpora. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC2000)*, pages 825–830, Athens. ELRA.

Sake Jager, John A. Nerbonne, and A.J. van Essen, editors. 1998. *Language Teaching and Language Technology*. Swets & Zeitlinger, Lisse.

Michael Levy, editor. 1997. *Computer-Assisted Language Learning. Context and Conceptualization*. Clarendon Press, Oxford.

Monica Monachini and Nicoletta Calzolari. 1996. Synopsis and comparison of morphosyntactic phenomena encoded in lexicons and corpora. a common proposal and applications to European languages. EAGLES Document EAG-CLWG-MORPHOSYN/R.

John Nerbonne and Petra Smit. 1996. GLOSSER-RuG: In support of reading. In *COLING–96. The 16th international conference on computational linguistics. Proceedings, vol. 2*, pages 830–835, Copenhagen. Center for Sprogteknologi.

Kristina Nilsson and Lars Borin. 2002. Living off the land: The Web as a source of practice texts for learners of less prevalent languages. In *Proceedings of LREC 2002*, Las Palmas, Canary Islands, Spain. ELRA. To appear.

Mari Broman Olsen, editor. 1999. *Computer Mediated Language Assessment and Evaluation in Natural Language Processing. A joint ACL–IALL symposium*. Retrieved from the WWW in July 1999: `http://umiacs.umd.edu/~molsen/acl-iall/accepted.html`.

Fredrik Olsson. 2002. *Requirements and Design Considerations for an Open and General Architecture for Information Refinement*. Number 35 in Reports from Uppsala University, Department of Linguistics, RUUL. Uppsala University, Department of Linguistics.

Martha C. Pennington, editor. 1996. *The Power of CALL*. Athelstan, Houston, Texas.

Jack C. Richards and Gloria P. Sampson. 1974. The study of learner English. In Jack C. Richards, editor, *Error Analysis. Perspectives on Second Language Acquisition*. Longman, London.

Rafael Salaberry. 1999. Call in the year 2000: Still developing the research agenda. *Language Learning & Technology*, 3(1):104–107. http://llt.msu.edu/.

Geoffrey Sampson. 2000. Where should annotation stop? In Anne Abeille, Torsten Brants, and Hans Uszkoreit, editors, *Proceedings of the Workshop on Linguistically Interpreted Corpora. LINC-2000*, pages 29–34. Held at the Centre Universitaire, Luxembourg, August 6, 2000.

Anju Saxena and Lars Borin. 2002. Locating and reusing sundry NLP flotsam in an e-learning application. In *Proceedings of LREC 2002 workshop on Customizing Knowledge in NLP Applications: Strategies, Issues, and Evaluation*. To appear.

David Schneider and Kathleen F. McCoy. 1998. Recognizing syntactic errors in the writing of second language learners. In *COLING-ACL '98. Proceedings of the Conference, Vol. II*, pages 1198–1204, Montréal. Université de Montréal.

Mathias Schulze, Marie-Josée Hamel, and June Thompson, editors. 1999. *Language Processing in CALL*. EUROCALL/CTI Centre for Modern Languages, Hull.

Colin Smythe and Eric Shepherd. 2000. IMS question & test interoperability information model specification. version 1.01 – final specification. Retrieved from the WWW in December 2000: http://www.imsproject.org/question/qtinfo101.html.

Merryanna L. Swartz and Masoud Yazdani, editors. 1992. *Intelligent Tutoring Systems for Foreign Language Learning*. Springer, Berlin.

P. Wittenburg, D. Broeder, and B. Sloman. 2000. Metadescription for language resources. EAGLES/ISLE. a proposal for a meta description standard for language resources. Retrieved from the WWW in May 2001: http://www.mpi.nl/world/ISLE/.

Michael Zock. 1996. Computational linguistics and its use in real world: the case of computer assisted-language [sic] learning. In *COLING–96. The 16th International Conference on Computational Linguistics. Proceedings, vol. 2*, pages 1002–1004, Copenhagen. Center for Sprogteknologi.

Personal Names in Unrestricted Chinese Texts: Nature and Identification

Benjamin K. TSOU, Lawrence Y. L. Cheung

Language Information Sciences Research Centre
City University of Hong Kong
Tat Chee Avenue, Kowloon Tong,
Hong Kong

{rlbtsou, rlylc}@cityu.edu.hk

Abstract

The detection of personal names as well as proper names, and the identification of unknown words in unrestricted texts are critical tasks in NLP for East Asian languages, especially for word segmentation, information retrieval and machine translation. This is even more critical for Chinese which uses almost exclusively only the Chinese script and has little overt morphological markings and no equivalent use of capital letters for proper nouns as in English. This paper: (1) discusses the extent of the problems in some relevant IT applications, (2) analyzes the structure of Chinese personal names, and (3) presents some relevant processing strategies and the supporting language resources in general. Differences among Chinese personal names in Beijing and in Hong Kong are highlighted. It is argued that the awareness of variation in names across different Chinese communities constitutes a critical factor in enhancing the effectiveness of Chinese personal name identification algorithms.

Keywords: Chinese, personal name identification, word segmentation, Chinese IT applications, Chinese linguistic differences

1. Introduction

Personal names constitute an important linguistic symbol in conveying meaning. They are anchors of ideas, events, cultural artifacts, etc., e.g. Nobel Prize, Newtonian physics, Clinton-like behaviour and Thatcherism. Personal names provide a rich source for terminology in many domains. Efficiency in personal name identification is important for improving the detection and extraction of terms in the field of computational terminology. The last few years have seen the growth of research in this area (Miller et al., 1999; Cucchiarelli and Velardi, 2001). Named entity recognition was highlighted as an evaluation task in the Sixth and Seventh Message Understanding Conferences (MUC-6 and MUC-7) and First and Second Multilingual Entity Task (MET-1 and MET-2).

Because of the diverse and important linguistic differences between Chinese and English, personal name identification in Chinese involves many more complex issues than in English, e.g., word segmentation, absence of capital/small letter distinction, morphological paucity, syntactic ambiguity, and significant social and cultural differences among Chinese communities (Tsou and Kwong, 2001). Recent statistics from Chinese corpora provides an indicative range of personal names appearing in different domains (Tsou, 2000; Tsou, 2001). Table 1 shows that personal names account for as much as 16.8% of all word types in the 3-year LIVAC[1] newspaper corpus. They represent up to 2.4% of the word tokens in the 29 million character corpus.

	Newspaper Headlines[2]		Newspaper Texts[2]	Court Proceedings
%	Hong Kong (1 yr)	Taiwan (1 yr)	6 Chinese Communities (3 yrs)	Hong Kong (1 case)
Type	4.5	4.2	12.8 to 16.8	4.6
Token	4.4	3.7	1.6 to 2.4	0.6

Table 1 Amount of personal names in different domains

Because of the inherent linguistic problems above, the processing of Chinese personal names (as well as other named entities) in NLP requires much more than itemized listing, and poses a serious challenge.

The rest of the paper will be divided into three main sections: (1) assesses some relevant basic problems encountered in IT applications, (2) introduces the structure of Chinese personal names and the relevant processing strategies, and (3) highlights the importance of building language resources for personal name extraction.

[1] LIVAC synchronous corpus collects newspaper texts every four days since 1995 from Chinese newspapers in 6 Chinese communities including Beijing, Hong Kong, Macau, Shanghai, Singapore and Taiwan.
(LIVAC website: http://www.rcl.cityu.edu.hk/livac)

[2] The estimation is based on the 3 year data (1995—98) from the LIVAC corpus. It contains 29 million characters.

2. Significance of Personal Names in IT Applications

Efficient identification of personal names is crucial in many IT applications. Poor management of personal names in these systems can compound the errors in other NLP modules, resulting in serious deterioration of system performance. Cheung et al. (2002) conducted tests showing that poor personal name processing results in serious webpage retrieval errors.

(1) 陳中將與俄羅斯選手爭奪跆拳道金牌 (Sina) [3]
Chen Zhongjiang will compete with a Russian athlete for the gold medal for Taekwondo.

(2) 司令陳中將視導南測中心受訓部隊 (Google Chi.)[3]
Admiral Chen Zhongjiang inspected the trainee army force in Nance centre.

For example, the examined search engines mistook 中將 *zhongjiang* in (1) and (2) as the common noun for the military rank of "lieutenant general", whereas, in fact, they represent given names in the above contexts. The problem is similar to identifying *Dean Martin*, the well known American entertainer, as the head of a faculty in a university.

Tsou and Kwong (2001) also reported that the Chinese-to-English machine translation systems [4] have serious but unrecognized problems handling personal names. Table 2 shows that all four machine translation systems perform rather poorly in personal name identification. The probable cause for the errors is the use of static name list to identify personal names. The above demonstrates that IT applications need far more sophisticated algorithms than simple character matching and name database to adequately detect personal names in Chinese texts.

Data Source	Translation Accuracy			
	EWGate	*TongYi*	*Transtar*	*WorldLingo*
Hong Kong	24%	5%	9%	15%
Beijing	30%	6%	20%	56%
Taiwan	19%	0%	5%	16%

Table 2 Translation accuracy of personal names

3. Processing Chinese Personal Names: Challenges and Strategy

3.1. Challenges in Processing Chinese Personal Names

The basic structure of modern Chinese personal names is largely similar across different Chinese communities. Although the frequent length is 2 to 3 characters, the maximum can be as long as 6 characters. Table 3 shows the possible structures of Chinese personal name. Chinese personal names begin with a one- or two-character surname, followed by a one- or two-character given name. The name of a married female may be preceded by her husband's surname, as in (e) and (f). The unique structure

	Full Name	Husband's Surname			Surname			Given Name		Length
		H1	(H2)	+	S1	(S2)	+	G1	(G2)	
a.	李鵬 *Li Peng*				李 *Li*			鵬 *Peng*		2
b.	鄧小平 *Deng Xiaoping*				鄧 *Deng*			小 *Xiao*	平 *Ping*	3
c.	諸葛亮 *Zhuge Liang*				諸 *Zhu*	葛 *Ge*		亮 *Liang*		3
d.	東方聞櫻 *Dongfang Wenying*				東 *Dong*	方 *Fang*		聞 *Wen*	櫻 *Ying*	4
e.	陳方安生 *Chen Fang Ansheng*	陳 *Chen*			方 *Fang*			安 *An*	生 *Sheng*	4
f.	諸葛東方聞櫻 *Zhuge Dungfang Wenying*	諸 *Zhu*	葛 *Ge*		東 *Dong*	方 *Fang*		聞 *Wen*	櫻 *Ying*	6

Table 3 Structure of Chinese personal names

[3] *Google* [Big5 Chinese] URL: http://www.google.com/intl/zh-TW and *Sina* URL: http://www.sina.com.cn

[4] (1) Transtar V3.0, (2) TongYi '98, (3) *WorldLingo* (http://www.worldlingo.com), (4) EWGate: (http://www.EWGate.com/ewtranslite.html)

is found in speech or writing of formal register in some Chinese communities such as Hong Kong.

Apart from variable length, several characteristics make Chinese personal name processing difficult:

(a) There is no explicit morphological marking or capitalization for names in Chinese.

(b) Chinese texts do not have explicit word boundary.

(c) The character set for surnames and given names is a subset of Chinese characters for common Chinese words, and hence readily gives rise to structural ambiguity.

(d) Some personal names may be simple mono-syllabic words.

(e) Some polysyllabic words can be embedded in Chinese personal names, e.g. 王朝聞 *Wang Chaowen* (王朝 *wangchao* = dynasty), 馬勝利 *Ma Shengli* (勝利 *shengli* = victory) and 嚴肅 *Yan Su* (嚴肅 *yansu* = serious(ly)).

3.2. Basic Strategies

The complexity of Chinese personal name identification task calls for a combination of different processing strategies. They can be broadly divided into statistical approach and linguistic approach.

3.2.1. Linguistic Approach

Linguistic context provides important cues to locate Chinese personal names. Syntactic structures and lexical collocation provide good indication on whether or not the character string immediately before or after it is a potential personal name, e.g. 張志偉先生 (*Mr.* Zhang Zhiwei) and 朱鎔基總理 (*Premier* Zhu Rongji). Sun et al. (1995) integrates features to detect frequently used patterns, lexical items and syntactic structures that are useful for identifying names. For example, personal names often precede verbs like 說 *shuo* (say), 指出 *zhichu* (point out), etc. Lü et al. (2001) detect personal names by evaluating the interaction between potential personal names and neighbouring words. The POS co-occurrence restriction is checked and the best segmentation for potential name string is computed so as to generate the most probable context. Luo and Song (2001) studied the structure of personal name and place name formation. The linguistic knowledge is represented as a set of generative rules in finite state automata. Additional exceptional handling is added to deal with easily confused ambiguous contexts.

3.2.2. Statistical Approach

Statistical approach has been the most popular approach for name identification task. Previous studies typically exploited the character distribution frequency in different parts of a name and designed algorithms to extract string patterns that match the distributional criteria. For example, Sun et al. (1995) and Song and Tsou (2001) reported that about 400 characters[5] could cover over 99% of all Chinese surnames in texts. Furthermore, some character combinations in given names are more frequent than others. Cheung et al. (2002) also pointed out that there are significant variations among Chinese communities. The character preference in given names varies depending on a range of factors like gender, geography, character position in a given name, social changes, etc. The character probability is approximated by frequency distribution from large text corpora or name databases.

Sun et al. (1995) and Zheng et al. (1999) evaluated every candidate string by computing mutually exclusive probability for the 3 characters in a name candidate string, as in (6).

(6) $p_{pn}(c_1\, c_2\, c_3) = p_{sur}(c_1) * p_{m1}(c_2) * p_{m2}(c_3)$

where
$p_{pn}(s)$ = probability of candidate string s being a personal name
$p_{sur}(x)$ = probability of character x being a surname
$p_{m1}(x)$ = probability of character x being the first character of a given name
$p_{m2}(x)$ = probability of character x being the second character of a given name

Lü et al. (2001) proposed to measure probability of a potential name string by considering the probability of the 3 characters in a name candidate string as mutually inclusive events, as in (7) adapted from Lü et al. (2001).

(7) $p_{pn}(c_1\, c_2\, c_3) = p_{1F}(c_1) + p_{1M}(c_2) + p_{nE}(c_3)$

where
$p_{pn}(s)$ = probability of candidate string s being a personal name
$p_{1F}(x)$ = probability of character x being a surname
$p_{1M}(x)$ = probability of character x being the first character of a given name
$p_{nE}(x)$ = probability of character x being the second character of a given name

Most Chinese personal name identification algorithms incorporate linguistic and statistical techniques. These hybrid systems have been reported to achieve 80—90% precision and recall rates (Sun et al., 1995; Lü et al., 2001; Luo and Song, 2001).

[5] There are 21,886 characters in the GBK Chinese character set.

4. Personal Name Language Resources for Terminology Extraction

Statistical frequency data, as discussed in Section 3, has to be based on empirical data from large text corpora. Thus relevant personal name databases become a critical resource to support name identification systems and to customize algorithms. At least four major dimensions should be adequately addressed in the construction of personal name language resources, including: (1) structural distribution, (2) character frequency of personal names, (3) character co-occurrence for given names, and (4) communal differences. The significance and relevance of appropriate personal name database cannot be overemphasized because of the rarely understood magnitude of variation of personal names among Chinese communities which is much greater than that existing in English speaking communities. We will illustrate the differences in personal name patterns by using name databases taken from Beijing and Hong Kong.[6]

4.1. Structural Distribution

Single-character surnames predominate both databases, accounting for over 99%, as in Table 4. This suggests that double-character surnames may be handled separately using item listing in view of its very limited number of types and tokens. The data shows a divergence in the preference for single- and double-character given names in Beijing and in Hong Kong. Single-character names account for 29% of the Beijing database.[7] In contrast, single-character given names only cover 2% of the data for Hong Kong. The findings are crucial to the prioritization of rules related to the length of personal names in identification algorithms.

	Surname		Given Name	
%	Beijing	HK	Beijing	HK
Single-Character	99.9	99.6	29.1	2.1
Double-Character	0.1	0.4	70.9	97.9

Table 4 Distribution of name structures

[6] The Beijing name database has 125,033 names, and is drawn from a county in Beijing. They are representative of names in Mainland China because the county population is composed of migrants coming from different provinces of China. The Hong Kong database contains 11,358 names. They are student and staff names taken from the Registrar's Office, City University of Hong Kong.

[7] Sun et al. (1995) reported that single-character given names account for about 37% of the name database for all students' names (10 years) at Tsinghua University in Beijing.

4.2. Character Frequency of Personal Names

Not all characters are equally probable in being different parts of a Chinese personal name. All studies mentioned in Section 3 have exploited such characteristics to different extent. Table 5, 6 and 7 show that the ten most frequently used surnames, first character and second character of given names.

	Beijing				Hong Kong		
Rank	Surname	%	Cum. %	Rank	Surname	%	Cum. %
1	王 *Wang*	9.1	9.1	1	陳 *Chen*	10.2	10.2
2	張 *Zhang*	8.3	17.4	2	黃 *Wang*	6.7	16.9
3	李 *Li*	7.9	25.3	3	李 *Li*	5.9	22.8
4	劉 *Liu*	6.5	31.8	4	梁 *Liang*	4.6	27.4
5	陳 *Chen*	3.2	35.0	5	林 *Lin*	4.2	31.6
6	趙 *Zhao*	3.2	38.2	6	張 *Zhang*	3.6	35.2
7	楊 *Yang*	3.0	41.2	7	劉 *Liu*	3.0	38.2
8	孫 *Sun*	2.0	43.2	8	吳 *Wu*	3.0	41.2
9	馬 *Ma*	1.7	44.9	9	何 *He*	2.8	44.0
10	吳 *Wu*	1.6	46.5	10	鄭 *Zheng*	2.1	46.1

Table 5 10 most frequent single-character surnames in Beijing and Hong Kong

(Shaded items appear in both columns.)

	Beijing				Hong Kong		
Rank	G1	%	Cum. %	Rank	G1	%	Cum. %
1	淑 *shu*	3.2	3.2	1	嘉 *jia*	3.8	3.8
2	玉 *yu*	3.1	6.3	2	偉 *wei*	3.7	7.5
3	秀 *xiu*	2.9	9.1	3	志 *zhi*	3.5	11.0
4	曉 *xiao*	2.6	11.7	4	家 *jia*	2.8	13.8
5	文 *wen*	2.3	14.0	5	詠 *yong*	2.2	16.0
6	建 *jian*	2.2	16.2	6	慧 *hui*	2.1	18.1
7	志 *zhi*	1.9	18.0	7	國 *guo*	2.0	20.1
8	小 *xiao*	1.8	19.8	8	文 *wen*	2.0	22.0
9	桂 *gui*	1.7	21.5	9	佩 *pei*	1.9	24.0
10	春 *chun*	1.4	22.8	10	麗 *li*	1.9	25.9

Table 6 10 most frequent first characters (G1) of double-character given names

(Shaded items appear in both columns.)

	Beijing				Hong Kong		
Rank	G2	%	Cum. %	Rank	G2	%	Cum. %
1	華 *hua*	3.6	3.6	1	儀 *yi*	3.1	3.1
2	英 *ying*	3.4	7.0	2	華 *hua*	2.3	5.4
3	蘭 *lan*	2.1	9.1	3	明 *ming*	2.2	7.6
4	平 *ping*	1.9	11.0	4	敏 *min*	2.2	9.8
5	珍 *zhen*	1.8	12.8	5	文 *wen*	2.1	11.9
6	明 *ming*	1.7	14.5	6	玲 *ling*	1.9	13.7
7	榮 *rong*	1.6	16.1	7	珊 *shan*	1.7	15.4
8	生 *sheng*	1.5	17.6	8	欣 *xin*	1.6	17.0
9	芳 *fang*	1.3	18.9	9	輝 *hui*	1.6	18.6
10	琴 *qin*	1.3	20.1	10	雯 *wen*	1.6	20.1

Table 7 10 most frequent second characters (G2) of double-character given names

(Shaded items appear in both columns.)

The character type for Chinese surnames is fairly limited in actual data. In Table 5, the ten most frequent surnames cover over 46% of the name tokens though the ranking of surnames is quite different in both databases. For example, the most frequent surname 王 *Wang* in Beijing is ranked as 14th in the Hong Kong. In contrast, the character types for given names are far more diverse. In the Hong Kong database, there are over 820 character types for given names as opposed to 257 character types for surnames. As shown in Table 6 and 7, the ten most frequently used G1 and G2 character cover no more than 26% of all name tokens in both databases respectively.

4.3. Character Co-occurrence for Given Names

Apart from localized character preference in given names, our data also reveals that the character combinations of double-character given names are far from being random. Previous research tended to consider the probabilities of each character position in isolation, and ignored interesting patterns of character co-occurrence in given names. The information is useful for resolving ambiguity given rise by the diverse character types in given names. Here are two examples for the two most common G1 characters from Hong Kong database: 嘉 *jia* and 偉 *wei*. Given G1 = 嘉 *jia* / 偉 *wei*, there is about 30% of chance that the given name is one of the combinations in a—e and f—j respectively. (Table 8)

	Combina-tion	%	Cum. %		Combina-tion	%	Cum. %
a	嘉 + 敏 *jia + min*	11.2	11.2	f	偉 + 強 *wei + qiang*	6.4	6.4
b	嘉 + 儀 *jia + yi*	6.5	17.7	g	偉 + 文 *wei + wen*	5.7	12.1
c	嘉 + 雯 *jia + min*	4.6	22.3	h	偉 + 雄 *wei + xiong*	5.7	17.7
d	嘉 + 琦 *jia + qi*	4.3	26.6	i	偉 + 傑 *wei + jie*	*5.4*	23.2
e	嘉 + 慧 *jia + wei*	3.6	30.1	j	偉 + 明 *wei + ming*	5.2	28.3

Table 8 *5* most frequent combinations provided G1 = 嘉 *jia* / 偉 *wei*

4.4. Communal Differences

Previous studies do not seem to pay much attention to the sociolinguistic aspects of name variation among Chinese communities. It has been mentioned in Section 4.1 that there are far more single-character given names in the Beijing database. If we further compare the columns for Beijing and Hong Kong in Table 6 and 7, only two characters overlap. The divergence in character preference is obvious in the two databases. The implication is that name identification algorithms using statistical approach should maintain character probability derived from various Chinese communities in order to maximize the performance. Other sociolinguistic differences such as married female's names, nicknames, etc, have yet to be studied. The assumption that personal name identification can be simplistically tackled on the basis of personal name language resources from a single community will certainly be problematical for NLP applications that have to process unrestricted texts from different geographical locations.

5. Further Works

Based on Section 4.2 and 4.3, further investigation into the statistical distribution of personal names can be done. First, it seems that previous studies tended to have overlooked character co-occurrence phenomenon. Co-occurrence probability can be used to improve existing algorithms for Chinese personal name tagger. For example, instead of merely utilizing the probability of a candidate character being part of a name, the tagger may give a higher rating to those candidate strings whose G1 and G2 combination is commonly found in Chinese names. Statistical studies like those in Section 4.3 will be conducted for all character combination in the two databases to identify high frequency patterns.

Second, as we mentioned earlier and noted by a reviewer, gender is a significant factor in character choice for given names. Such data may find applications in transcription system and speech recognition application such as caller name identification. The recognition engine may first determine the caller's gender based on the speaker's voice pitch and then select the appropriate probability database for name identification task accordingly.

Third, the communal differences revealed by our preliminary analysis suggest that name databases from other Chinese communities are important language resources. For example, more name databases will be collected (e.g. Shanghai, Taiwan and Singapore) for comparison.

6. Conclusion

Personal names provide an important source for new terms in text processing. Personal name identification is crucial to terminology extraction. This paper discusses the challenge and basic strategies in personal name identification in unrestricted Chinese texts. The review of IT applications shows that reliability in the processing of Chinese personal names is still far from acceptable. This situation contributes to serious errors in other NLP tasks such as incorrect data retrieval and parsing. Current Chinese personal name identification systems capitalize on linguistic and statistical techniques to deal with the processing. To adequately support such systems, personal name language resources are critical. Four dimensions have been highlighted in the construction of such resources, including (1) structural distribution, (2) character frequency of personal names, (3) character co-occurrence for given names, and (4) communal differences. Despite the potential contribution to the identification task, the latter two dimensions seem to have gone largely unnoticed in the literature. More empirical study of personal names will be beneficial to the performance improvement of personal name identification systems.

7. Acknowledgement

This research study is supported by the Language Information Sciences Research Centre, City University of Hong Kong and by a Competitive Earmarked Research Grant (CityU 1238/00H) from the Research Grant Council of Hong Kong and supported by NTT Service Integration Laboratory. We would also like to thank Rou SONG for his kindness in providing us with the Beijing personal name database, and Registrar's Office, City University of Hong Kong, for their contribution to our Hong Kong personal name database.

8. References

Cheung, L., B. K. Tsou and M. Sun. (2002) Identification of Chinese Personal Names in Unrestricted Texts. *Proceedings of the 16th Pacific Asia Conference on Language, Information and Computation*, Cheju, Korea, pp. 28—35.

Cucchiarelli, A. and P. Velardi. (2001) Unsupervised Named Entity Recognition Using Syntactic and Semantic Contextual Evidence. *Computational Linguistics* 27 (1): 123—131.

Luo, Z. and R. Song. (2001) Integrated and Fast Recognition of Proper Noun in Modern Chinese Word Segmentation. [in Chinese] *Proceedings of International Conference on Chinese Computing 2001*, Singapore. pp. 323—328.

Lü, Y., T. Zhao, M. Yang, H. Yu and S. Li. (2001) Leveled Unknown Chinese Words Resolution by Dynamic Programming. [in Chinese] *Journal of Chinese Information Processing*, 15 (1), Beijing, China.

Miller, D., R. Schwartz, R. Weischedel, and R. Stone. (1999) Named entity extraction from broadcast news." *Proceedings of DARPA Broadcast News Workshop*. Herndon, VA.

Song, R. and B. K. Tsou. (2001) Preliminary Study on Chinese Proper Noun. [in Chinese] *Proceedings of the 20th Anniversary Conference of the Chinese Information Processing Society of China*. November 2001. pp. 14—19.

Sun M., C. Huang, H. Gao and J. Fang. (1995) Identifying Chinese Names in Unrestricted Texts. [in Chinese] *Journal of Chinese Information Processing*, 9 (2), Beijing, China.

Tsou, B. K. (2000) Lexical Variation in Chinese: The Windows Approach. (Invited paper) Annual Research Forum, Linguistic Society of Hong Kong. December 2000.

Tsou, B. K. (2001) Corpus, Information Mining and the New Global Village. (Keynote speech) *Proceedings of 6th Natural Processing Pacific Rim Symposium*, Tokyo, November 2001. pp. 9—18.

Tsou, B.K. and Kwong, O.Y. (2001) Evaluating Chinese-English Translation Systems for Personal Name Coverage. *Proceedings of the MT Summit VIII Workshop on MT 2010 -- Towards a Road Map for MT*, Santiago de Compostela, Spain.

Zheng, J., X. Lin and H. Tan. (2000) The Research Chinese Names Recognition Method Based on Corpus. [in Chinese] *Journal of Chinese Information Processing*, 14 (1), Beijing, China.

Changes in the Etymological Type of New Terminology in Japanese
-The Decrease of Sino-Japanese and Increase of Alphabetical Terms-

SHIODA, Takehiro

NHK Broadcasting Culture Research Institute
Mori Tower 16F, 2-5-1
Tokyo 105-6216, Japan
sioda@culture.nhk.or.jp

Abstract

In this paper, the author surveys how the proportions of etymological type have changed in current computer-related terms of Japanese. As a result of inquiry regarding recent computer terminology, the fact that the decreasing of Sino-Japanese words and the increasing of Alphabetical words has confirmed.

Introduction

Japanese words are conventionally divided into three etymological types, known as *goshu* in Japanese, according to whether they are of pure Japanese, Sino-Japanese or Western-loans. In this paper, the author surveys how the proportions of each type have changed in current computer-related terms, and considers what the future may hold for Japanese technical terms.

First, some technical terms required for the analysis are explained, and previous studies introduced. The materials and analytical method are then described, and the results reported. The results of opinion polls on people's thoughts regarding the use of foreign words are also introduced, and a proposal on word coining is presented.

1 "GOSHU " in Japanese Linguistics

The term *goshu* refers to a basic convention used in classifying the parts of the Japanese vocabulary. It is the taxonomical concept for defining words according to their etymological source. The three basic types that are taken to constitute the Japanese vocabulary are the words of pure Japanese, Sino-Japanese and Western-loan.

The pure Japanese words, *wago*, are the words of traditional Japanese origin. These are frequently found in terms that express fundamental concepts in Japanese. They are written in *hiragana* syllabary or *kanji* (Chinese characters) in general.

The Sino-Japanese words, *kango*, can primarily be described as words that are borrowed from Chinese. However, the *kango* are read in a Japanese, not a Chinese way, despite the use of the Chinese characters. (This is similar to the many different pronunciations of the word *euro*, which varies so much from language to language.)
There is also the concept of *wasei-kango*, namely Sino-Japanese words created in Japan, as a subdivision of *kango*. These are unique Japanese coinages that use Chinese morphemes.
The traditional scientific terms include many Sino-Japanese words. It is usual to use Chinese characters when writing these Sino-Japanese terms. It has been observed that weight of the Sino-Japanese words in Japanese language is similar to that of words of Latin origin in English (Miyajima, 1995).

The Western-loans, called *gairaigo*, are mostly loan words from Western languages (mainly English), and sometimes the words of not-western origin are also included exceptionally. The newest terms include many words of western origin. It is usual to use *katakana* syllabary when writing these Western-loans, but the alphabet is also used in some cases.

These three types compose the fundamental taxonomy of Japanese etymological word types.

In addition, some new words are formed by combining the different types. These hybrid words are called *konshugo*.

2 Previous research

It has been shown in quantitative terms that the use of *kango*, Sino-Japanese words, was chiefly utilized in new coinages around 1900, and that ratio gradually decreased thereafter (Miyajima, 1967). This tendency has continued in recent years and the word-formation capability of *kango* fell sharply in the very short period from 1960 to 1980 (Nomura, 1984). As for writing means, it has been predicted that the use of Chinese characters will decrease and that of the alphabet will increase from now on (Kabashima, 1981).

3 Purpose of inquiry

In order to predict future transitions in Japanese terminology, the present situation was gauged with reference to the following points:

1. It has been observed that the word-formation capability of *kango* has been decreasing. What is the rate of this decrease?

2. It is known that the proportion of Western-loans is increasing in Japanese. The author believes that the increase may be greatest for alphabetical words. Can this be demonstrated quantitatively?

In this paper, the author reports the results obtained regarding computer terminology.

4 Procedure of inquiry

Subject of inquiry:
"Gendai Yoogono Kiso Chisiki (Basic knowledge of contemporary words)" 1985, 1990, 1995, 2000: Tokyo, Jiyuu Kokuminsha.

This book is a single volume encyclopedia published annually. It provides rich data for considering the status and progress of new words from year to year. For this study, the entries related to the computer field (computer terms, office automation terms, etc.) were extracted.
Each entry was classified according to the *goshu* category.

1. We observe the transitions of Sino-Japanese and Western-loans in the first, *goshu* classification phase. Since there are very few pure Japanese words, these are disregarded here. "*Katakana* words" and "alphabetical words" are provided as sub-classifications of Western-loans, and the transitions for each are noted. Here, we only observe the number of entries belonging to single *goshu* categories. *Konshugo* are taken up in the second phase below.

2. Next, consideration is also given to the hybrid words, *konshugo*. (This can be described as classification by *goshu* element). Since there are also very few elements of pure Japanese words here, these are again disregarded.

A hybrid word consisting, for example, of one Sino-Japanese and one Western-loan is counted once in each of the Sino-Japanese element and Western-loan element categories. For convenience, however, a term consisting of multiple Sino-Japanese elements is counted only once in the Sino-Japanese element category.

Examples:
「情報検索」(information retrieval):
Scores 1 for the Sino-Japanese element
「エレクトロニック・バンキング」
(electronic banking):
Scores 1 for the *Katakana* word element
「磁気ディスク装置」 (magnetic disk unit):
Scores 1 for the Sino-Japanese element
Scores 1 for the *Katakana* word element
「ＯＣＲ」(Optical Character Recognition):
Scores 1 for the alphabetical word element
「双方向ＣＡＴＶ」(two-way CATV):
Scores 1 for the Sino-Japanese element
Scores 1 for the alphabetical word element

5 Results and discussion

5.1 1st phase (classification by *goshu*)

	1985	*1990*	*1995*	*2000*
Total	398	344	357	402
Sino-Japanese	54(13.6%)	46(13.4)	51(14.3)	37(9.2)
Katakana	136(34.1)	108(31.4)	114(31.9)	118(29.4)
Alphabetical	19(4.8)	30(8.7)	27(7.6)	66(16.4)

(*see* Figure 1)
→ The rates for Sino-Japanese words and alphabetical words were substantially reversed from 1995 to 2000.

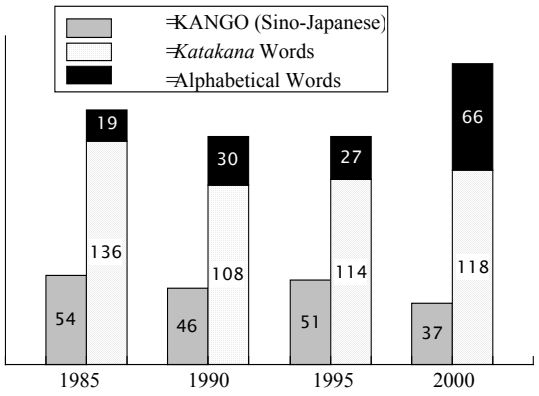


Figure 1: The transition by GOSHU in computer te

5.2 2nd phase (classification by *goshu* element)

	1985	*1990*	*1995*	*2000*
Total	398	344	357	402
Sino-Japanese	218(54.8)	186(54.1)	193(54.1)	155 (38.6)
Katakana	305(76.6)	239(69.5)	251(70.3)	270(67.2)
Alphabetical	66(16.6)	86(25.0)	79(22.1)	158 (39.3)

(*see* Figure 2)

→ The rates for Sino-Japanese and alphabetical elements drew much closer to each other in the data for 1995 to 2000.

Prospect: The likelihood of a further increase in the rate of use of alphabetical words appears to be quite strong.

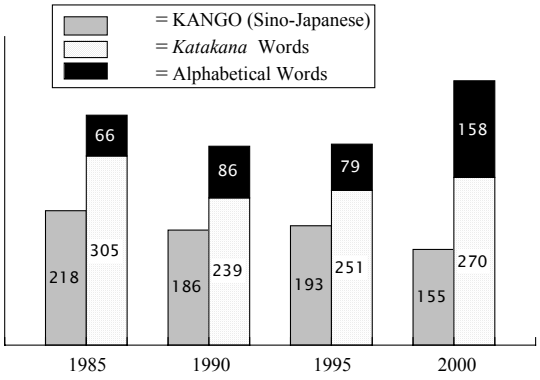


Figure 2: The transition by GOSHU element

6 Views on Western-loans

The excessive use of words of foreign origin can hinder communication. We next introduce some results on this subject from public opinion polls.

"Do you feel that many loan words and other foreign words are used in everyday Japanese?"
Frequent: 51.6%
Occasional: 32.2%
(Agency for Cultural Affairs, 2000)

"Have you been troubled because you cannot understand the meaning of a *katakana* word in newspaper or TV?"
Frequently : 17.1%
Occasionally : 37.5%
(Agency for Cultural Affairs, 1997)

The entry of new foreign terms cannot be prevented. But, as these surveys indicate, we should be aware of the dangers of excess.

7 Concluding remarks

It has been observed that one of the merits of the increase in foreign words is the acceptance of terms that are understood internationally (Ishiwata, 2001). Alphabetical words, in particular, can be read and understood by those who cannot read Japanese script, so the level of international communicability is very high. The risk is that more fluent international communication may be matched by weaker internal communication. The use of such words as technical terms has clear merits, but thought is also required to the selection of words that are best able to acquire general acceptability within the specific language-speaking group concerned. We should remember that *not* all the people understand English.

Some technical terms do gradually come to be used as general terms in each language. Those who coin new terms or standardize the terminology would, therefore, be well advised to consider their suitability for both international and internal communication purposes, with the awareness that these decisions may have some future influence on general terms kept clearly in mind.

References

Agency for Cultural Affairs (Bunkachoo). (1997, 2000). *Kokugoni kansuru yoron choosa (Census on the Japanese Language)*. Tokyo: Ookurashoo insatsukyoku.

Inoue, Fumio. (2001). English as a Language of Science in Japan. From Corpus Planning to Status Planning. *The Dominance of English as a Language of Science -Effects on Other Languages and Language Communities*. Berlin/New York: Mouton de Gruyter.

Ishiwata, Toshio. (2001). *Gairaigo no soogooteki kenkyuu (Comprehensive Study on Western-origin borrowed vocabulary)*. Tokyo: Tookyoodoo shuppan.

Kabashima, Tadao. (1981). *Nihongo wa doo kawaruka (How does Japanese change?)*. Tokyo: Iwanami Shoten.

Miyajima, Tatsuo. (1967). Kindai-goi no keisei (Formation of the modern vocabulary). *Kokuritsu kokugo kenkyuujo ronshuu (Collected Papers of The National Language Research Institute) 3*. Tokyo: Shuuei Shuppan.

Miyajima, Tatsuo. (1995). A Contrastive Study of Vocabulary Growth in Different Languages - French, English, Chinese, and Japanese. *Lexical Knowledge in the Organization of Language*. Amsterdam/ Philadelphia: John Benjamins Publishing Company.

Nomura, Masaaki. (1984). Goshu to zoogoryoku (The etymological type and the capability of word-formation). *Nihongogaku 3-9*, 1984.9. Tokyo: Meiji Shoin.

Shioda, Takehiro. (2000). Japanese and Korean Terminologies Reviewed from a Linguistic Perspective. *Proceedings of Workshop on Terminology Resources and Computation*, Held in conjunction with the LREC2000. Athens.

Shioda, Takehiro. (2002). Senmon-yoogo ni okeru arufabetto-go no zooka (The Increase of Alphabetical Words in Japanese Terminology). *Journal of Japan Society of Information and Knowledge,* Vol.12, No.1. Tokyo: Japan Society of Information and Knowledge.

A Corpus-based Approach to Term Bank Construction

Bai Xiaojing, Hu Junfeng, Zan Hongying, Chen Yuzhong, Yu Shiwen

Institute of Computational Linguistics, Peking University, China

E-mail: {baixj, hujf, zanhy}@pku.edu.cn

Abstract

In this paper, a corpus-base approach is presented in the construction of the information science and technology term bank in which domain classification, reference and part of the definition are extracted from corpus. Farther experiments show that the structure analysis of the terms can be helpful in the corpus-based domain classification of the terms.

1. Introduction

Currently, a joint project is under way between China National Institute of Standardization(CNIS) and the Institute of Computational Linguistics(ICL), Peking University to construct a term bank in the field of information science and technology. The project aims at :

1. an ontology system
2. a corpus for term bank construction
3. a corpus-based terminology extraction program
4. a constructed term bank and the related specifications and standards, and others for terminologies in the field of information science and technology

The implementation of the whole project features various approaches, among which the corpus-based one constitutes our present focus.

The corpus in this project consists of two parts, an essential corpus of 15 million Chinese characters and an extension corpus of 60 million and more, responsible for different tasks respectively. The corpus-based approach enables us to address the goals of our project by the following schemes:

1. Categorization of the terminologies in our term bank
2. Assistance for defining the terminologies in our term bank
3. Training and testing of the automatic extraction program

Now, initial plans have been made for the implementation of these schemes, with experiments conducted in support of our further efforts.

2. The Classification Scheme of Information science and Technology

An ontology system is very important for the standardization of the term bank establishment. Up to now, there still do not have a ready-made classification scheme of information science and technology, not to say to put each specific terminology into one specific domain category. So the first thing for constructing the term bank in the field of information science and technology is to build an appropriate knowledge category system or concept system.

The information science and technology field contains not only the computer and communication subjects. In general, this field includes all subjects relative to information. Now there is no acknowledged opinion that bounds this field. We intend to set up an appropriate and practical classification while make it integrated with the some existed international or national standards. We have referred to the ACM Computing Classification System, ICS(the International Classification for Standards), CLC(the Chinese Library Classification), computer encyclopedias, and some technical dictionaries. After we have consulted many materials, we classify the

knowledge of information science and technology field into five subjects:

1. pandects of information science and technology
2. computer
3. automatization
4. telecommunication
5. electronics

under each subject we provide four subclass: theory, technology, application and product & material. We also have set up a mapping between ICS and our classification. For example, ICS:35:220 are integrate into our classification in data storage device(its classification number is 020403).

Generally, our classification is on the second level of subjects, and some detail on the third or fourth level. Frankly, Our knowledge classification system has fewer hierarchical levels. The reason is that we plan to get a more general and shallow classification and to avoid the frequent modification of the structure of the term bank due to the slight change of term category. The change of terms' intension and extension will be reflected through some attributes in our term bank. The attributes in the term bank are very easily modified or expanded.

3. Corpus Compilation

For the essential corpus, we turn to experts in the field of information science and technology. All the texts are chosen and provided by experts of specified branches.

In the meaning time, with the help of a program, field experts will tag all the terms and the related information in the corpus, i.e., categorize them into the very branches of the field they belong to. The essential corpus is built for data training in the automatic extraction program.

For the extension corpus, the size is more than 60 million Chinese characters. In this corpus, we can get concordance and collocation information about the terms, as automatic processing will be possible for this part, and further, considerable amount of useful information, which can facilitate the definition of the terms, can be extracted from the corpus. Moreover, this corpus will serve as a test set for the terminology extraction program.

4. Corpus-based Categorization of Terminologies

Up till now, a basic framework has been drafted out for the purpose of categorization, while the terminologies available now are more than 70,000. Given the possibility that the initial framework can be developed to a sound system for categorization, locating the Terms into this system will still be a hard job.

It is in this consideration that we come up with the corpus-based approach. The essential corpus provided by various field experts carries the field information and the terminology tagging. Terminologies tagged by field experts are to be compared with the Terms. This is designed to be a process of matching, after which the Terms can be put into their respective categories. In other words, we try to classify the terms according to their distribution in the corpus. For the first step, as a test, we obtained 100 texts (258,045 characters in total) about Computer Network, with 2,486 different terms tagged out (i.e., 2,486 terminologies are regarded as valid). Considerable terms, which are unlikely network ones, proved otherwise in the corpus.

For example:缓冲/cache, which does not seem to be an OS term in Chinese, is a true network concept in the following sentence: "与我们熟悉的磁盘缓冲技术类似，Internet 缓冲是在一台本地服务器上开辟一块缓冲区，保存访问 Internet 时获得的数据，这样在以后的浏览过程中如果还是访问那些网页，就不需要再次访问Internet ，而直接从缓冲中获得数据就可以了".

That means corpus based categorization can give a more accurate description of the field information about the terms. This will benefit not only the term categorization, but also the definition of the terms. In some cases, it can

even give us clues to find out terms with different shades of meaning.

5. Corpus-based Reference for Terminology Definition

Accuracy and standardization in defining terminologies also attract our attention and efforts. In the database of our term bank, there is a field named Reference, storing contexts of the Terms from the whole corpus, which are deemed as competent reference. Reference for terminology definition can be at various levels, namely, it can be sentence(s), paragraph(s) or even full text(s). Here the role of the corpus is significant, as it contains all the information that will be filled into the Reference field, and what is more, we are expecting templates for terminology reference or even for terminology definition, to be learned from the essential corpus and then applied to the extension part, thus achieving the corpus-based automatic referencing. In addition to category and terminology tagging, our field experts also have to tag the text contents that they regard as the competent references for terminologies. A program is designed to extract a language unit bearing a reference tag (starting with <Reference> and ending with </Reference>) containing or following a terminology tag (starting with <Term> and ending with </Term>), which is recognized as the reference information for the tagged terminology and will then be stored in the Reference field accordingly. The following are three examples.

Example 1: (a single sentence)

<Reference><Term>Vo I P **</Term>**可以定义为以IP 包交换的方式传输话音。**</Reference>**

Example 2: (a paragraph)

<Reference><Term>Vo I P 网关**</Term>**主要提供PSTN 电话通信网络与IP 网络的接口和转换。目前，一般采用H.323 作为IP 网络信令和SS7 作为PSTN 的信令。在这个市场的设备提供商中既有传统的数据网络公司如3Com 、Cisco 等，也有老牌的电信设备提供商如Alcatel 、Ericsson 、Nortel 、Lucent 等，以及Sonus 、Clarent 、convergent network 、Nuera 等公司。**</Reference>**

Example 3: (a full text)

<Reference>何谓**<Term>**DHC**P</Term>**？动态主机配置协议（Dynamic Host Configuration Protocol，DHCP）从原有的BootP 协议发展而来，原来的目的是为无盘工作站分配IP 地址的协议，当前更多地用于对多个客户计算机集中分配IP 地址以及IP 地址相关的信息的协议，这样就能将IP 地址和TCP/IP 的设置统一管理起来，而避免不必要的地址冲突的问题，因此常常用在网络中对众多DOS/Windows 计算机的管理方面，节省了网络管理员手工设置和分配地址的麻烦。中继代理服务器必须知道DHCP 服务器的地址，还要知道如何把接收到的报文转发给该服务器**</Reference>**

Sufficient data will avail us of the opportunity to learn reference templates, like “XX 可以定义为/can be defined as XX” in Example1; “XX 主要提供/is mainly for XX” in Example 2 etc. These are sample templates that can be used to extract the definition of the terms from corpus. Surely there can only have small number of the terms that can find definition directly from corpus, but the corpus-based contextual information, such as concordance and collocation are also helpful for experts to analysis the meaning and give the proper definition of the terms.

6. Automatic Extraction of Terminologies from Corpus

The third scheme is based on the understanding that the internal structure of terminologies is also a source of valuable knowledge for term bank construction. In this project, the internal structure of a terminology consists of three elements: 1) term constituents, including prefixes, suffixes, words and phrases that are frequently used in related technical documents, e.g., “性” and “接口”; 2) POS; and 3) semantic categories, each describing the common feature of a group of term constituents, like

the semantic category “equipped with/without a system of wires” derived from “无线” and “有线 ”. Patterning the internal structure of terminologies is a prerequisite to the automatic extraction of terminologies from the corpus. On the one hand, we analyze the Terms, together with those from the essential corpus and tagged by our field experts, and pattern their structures, using term constituents and POS information, e.g., “noun + 接口”. On the other hand, we generate new terms, replacing term constituents of the same categories in exiting terms with the other.

With “有线通讯”, “有线电视”, “有线电报”, for instance, we generate “无线通讯”,“无线电视”, “无线电报”. The automatic extraction program will then use the structure patterns and the new terms generated to extract terminologies from the extension corpus, either by character matching or by POS matching, or both. Large in amount as they are, the terminologies we have obtained r reach up till now. In this sense, the extension corpus is both a test set for the automatic extraction program and a source for additional terminologies by using the program. It therefore are still far from being enough. Considering the limited sources, we have to rely on the extension corpus for the automatic extraction of terminologies that remain out of oucalls our attention to the competence and performance of our corpus, and especially, the extension part.

7. Conclusion

We have devised the initial schemes for the application of the corpus-based approach to

1. the categorization of existing terminologies in our term bank
2. the learning of reference templates and the extraction of reference information from the corpus
3. the modeling of automatic terminology extraction

Experiments show that corpus can be very useful to illuminate the meaning of terms, which will help a lot to standardize the terms in the future.

References

1. Angelo, Robert. *A Synopsis of Wittgenstein’s Logic of Languag*e. http://www.roangelo.net/logwitt.
2. Feng, Zhiwei, , (1997). *An Introduction to Modern Terminolog*y. Yuwen Press
3. Sinclair John, (1991). *Corpus, Concordance, Collocation.* Oxford University Press.
4. Kennedy Graeme, (2000). *An Introduction to Corpus Linguistic*s. Foreign Language Teaching and Research Press
5. http://www.acm.org/class/1998/
6.http://www.iso.ch/iso/en/CatalogueListPage.CatalogueList
7. Chinese Library Classification, Version4.0, Beijing library press, China
8. Zan Hongying, Hu Junfeng, et al (2002) , Construction of the Term Bank, TAHK2002

Standards for Language Resources

Nancy Ide,* Laurent Romary†

* Department of Computer Science
Vassar College
Poughkeepsie, New York 12604-0520 USA
ide@cs.vassar.edu
† Equipe Langue et Dialogue
LORIA/INRIA
Vandoeuvre-lès Nancy, FRANCE
romary@loria.fr

Abstract

This paper presents an abstract data model for linguistic annotations and its implementation using XML, RDF and related standards; and to outline the work of a newly formed committee of the International Standards Organization (ISO), ISO/TC 37/SC 4 Language Resource Management, which will use this work as its starting point. The primary motive for presenting the latter is to solicit the participation of members of the research community to contribute to the work of the committee.

1. Introduction

The goal of this paper is two-fold: to present an abstract data model for linguistic annotations and its implementation using XML, RDF and related standards; and to outline the work of a newly formed committee of the International Standards Organization (ISO), ISO/TC 37/SC 4 Language Resource Management, which will use this work as its starting point. The primary motive for presenting the latter is to solicit the participation of members of the research community to contribute to the work of the committee.

The objective of ISO/TC 37/SC 4 is to prepare international standards and guidelines for effective language resource management in applications in the multilingual information society. To this end, the committee will develop principles and methods for creating, coding, processing and managing language resources, such as written corpora, lexical corpora, speech corpora, dictionary compiling and classification schemes. The focus of the work is on data modeling, markup, data exchange and the evaluation of language resources other than terminologies (which have already been treated in ISO/TC 37). The worldwide use of ISO/TC 37/SC 4 standards should improve information management within industrial, technical and scientific environments, and increase efficiency in computer-supported language communication.

2. Motivation

The standardization of principles and methods for the collection, processing and presentation of language resources requires a distinct type of activity. Basic standards must be produced with wide-ranging applications in view. In the area of language resources, these standards should provide various technical committees of ISO, IEC and other standardizing bodies with the groundwork for building more precise standards for language resource management.

The need for harmonization of representation formats for different kinds of linguistic information is critical, as resources and information are more and more frequently merged, compared, or otherwise utilized in common systems. This is perhaps most obvious for processing multi-modal information, which must support the fusion of multimodal inputs and represent the combined and integrated contributions of different types of input (e.g., a spoken utterance combined with gesture and facial expression), and enable multimodal output (see, for example, Bunt and Romary, 2002). However, language processing applications of any kind require the integration of varieties of linguistic information, which, in today's environment, come from potentially diverse sources. We can therefore expect use and integration of, for example, syntactic, morphological, discourse, etc. information for multiple languages, as well as information structures like domain models and ontologies.

We are aware that standardization is a difficult business, and that many members of the targeted communities are skeptical about imposing any sort of standards at all. There are two major arguments against the idea of standardization for language resources. First, the diversity of theoretical approaches to, in particular, the annotation of various linguistic phenomena suggests that standardization is at least impractical, if not impossible. Second, it is feared that vast amounts of existing data and processing software, which may have taken years of effort and considerable funding to develop, will be rendered obsolete by the acceptance of new standards by the community. To answer both of these concerns, we stress that the efforts of the committee are geared toward defining *abstract models* and *general frameworks* for creation and representation of language resources, rather than specific formats. These models should, in principle, be abstract enough to accommodate diverse theoretical approaches. The model so far developed in ISO TC/37 for terminology, which has informed and been informed by work on representation schemes for dictionaries and other lexical data (Ide, *et al.*, 2000) and syntactic annotation (Ide & Romary, 2001) demonstrates that this is not an unrealizable goal. Also, by situating all of the standards development squarely in the framework of XML and related standards such as RDF, DAML+OIL, etc., we hope to ensure not only that the standards developed by the committee provide for compatibility with established and widely accepted web-based technologies, but also that

transduction from legacy formats into XML formats conformant to the new standards is feasible.

ISO/TC 37/SC 4 will liaison with ISLE (International Standards for Language Engineering), which has implemented various recent efforts to integrate EC and US efforts for language resources. Where possible, these and other standards set up in EAGLES will be incorporated into the ISO standards. ISO/TC 37/SC 4 will also broaden the work of EAGLES/ISLE by including languages (e.g. Asian languages) that are not currently covered by EAGLES/ISLE standards.

At present, language professionals and standardization experts are not sufficiently aware of the standardization efforts being undertaken by ISO/TC 37/SC 4. Promoting awareness of future activities and rising problems, therefore, will be a crucial factor in the success of the committee, and will be required to ensure widespread adoption of the standards it develops. An even more critical factor for the success of the committee's work is to involve, from the outset, as many and as broad a range of potential users of the standards as possible. This presentation serves as a call for participation to the linguistics and computational linguistics research communities.

3. Objectives

ISO TC37/SC 4's goal is to develop a platform for the design and implementation of linguistic resource formats and processes in order to facilitate the exchange of information between language processing modules. This will be accomplished by defining a *common interface format* capable of representing multiple kinds of linguistic information. The interface format must support the communication among all modules in the system, and be adequate for representing not only the end result of interpretation, but also intermediate results.

A well-defined representational framework for linguistic information will also provide for the specification and comparison of existing application-specific representations and the definition of new ones, while ensuring a level of interoperability between them.

3.1. Requirements

Very generally, a linguistic representation framework must meet the following requirements:

- *Expressive adequacy:* the framework should be expressive enough to represent all varieties of linguistic information;
- *Semantic adequacy:* the representation structures should have a formal semantics, i.e., their definition should provide a rigorous basis for further processing (e.g., deductive reasoning, statistical analysis, generation, etc.).

Providing interface formats within a system architecture demands that "incremental" construction of intermediate and partial representations be supported. In addition, if the construction of a final representation does not succeed, the representation must capture the information required to enable appropriate system action. This dictates additional requirements:

- *Incrementality:* support for the various stages of input interpretation and output generation, allowing both early and late fusion and fission.
- *Uniformity:* the representation of various types of input and output should utilize the same "building blocks" and the same methods for combining complex structures composed of these building blocks.
- *Underspecification and partiality*: support for the representation of partial and intermediate results, including the capture of unresolved ambiguities.

Finally, the representational framework must be accommodate the developing field of language processing system design by satisfying these further requirements:

- *Openness:* the framework should not depend on a single linguistic theory, but should enable representations based on different theories and approaches;
- *Extensibilty*. The framework should be compatible with alternative methods for designing representation schemas (e.g., XML) rather than being tied to a specific schema.

3.2. Methodology

A working group of SC 4 (WG1/WI-1) has been charged with the task of defining a linguistic annotation framework, which will be used by other SC 4 working groups to develop more precise specifications for particular annotation types. The full list of SC 4 working groups is as follows:

- WG1/WI-0: Terminology for Language Resources
- WG1/WI-1: Linguistic annotation framework
- WG1/WI-2: Meta-data for multimodal and multilingual information
- WG2/WI-3: Structural content representation (syntax and morphology)
- WG2/WI-4: Multimodal content representation
- WG2/WI-5: Discourse level representation
- WG3/WI-6a: Multilingual text representation
- WG4/WI-7: Lexicons
- WG5/WI-8: Validation of language resources
- WG5/WI-9: Net-based distributed cooperative work for the creation of LRs

We focus here on the work of WG1/WI-1, which will serve as the starting point for that of most of the others. This group will propose a *data architecture* consisting of basic mechanisms and data structures for linguistic annotation and representation, comprised of the following:

- *Basic components*: the basic constructs for building representations of linguistic information; specifically, identification of types of building blocks and ways to connect them.
- *General mechanisms*: representation techniques that make the annotations more compact and flexible and enable linking them to external sources of information; for example, sub-structure labeling, argument under-specification, restrictions on label values and/or disjunctions or lists to represent ambiguity or partiality, structure sharing; linking to

domain models, linking to other levels of annotation, etc.

- *Contextual data categories*: administrative (meta-) data relevant for processing, such as environment data (e.g., time stamps, spatial information); processing information (e.g., module that produced the representation; confidence level); interaction information (speaker, audience, etc.).

The following section outlines a linguistic framework which will serve as the starting point for development within SC 4. The current model is based on work on development of annotation formats for lexicons (Ide, et al., 2001), morphosyntactic and syntactic annotation (Ide & Romary, 2001a; Ide & Romary, 2001b; Ide & Romary, forthcoming), and which has been further developed within TC37/SC4 for the definition of TMF (Terminological Markup Framework; ISO 16642, under DIS ballot).

4. A Framework for Linguistic Annotation

Our fundamental assumption is that representation formats for linguistic data and its annotations can be modeled by combining a structural *meta-model,* that is, an abstract structure shared by all documents of a given type (e.g. syntactic annotation), with a set of *data categories* that are associated with the various components of the meta-model. Our work in SC4 is concerned, first, with identification of a reduced set of meta-models that can be used for any type of linguistic data and its annotations. Data categories, on the other hand, are defined by the implementer; interoperability among formats is ensured by providing a *Data Category Registry* in which the categories and relations required for a particular type of annotation are precisely defined.

The model for linguistic annotation must satisfy two general criteria:

1. It must be possible to instantiate it using a standard representational format;
2. It must be designed so as to serve as a pivot format into and out of which proprietary formats can be transduced, in order to enable comparison and merging, as well as operation on the data by common tools.

4.1. Abstract model for annotation

At its highest level of abstraction, an annotation is a set of data or information (in our case, linguistic information) that is associated with some other data. The latter is what could be called "primary" data (e.g., a part of a text or speech signal, etc.), but this need not be the case; consider, for example, the alignment of parallel translations, where the "annotation" is a link between two primary data objects (the aligned texts). Typically, primary data objects are represented by "locations" in an electronic file, for example, the span of characters comprising a sentence or word, or a point at which a given temporal event begins or ends (as in speech annotation). As such, at the base primary data objects are relatively simple in their structure; more complex data objects may consist of a list or set of contiguous or non-contiguous locations. Annotation objects, on the other hand, often have a more complex internal structure: syntactic annotation, for example, may be expressed as a tree structure, and may include more elemental annotations such as dependency relations (which is itself an annotation relating two objects, where the relation is directional (dependent-to-head)).

Thus, we can conceive of an annotation as a one- or two-way link between an annotation object and a point (or a list/set of points) or span (or a list/set of spans) within a base data set. Links may or may not have a semantics--i.e., a type--associated with them. Points and spans in the base data may themselves be objects, or sets or lists of objects. This abstract formulation can serve as the basis for defining a general model for linguistic annotation that can be realized in a standard representational format. In fact, this model is consistent with well-established data modeling concepts used in diverse areas, including knowledge representation (KR), object-oriented design, and database systems, and which inform fundamental data structures in computer science (trees, graphs, etc.) and database design (notably, the Entity-Relationship (ER) model). As such, the model provides us with established means to describe our data objects (in terms of composition, attributes, class membership, applicable procedures, etc.) and relations among them, independent of their instantiation in any particular form. It also ensures that standardized representation formats exist that can instantiate the model.

One way to represent linguistic annotation in terms of the abstract model is as a graph of elementary *structural nodes* to which one or more *information units* are attached. The distinction between the structure of annotations and the informational units of which it is comprised is, we feel, critical to the design of a truly general model for annotations. Annotations may be structured in several ways; perhaps the most common structure is hierarchical. For example, phrase structure analyses of syntax are structured as trees; in addition, hierarchy is often used to break annotation information into sub-components, as in the case of lexical and terminological information.

There are several special relations *among* annotations that must be represented in the model, including the following:

- *Parallelism:* two or more annotations refer to the same data object;
- *Alternatives:* two or more annotations comprise a set of mutually exclusive alternatives (e.g., two possible part-of-speech assignments, before disambiguation);
- *Aggregation:* two or more annotations comprise a list or set that should be taken as a unit.

Information units or *data categories* provide the semantics of the annotation. Data categories are the most theory and application-specific part of an annotation scheme. We do not attempt to define the relevant data categories for given types of annotation. Rather, we propose the development of a Data Category Registry to provide a framework in which the research community can formally define data categories for reference and use in annotation. To make them maximally interoperable and consistent with existing standards, data categories can be defined using RDF schemas to formalize the properties and relations associated with each. Note that RDF descriptions function much like class definitions in an

object-oriented programming language: they provide, effectively, templates that describe how objects may be instantiated, but do not constitute the objects themselves. Thus, in a document containing an actual annotation, several objects with the same type may be instantiated, each with a different value. The RDF schema ensures that each instantiation is recognized as a sub-class of more general classes and inherits the appropriate properties.

A formally defined set of categories will have several functions: (1) it will provide a precise semantics for annotation categories that can be either used "off the shelf" by annotators or modified to serve specific needs; (2) it will provide a set of reference categories onto which scheme-specific names can be mapped; and (3) it will provide a point of departure for definition of variant or more precise categories. Thus the overall goal of the Data Category Registry is not to impose a specific set of categories, but rather to ensure that the semantics of data categories included in annotations (whether they exist in the Registry or not) are well-defined and understood.

5. An Example

We illustrate a simple application of the framework presented above for the domain of morpho-syntactic annotation. For the purposes of illustration, it is necessary to make technical choices concerning the representation format. XML and related standards developed by the World Wide Web consortium appear at present to provide the best means to represent information structures intended to be transmitted across a network. For the purposes of linguistic resource representation, XML provides several important features:

- it is both Unicode and ISO 10646 compatible;
- XML namespaces provide the options of combining element definitions from multiple sources in an XML document, thereby fostering modularity and reuse;
- XML schemas provide a powerful means to define, constrain, and extend definitions of the structure and contents of classes of XML documents and document sub-parts;
- W3C has defined accompanying standards for inter- and intra-document linkage (XPath, XPointer, and Xlink) as well as document traversal and transformation (XSLT);
- XML is fully integrated with emerging standards such as the Resource Definition Framework (RDF) and DAML+OIL, which can be "layered" on top of XML documents to provide a formal semantics defining XML-instantiated objects and relations.

We have defined an XML format for representing linguistic annotations called the *Generic Mapping Tool (GMT*). The GMT defines XML elements for encoding annotation structure (primarily, a nestable `<struct>` element) and data categories (a nestable `<feat>` tag). A `<seg>` element provides a pointer to the annotated data using XPointers. Relations among objects can be specified explicitly using a `<rel>` element or may be implicit in the hierarchical nesting of `<struct>` elements. The GMT is described in detail in Ide & Romary, 2001b. We stress, however, that the details of the XML format—in particular, element names—is arbitrary; the only requirement is that the underlying data model can be expressed using the format.

5.1. Morpho-syntactic annotation

Morpho-syntactic annotation provides a good example of how the data model instantiated in the GMT is applied, and demonstrates some of the mechanisms required for representing annotations in general. Morpho-syntactic annotation involves the identification of word classes over a continuous stream of word tokens. The annotations may refer to the segmentation of the input stream into word tokens, but may also involve grouping together sequences of tokens or identifying sub-token units (or morphemes), depending on the language under consideration and, in particular, the definitions of "word" and "morpheme" as applied to this language. The description of word classes may include one or several features such as syntactic category, lemma, gender, number etc., which is again dependent on the language being analyzed.

Morpho-syntactic annotation can be represented by a single type of structural node (named W-level) representing a word-level structure unit. One or several information units are associated with each structural node.

For the purposes of illustration, we identify the following data categories (in practice these would be defined in reference to categories in the Data Category Registry):

- /lemma/: contains or points to a reference word form for the token or sequence of tokens being described;
- /part of speech/: a reference to a morpho-syntactic category;
- /confidence/: a confidence level assigned by the manual or automatic annotator in ambiguous cases.
- /gender/: the grammatical gender information associated with a word token or a sequence of word tokens;
- /number/: the grammatical gender information associated with a word token or a sequence of word tokens;
- /tense/: the grammatical tense information associated with a word token or a sequence of word tokens;
- /person/: the grammatical person information associated with a word token or a sequence of word tokens.

The following provides an example of the morpho-syntactic annotation of the sentence "Paul aime les croissants" in the GMT format:[1]

```
<struct type="MSAnnot">
 <struct type="W-level">
   <feat type="lemma">Paul</feat>
   <feat type="pos">PNOUN</feat>
      <seg target="#w1"/>
 </struct>
 <struct type="W-level">
   <feat type="lemma">aimer</feat>
   <feat type="pos">VERB</feat>
   <feat type="tense">present</feat>
   <feat type="person">3</feat>
   <seg target="#w2"/>
 </struct>
```

[1] For brevity, we use an abbreviated pointer syntax to refer to the primary data in this example.

```
<struct type="W-level">
  <feat type="lemma">le</feat>
  <feat type="pos">DET</feat>
  <feat type="number">plural</feat>
  <seg target="#w3"/>
</struct>
<struct type="W-level">
  <feat type="lemma">croissant</feat>
  <feat type="pos">NOUN</feat>
  <feat type="number">plural</feat>
  <seg target="#w4"/>
</struct>
</struct>
```

Note that there is no limit to the number of information units that may be associated with a given structural node (as opposed to the text based representations that are usually provided by available POS taggers). It is also possible to structure the annotations by embedding <feat> elements to reflect a more complex feature-based annotation, or by pointing to a lexical entry providing the information.

In some cases, the morpho-syntactic annotation of a word or sequence of words requires a hierarchy of word level structures (e.g., when a word token results from the combination of several morphemes that must be annotated independently). For example, some occurrences of the token "du" in French can be analyzed as the fusion of the preposition "de" with the determiner "le" (as in "la queue *du* chat"). This is handled by embedding word-level structures as follows:

```
<struct type="W-level">
  <seg target="#w1"/>
  <struct type="W-level">
     <feat type="lemma">de</feat>
     <feat type="pos">PREP</feat>
  </struct>
  <struct type="W-level">
     <feat type="lemma">le</feat>
     <feat type="pos">DET</feat>
  </struct>
</struct>
```

Conversely, annotation of compound words may involve associating a single lemma to a sequence of word tokens at the surface level. In this case, the lemma is attached to the higher level of embedding and reference to the source is given at the leaves of the hierarchy, as in the following representation of the compound "pomme de terre" in French :

```
<struct type="W-level">
  <feat type="lemma">
        pomme_de_terre</feat>
  <feat type="pos">NOUN</feat>
  <struct type="W-level">
     <seg target="#w1"/>
     <feat type="lemma">pomme</feat>
     <feat type="pos">NOUN</feat>
  </struct>
  <struct type="W-level">
     <seg target="#w2"/>
     <feat type="lemma">de</feat>
     <feat type="pos">PREP</feat>
  </struct>
  <struct type="W-level">
     <seg target="#w3"/>
     <feat type="lemma">terre</feat>
     <feat type="pos">NOUN</feat>
  </struct>
</struct>
```

The ability to specify a hierarchical structure where needed enables specification of the level of granularity required. This is especially critical for a representation scheme, since the granularity of the segmentation in (or associated with) the primary data may not directly correspond to the level of granularity required for the annotation.

5.1.1. Alternatives

Morpho-syntactic annotation can be used to illustrate the representation of both structural and informational alternatives, which arises when a given word token is associated with two or more word classes. For example, the French word "bouche" which can be derived both from the verb "boucher" and the noun "bouche", which can be represented as follows:

```
<struct type="W-level">
  <seg target="#w1"/>
  <alt>
    <feat type="lemma">boucher</feat>
    <feat type="pos">VERB</feat>
    <feat type="tense">present</feat>
    <feat type="confidence">0.4</feat>
  </alt>
  <alt>
    <feat type="lemma">bouche</feat>
    <feat type="pos">NOUN</feat>
    <feat type="confidence">0.6</feat>
  </alt>
</struct>
```

5.1.2. Relating annotation levels

We assume the use of stand-off annotation; that is, an annotated corpus is represented as a lattice of stand-off annotation documents pointing to a primary source or intermediate annotation levels. However, depending on the point of view, the relations between various annotation levels can be more or less explicit. It is possible to identify three major ways to relate different levels of annotation: temporal anchoring, event-based anchoring, and object-based anchoring.

Temporal anchoring associates positional information to each structural level. This positional information is typically represented as a pair of numbers expressing the starting point and ending point of the segment being described. To do so in our framework, we introduce two attributes for the <seg> element:

- /startPosition/: the temporal or offset position of the beginning of the current structural node;
- /endPosition/: the temporal or offset position of the end of the current structural node.

For example, the following associates a phonetic transcription with a given portion of a primary text:

```
<struct type="phonetic">
  <seg startsAt="2300"
       endsAt="3200"/>
  <feat type="phone">iy</feat>
</struct>
```

We also define an event-based anchoring, which effectively introduces a structural node to represent a location in the text, to which all annotations for the object

at that location can refer. This strategy is useful in two cases:

- Situations where it is not possible or desirable to modify the primary data by inserting markup to identify specific objects or points in the data (e.g., speech annotation, associated with a speech signal, or in general any "read-only" data).
- Primary data marked with "milestones", such as time stamps in speech data, where spans across the various milestones must be identified. In this case, the `<struct>` elements represent the markup for segmentation (e.g., segmentation into words, sentences, etc.).[2]

To represent this, we introduce a specific type of structural node, named *landmark*, which is referred to by annotations for the defined span, as follows:

```
<struct type="landmark">
    <seg startsAt="2300"
         endsAt="3200"/>
</struct>
```

The third mechanism, object-based anchoring, enables pointing from a given level to one or several structural nodes at another level. This mechanism is particularly useful to make dependencies between two or more annotation levels explicit. For example, syntactic annotation can refer directly to the relevant nodes in a morpho-syntactically annotated corpus, in order, for example, to identify the correct NP "le chat" in "la queue du chat", as shown below:

```
<!-- Morphosyntactic level -->
<struct type="W-level">
   <seg target="#w3">
   <struct type="W-level">
     <seg target="#w3.1">
     <feat type="lemma">de</feat>
    <feat type="pos">PREP</feat>
    </struct>
     <struct type="W-level">
       <seg target="#w3.2">
        <feat type="lemma">le</feat>
       <feat type="pos">DET</feat>
       <feat type="gender">masc</feat>
    </struct>
  </struct>
   <struct type="W-level">
       <seg target="#w4">
       <feat type="lemma>chat</feat>
       <feat type="pos">NOUN</feat>
  </struct>
</struct>
<!-- Syntactic level (simplified) -->
<struct>
   <feat type="synCat">NP</feat>
   <seg targets="w3.2 w4"/>
</struct>
```

[2] The annotation graph (AG) formalism (Bird and Liberman, 2001) was explicitly designed to deal with time-stamped data. However, we feel the AG is not sufficiently general because (1) AG reifies the "arc" and distinguishes it from identification of spans via, e.g., XML tags; and (2) AG requires *ad hoc* mechanisms to deal with hierarchically organized annotations. In both cases, AG requires different mechanisms to treat analogous constructs.

5.2. Summary

The framework presented here for linguistic annotation is intended to allow for variation in annotation schemes while at the same time enabling comparison and evaluation, merging of different annotations, and development of common tools for creating and using annotated data. We have developed an abstract model for annotations that is capable of representing the necessary information while providing a common encoding format that can be used as a pivot for combining and comparing annotations, as well as an underlying format that can be manipulated and accessed with common tools. The details presented here provide a look "under the hood" in order to show the flexibility and representational power of the abstract scheme. However, the intention is that annotators and users of annotation schemes can continue to use their own or other formats with which they are comfortable; as long as the underlying data model is the same, translation into and out of this or any other instantiation of the abstract format will be automatic.

Our framework for linguistic annotation is built around some relatively straightforward ideas: separation of information conveyed by means of structure and information conveyed directly by specification of content categories; development of an abstract format that puts a layer of abstraction between site-specific annotation schemes and standard specifications; and creation of a Data Category Registry to provide a reference set of annotation categories. The emergence of XML and related standards, such as RDF, provides the enabling technology. We are, therefore, at a point where the creation and use of annotated data and concerns about the way it is represented can be treated separately—that is, researchers can focus on the question of *what* to encode, independent of the question of *how* to encode it. The end result should be greater coherence, consistency, and ease of use and access for linguistically annotated data.

6. Conclusion

ISO TC37/SC4 is just beginning its work, and will use the general framework discussed in the preceding sections as its starting point. However, the work of the committee will not be successful unless it is accepted by the language processing community. To ensure widespread acceptance, it is critical to involve as many representatives of the community in the development of the standards as possible, in order to ensure that all needs are addressed. This paper serves as a call for participation to the language processing community; those interested should contact the TC 37/SC 4 chairman (Laurent Romary: romary@loria.fr).

7. References

Bird, S. & Liberman, M. (2001). A formal framework for linguistic annotation. *Speech Communication,* 33:1-2, 23-60.

Bunt. H. & Romary, L. (to appear). Towards Multimodal Content Representation. *Proceedings of the Workshop on International Standards for Terminology and Language Resource Management*, Las Palmas, May 2002.

Ide, N. & Romary, L. (2001b). A Common Framework for Syntactic Annotation. *Proceedings of ACL'2001,* Toulouse, 298-305.

Ide, N., Kilgarriff, A., & Romary, L. (2000). A Formal Model of Dictionary Structure and Content. *Proceedings of Euralex 2000,* Stuttgart, 113-126.

Ide, N. & Romary, L. (2001a). Standards for Language Resources, *IRCS Workshop on Linguistic Databases*, Philadelphia, 141-49.

Ide, N. & Romary, L. (forthcoming). Encoding Syntactic Annotation. In Abeillé, A. (ed.). *Treebanks: Building and Using Syntactically Annotated Corpora.* Dordrecht: Kluwer Academic Publishers.

Sponsors

Research Academic Computer Technology Institute
University of Patras

Co-operating Organisations

Global Wordnet Association

The Workshop Programme

8:30 - 13:30 Morning Session: Wordnet Applications, Evaluation and Standardization

8:30 - 8:45 Welcome & Introduction; Overview about the workshop

8:45 - 9:30 **KEYNOTE SPEECH**
Christiane Fellbaum (University of Princeton):
Going global: Issues in the standardization of wordnets

9:30 - 10:00 Neeme Kahusk (University of Tartu):
A Lexicographer's Tool for Word Sense Tagging according to Wordnet

10:00 - 10:30 Piklu Gupta (Fraunhofer Institut Darmstadt):
Approaches to Checking Subsumption in GermaNet

10:30 - 11:00 Graham Katz, Jahn-Takeshi Saito, Joachim Wagner, Philip Reuter, Sabine Reinhard & Michael Burke (University of Osnabrueck):
Evaluation of GermaNet: Problems using GermaNet for Automatic Word Sense Disambiguation

11:00 - 11:30 Coffee break

11:30 - 12:00 Karel Pala & Pavel Smrz (University of Brno):
Glosses in WordNet 1.5 and their Standardization/ Consistency
(The Exercise for BalkaNet)

12:00 - 12:30 Lothar Lemnitzer & Claudia Kunze (University of Tuebingen):
Standardizing Wordnets in a Web-compliant Format: The Case of GermaNet

12:30 - 13:00 Tomas Pavelek & Karel Pala (University of Brno):
Wordnet Standardization from a practical point of view

13:00 - 13:30 Overall Discussion and Conclusion for 1st Part of the Workshop:
* standardization and compatibility guidelines
* application and evaluation scenarios envisaged
* future actions

13:30 - 14:30 Lunch break

14:30 - 19:30 Afternoon Session: Wordnet Structures and Applications for the less-studied Languages

14:30 - 15:00 **Introduction Speech**
Prof. Dimitris N. Christodoulakis (University of Patras):
Structures of Semantic Networks

15:00 - 15:30 Dan Tufis (Romania Academy) & Dan Cristea (A.I. Cuza University):
Methodological issues in building the Romanian Wordnet and consistency checks in Balkanet

15:30 - 16:00 Gabor Proszeky (MorfoLogic) & Marton Mihaltz (Eotvos Lorand University, Budapest):
Automatism and User Interaction: Building a Hungarian Wordnet

16:00 - 16:00 Dimitris Avramidis, Maria Kyriakopoulou, George Kourousias, Sofia Stamou & Manolis Tzagarakis (University of Patras, *RA* CTI Patras):
Viewing Semantic Networks as Hypermedia

16:30 - 17:00 Coffee break

17:00 - 17:30 Ioannis - Dimitris Koutsoumpos, Manolis Tzagkarakis & Dimitris Christodoulakis (University of Patras, *RA* CTI Patras):
Requirements for Domain-Specific Wordnets

17:30 - 18:00 Kadri Vider (University of Tartu):
Notes about labeling semantic relations in Estonian Wordnet

18:00 - 18:30 Irina V. Azarova, Olga A. Mitrofanova, Anna Sinopalnikova & Ilya Oparin (State University St-Petersburg):
Building the Lexical Database for the Russian Language

18:30 - 19:00 Natalia V. Loukachevitch & Boris V. Dordov (Moscow State University):
Development and Use of Thesaurus of Russian Language RuThes

19:00 - 19:30 Overall Discussion and Conclusion for the 2nd Part of the Workshop

Workshop Organisers

Dimitris N. Christodoulakis, Patras University (Greece)
Claudia Kunze, Lothar Lemnitzer, University of Tuebingen (Germany)
Karel Pala, Masaryk University Brno (Czech Republic)

Workshop Programme Committee

Christiane Fellbaum, Princeton University (USA)
Piek Vossen, Irion Technology Delft (The Netherlands)
Kemal Oflazer, Sabanci University Istanbul (Turkey)
Jeroen Hoppenbrouwers, Tilburg University (The Netherlands)
Randee Tengi, Princeton University (USA)
Wim Peters, Sheffield University (GB)
Kadri Vider, University of Tartu (Estonia)
Julio Gonzalo, UNED Madrid (Spain)
Palmira Marrafa, University of Lisboa (Portugal)
Paul Buitelaar, DFKI Saarbruecken (Germany)
Andreas Wagner, University of Tuebingen (Germany)
Erhard Hinrichs, University of Tuebingen (Germany)
Simonetta Montemagni, University of Pisa (Italy)
Robert Ermers, Van Dale Data BV (The Netherlands)

Table of Contents

Workshop on Wordnet Structures and Standardization, and how these affect Wordnet Applications and Evaluation

Author Index

A Lexicographer's Tool for Word Sense Tagging According to WordNet

Neeme Kahusk

University of Tartu
Department of General Linguistics
Tiigi 78-204, 50410 Tartu, Estonia
nkahusk@psych.ut.ee

Abstract

This paper describes a Web-based tool for tagging word senses according to Estonian WordNet. The tool makes use of EuroWordNet import-export format that is converted into XML. The user interface is divided into three main parts that provide information about the word to be tagged: (1) context (2) morphological analysis and (3) entries in lexicon (WordNet). The tool is aimed to facilitate lexicographers' work with languages, where morphological information is important at word sense disambiguation. The advantages of the tool and problems met are discussed in detail.

1. Introduction

The task of tagging word senses is demanding for the lexicographers. They have to find the words to tag from the text, find is the word presented in the lexicon, is there an appropriate meaning for the word in the lexicon, and finally, assign the meaning to the word in question.

Estonian is an agglutinative language, rich of word forms. A lot of word forms are ambiguous, and before getting lemma some morphological analysis is needed.

In the very beginning, the linguists who did the job, had to edit a plain text file and write the appropriate sense of word after its morphological reading.

To carry out word sense disambiguation, lexicographer has to know what are the different senses of the word. Thatswhy (s)he needs to see at least, definition (gloss), and example(s) of usage, and one hyperonym. Up to now, the people who did the job, edited files with a simple text editor and used Polaris tool, that seriously limited the number of workplaces where the job could be done. This drawback, and the fact that editing a file, where one can see only one word on a line, followed by morphological analysis, is a potential source of errors, rose the need for a tool that would be more task-oriented and usable on client-server basis.

As a result of integration output of morphological analyser and an automatic WSD system for finding words not in the thesaurus and pre-selecting senses, a tool was created that makes word sense tagging more accurate and less time-consuming.

The lexicographer's tool is working in two stages, off-line (preparatory) and on-line.

By implementing the tool we have found several problems that were not noticed at manual file-editing process. The dividing into parts of speech is a bit different in wordnets and Estonian morphological tradition; morphology, syntax and semantics are more tightly connected to each other than one can suppose. An additional feature of tagging multi-word units is needed.

2. Preparatory stage

For off-line stage, the following data files are needed: (1) current thesaurus (in import-export format); (2) file to disambiguate by word senses — it should be analysed by Estmorf, and piped through fs2kym. To ensure that the analysed text file has correct format (each word must have exactly one analysis), a small vaidating script is applied to it.

During off-line or preparatory stage, current version of Estonian WordNet (EstWN) is converted into XML. Then semyhe is applied to the data in two runs: on first run, nouns are disambiguated, on second one, verbs are disambiguated.

2.1. Morphological analysis

Morphological analysis is carried out with Estmorf provided by Kaalep (1997). In its original form, Estmorf outputs for every word its structure (stem, affixes and suffixes), part of speech and inflectional categories.

```
pea
    pea+0 //_D_ //
    pea+0 //_S_ sg g, sg n, //
    pida+0 //_V_ o, //
    pida+0 //_V_ o, //
```

Figure 1: Output of Estmorf from word form 'pea'.

Declinable words are differentiated into following parts of speech: common nouns or substantives (_S_), proper nouns (_H_), adjectives with positive degree, comparative degree and superlative degree (_A_, _C_, and _U_ respectively), numerals (_N_ cardinal, _O_ ordinal), pronouns and acronyms (_Y_). Possible sets of inflectional categories are given on the same line, if they occur inside one paradigm (structure and part of speech). Figure 1 illustrates analysis of 'pea':

1. adverb ('soon'; uninflected),
2. noun ('head'; singular, genitive or nominative),
3. and 4. verb (two homonyms[1]: 'keep' and 'must', both imperative, the last one modal, but this analysis does not show such features).

[1]There are more meanings: in EstWN there are 13 senses of verb 'pidama', but they can divided into 2 groups — the modal (3 senses) and main (10 senses) ones.

```
pea
    pea+0 //_D_ //
    pea+0 //_S_ com sg gen //
    pea+0 //_S_ com sg nom //
    pida+0 //_V_ main imper pres ps2 sg ps af //
    pida+0 //_V_ main imper pres ps2 sg ps neg //
    pida+0 //_V_ main indic pres ps neg //
    pida+0 //_V_ mod imper pres ps2 sg ps af //
    pida+0 //_V_ mod imper pres ps2 sg ps neg //
    pida+0 //_V_ mod indic pres ps neg //
```

Figure 2: Output of Estmorf from word form 'pea' piped through fs2kym.

It turned out that the output of Estmorf is not very good for disambiguation purposes. At first, the Estmorf analysis line itself contains ambiguous readings (different inflectional categories, although being inside one paradigm). Second, in some cases, the differentiation into parts of speech is too detailed. The authors of Estmorf have developed a conversion program fs2kym that modifies the output. Unfortunately the last version of fs2kym is not fully documented yet, the output is pretty much the same as used by Puolakainen (2001) and Roosmaa et al. (2001) for morphological disambiguation based on constraint grammar and syntactic analysis.

In fs2kym output, substantives and proper nouns are tagged as '_S_ com' and '_S_ prop' respectively. So are numerals and ordinals, '_N_ card' stands for numeral and '_N_ ord' for ordinal. In the same way all adjectives are tagged as _A_, their degree is added with next token: '_A_ pos' for positive adjective, '_A_ comp' for comparative and '_A_ super' for superlative one. For verbs, fs2kym adds inflectional readings with all possible solutions. Figure 2 illustrates previous example analysed with Estmorf and piped through fs2kym:

1. adverb ('soon'; uninflected),
2. noun ('head'; singular, genitive),
3. noun ('head'; singular, nominative),
4. verb ('to keep; to consider'; main, imperative, present, 2. person, singular, personal, affirmative)
5. verb ('to keep; to consider'; main, imperative, present, 2. person, singular, personal, negative)
6. verb ('to keep; to consider'; main, indicative, present, personal, negative)
7. verb ('must; should'; modal, imperative, present, 2. person, singular, personal, affirmative)
8. verb ('must; should'; modal, imperative, present, 2. person, singular, personal, negative)
9. verb ('must; should'; modal, indicative, present, personal, negative)

2.2. Preliminary word sense tagging

Preliminary word sense tagging is done with Semyhe system, as described by Vider and Kaljurand (2001). The main idea of Semyhe is based on a similar system by Agirre and Rigau (1996), using distances between the nodes corresponding to the word senses in the WordNet tree and the density of the tree. Contrast to the Agirre and Rigau system, Semyhe disambiguates both, nouns and verbs. Nouns and verbs are disambiguated in two separate runs, as they do not share the same hyperonym-hyponym hierarchies.

Fs2kym-piped output of Estmorf serves as input for Semyhe. As our aim at present stage is generating a word-sense disambiguated corpus, the Estmorf output is disambiguated by hand, so every word has only one reading. Semyhe adds its output to Estmorf analysis, an example is given in Figure 3 (upper part). Semyhe analysis is added to substantives and main or modal verbs. After last piece of morphological info '@' is added, then lemma in dictionary form (singular nominative for substantives, supine affirmative illative for verbs). The last two fields are separated with colon, last number denotes number of senses found from EstWN, the last but one is sense number found by Semyhe. If Semyhe finds more than one possible analysis, then the alternatives are separated by number sign (#).

2.3. XML format

The import-export (i/e) format of a language wordnet in EuroWordNet is derived from GEDCOM standard (Louw, 1998). The GEDCOM format[2] itself is hierarchical, so the initial conversion into XML is rather simple.

The main idea in converting EWN i/e format into XML was simplicity of conversion and not well-formedness or size of resulting file. So the current version of XML format is a simple translation of GEDCOM-format into XML: node labels are translated into elements (with some exceptions explained below), and node contents are translated into values of attribute 'VALUE'. If element name consists of multiple words, element will be built from first letters of label name (PART_OF_SPEECH will be POS). Still there are some labels in EWN format that would be ambigous at such conversion. They differ only by plural ending. Such labels are converted so, that the ending 'S' is added to every subword: e.g. USAGE_LABELS will be <USLS> and USAGE_LABEL will be <UL> (Figure 4).

[2] http://www.gendex.com/gedcom55/55gctoc.htm

```
Pidas
    pida+s //_V_ main indic impf ps3 sg ps af // @ pidama:6:12
veidi
    veidi+0 //_D_ //
aru
    aru+0 //_S_ com sg part // 1 @ aru:1:1
ja
    ja+0 //_J_ crd //
lisas
    lisa+s //_V_ main indic impf ps3 sg ps af // @ lisama:3:3
liipsukese
    liipsu=ke+0 //_A_ pos sg gen //
liha
    liha+0 //_S_ com sg gen // @ liha:1:3
.
    . //_Z_ Fst //
-------------------------------------------------------
<s>
    <head id="1630" lemma="pidama" pos="V" class="main"
    rest="indic impf ps3 sg ps af" noofsenses="12" semyhe="6">Pidas</head>
    <other id="1631" pos="D">veidi</other>
    <head id="1632" lemma="aru" pos="S" class="com" rest="sg part"
    noofsenses="1" semyhe="1">aru</head>
    <other id="1633" pos="J" class="crd">ja</other>
    <head id="1634" lemma="lisama" pos="V" class="main"
    rest="indic impf ps3 sg ps af" noofsenses="3" semyhe="3">lisas</head>
    <other id="1635" pos="A" class="pos">liipsukese</head>
    <head id="1636" lemma="liha" pos="S" class="com"
    rest="sg gen" noofsenses="3" semyhe="1">liha</head>
    <other id="1637" pos="Z" class="Fst">.</other>
</s>
```

Figure 3: Upper: Output of Semyhe. Lower: The same sentence in XML format. Explanations in text. The analysed sentence can be translated like '[she] considered a bit and added a little slice of meat.'

The format will do for simple tasks like converting the thesaurus to form needed for literal browsing, but is not very suitable for more general tasks and is definitely not a good human-readable one.

There are several versions of EWN in XML that are more readable: (Kunze and Lemnizer, 2002; Smrz, 2002; Dowdall et al., 2002), and there is a special tool for viewing and editing WordNet in XML format: VisDic (Pavelek and Pala, 2002).

2.4. Text File in XML

The text file is also converted into XML. The format is similar to the one that was used at Senseval-2 task and training files, with some modifications. The `<sat>` elements are omitted (see sec. 4.3.), and `<other>` element is introduced for words being not heads, and for other tokens (punctuation marks). Morphological information is given as attributes for `<head>`[3]: `lemma, pos, class` and `rest`, the last one for other morphological reading. The identification number (position of token in text) is given as `id` attribute. Semyhe adds more attributes: `noofsenses` for number of senses in EstWN, `semyhe` for Semyhe applied sense number (Figure 3, lower part). Finally, the sense number assigned by lexicographer, will be inserted as value of `sense` attribute (not shown in the figure).

[3] `id, pos, class` and `rest`, if applicable, are added to `other` elements as well.

3. User Interface

After entering his/her name and selecting file to work with, user can move to main interface of the program.

The program window is divided into four frames: the main frame for text being analysed, morf frame and thesaurus frame. The lowest frame is for entering comments. In the uppermost frame user can browse text, words to disambiguate are in boldface, and depending on browser settings, underlined. Each word to disambiguate is preceeded by an identification number for references in comments.

User has to select appropriate sense for each word that needs disambiguation (these words are emphasized in bold and linkable). By clicking on appropriate word, user can see morphological information about the word (part of speech and word class), and thesaurus entries. The thesaurus entries are presented in a table: each row represents one synset. The 2nd column of the table shows members of synsets with sense numbers. In the 3rd column, there are explanations (glosses), and in last column there are ex-

amples of usage. The first column indicates hyperonym of each synset, displaying its first literal and sense number.

The sense numbers to select are immediately after the emphasised words in the text, as selection boxes. The sense that semyhe offered to the word is pre-selected. User has to select appropriate sense, and after finishing (or leaving the program) save his/her work with appropriate button in the lowest frame.

4. Problems of compatibility

4.1. Part of speech, WSD and syntax

There parts of speech used in EuroWordNet are: noun, proper noun, verb, adjective, adverb. Semyhe looks at Estmorf output only for nouns and verbs. With noun it gets, by default, `_S_ com` and `_S_ prop` — that is substantives and proper nouns. In EWN, numerals (`_N_ card` and `_N_ ord` in Estmorf output) are classified also as nouns. Seems to be a minor bug, but there is a famous example of homonymy in Estonian: 'viis' means number 'five', and 'a way to do something', and 'melody'. For an English analog, consider the homophony of '4' and 'for', for example. By using morphologically disambiguated text, we have already pre-selected one sense (or reduced the possible number of senses) and left the other(s). The same stands for some features, that belong to syntax: verb may be main, auxiliary or modal, by determining the type, we can tell the sense.

4.2. A word about encoding

As Latin alphabet is used to write Estonian, it seems that there should not be a problem with encoding. There are some umlaut letters in Estonian (ä, ö, ü and Ä, Ö, Ü) that rise no problems, since they can be found in many West-European languages and in Latin-1 encoding as well. Some ten years ago there have been some problems with another quite frequent letter 'õ, Õ', known as o tilde. It is in Latin-1 now and is OK, but historically there have been problems, as it was not included in so-called 'extended ASCII character set' provided by first PC-s running DOS.

There are some really 'nasty' letters in Estonian alphabet, s caron and z caron (š, ž, Š, Ž). They are not very frequent, but they figure in important foreign words like 'žanr' (genre), 'dušš' (shower), or 'garaaž' (garage), that do not have any synonyms without these 'horned' letters. There have been proposals to stop using them and replace them with 'sh' and 'zh', like in English word 'bush', but it can happen in Estonian that syllable boundary—or even word boundary in compounds—is between 's' and 'h' like in 'klaashelmes' (klaas+helmes, glass bead), so it is not reasonable to use 'sh' as ligature. These letters are not contained in Latin-1 character set.

The new standard sets Latin-15 as character set of Estonian, but many applications do not recognise it yet.

The caron letters are in Latin-2 (Windows 1250, Central Europe) encoding, but the places of 'õ' and 'Õ' are taken by 'ő' and 'Ő' (o with double acute, used in Hungarian). The bad news is, that our Polaris uses Windows 1250 encoding, and so are the export files. In order to get relevant results about words containing 'š' and 'ž', we had to convert the EWN export files into Latin-15 before applying semyhe. Still, some XML tools do not recognize Latin-15 encoding, so we must rebuild everything for at least UTF-8 encoding, to get rid of constant converting to and forth.

4.3. Multi-word expressions

There is still a problem with multi-word expressions. Semyhe does not recognise multi-word expressions at present stage, and so they get no sense number, nor display in thesaurus frame (unless they are synonyms of some one-word literal). So lexicographers have to mention the multi-word units separately in the comment field. The problem is more accute with multi-word verbs, as they may consist of words the senses of which by themselves have little, if anything, to do with the meaning of the whole phrase. Fortunately enough, we are going to have a representative list of of Estonian phrasal verbs and idiomatic expressions by Kaalep and Muischnek (2002). Even with the ready-made list the algoritm of founding multi-word semantic units from the text would not be trivial. The same problems that Kaalep and Muischnek met at compiling the database, will haunt us at finding multi-word units from text by semyhe: relevant words may be intervened by other words in the sentence, and we need to meet the ends of lexicon form of word (lemma) and the form that is used in the text. The question of multi-word phrases, verbs in particular, is not a minor one, as there are 1070 two-word phrases in EstWN, 824 of them verb phrases. That makes about 28% of all verb literals[4].

5. Conclusions and future improvements

The tool has turned out to be useable, but there are problems as well, some of them being technical, some theoretical.

We are using morphologically disambiguated text. For semantic analysis, only these nouns and main (or modal) verbs are presented, that are currently in the thesaurus. If there has been made a mistake during morphological disambiguation, a lexicographer using the program can not make any corrections directly, but only make notes about the mistake in the comments field.

Multi-word phrases missing from analysis is a serious drawback, especially in case of verbs. If user does not see the possibility of multi-word phrase in the thesaurus, then it takes him or her much more time to think about this possibility among others. This slows down the process of analysis and is a potential source of errors.

The possibility to see all senses of a word together, in one table, is an advantage that even Polaris does not afford. This gives us direct comparison of senses, that is useful not only for WSD task, but for improvement of the thesaurus as well.

6. References

E. Agirre and G. Rigau. 1996. Word sense disambiguation using conceptual density. In *Proceedings of the 16th International Conference on Computational Linguistics*, Copenhagen.

[4]These figures are calculated on EstWN version kb39, as it was used for Senseval-2

J. Dowdall, M. Hess, N. Kahusk, K. Kaljurand, M. Koit, F. Rinaldi, and K. Vider. 2002. Technical terminology as critical resource. To be published in LREC 2002 Proceedings.

H.-J. Kaalep and K. Muischnek. 2002. Using the text corpus to create a comprehensive list of phrasal verbs. To be published in LREC 2002 Proceedings.

H.-J. Kaalep. 1997. An estonian morphological analyser and the impact of a corpus on its development. *Computers and the Humanities*, 31:115–133.

C. Kunze and L. Lemnizer. 2002. Adapting GermaNet for the web. In *Proceedings of 1st International Global Wordnet Conference, January 21–25, 2002*, pages 174–181, Mysore, India. Central Institute of Indian Languages.

M. Louw. 1998. Polaris User's Guide. The EuroWordNet Database Editor. EuroWordNet (LE-4003), Deliverable D023D024.

T. Pavelek and K. Pala. 2002. Visdic — a new tool for wordnet editing. In *Proceedings of 1st International Global Wordnet Conference, January 21–25, 2002*, pages 192–195, Mysore, India. Central Institute of Indian Languages.

T. Puolakainen. 2001. *Eesti keele arvutigrammatika: morfoloogiline ühestamine*. Ph.d. diss., University of Tartu. In Estonian. English title: Computer Grammar of Estonian: Morphological Disambiguation.

T. Roosmaa, M. Koit, K. Muischnek, K. Müürisep, T. Puolakainen, and H. Uibo. 2001. *Eesti keele formaalne grammatika*. University of Tartu, Tartu, Estonia. In Estonian. English title: The Formal Grammar of Estonian.

P. Smrz. 2002. Storing and retrieving WordNet database (and other structured dictionaries) in XML lexical database management system. In *Proceedings of 1st International Global Wordnet Conference, January 21–25, 2002*, pages 201–206, Mysore, India. Central Institute of Indian Languages.

K. Vider and K. Kaljurand. 2001. Automatic WSD: Does it make sense of Estonian? In *Proceedings of SENSEVAL-2: Second International Workshop on Evaluating Word Sense Diasambiguating Systems*, pages 159–162.

```
0 @55718@ WORD_MEANING
  1 PART_OF_SPEECH "n"
  1 VARIANTS
    2 LITERAL "job"
      3 SENSE 2
      3 DEFINITION "what you should do for a living"
      3 EXTERNAL_INFO
        4 SOURCE_ID 1
          5 TEXT_KEY "08508615-n"
2 LITERAL "work"
  3 SENSE 1
  3 STATUS "New"
     3 DEFINITION "what you do for a living"
  3 USAGE_LABELS
    4 USAGE_LABEL "sub"
      5 USAGE_LABEL_VALUE "Medicine"
  3 FEATURES
    4 FEATURE "connotation"
      5 FEATURE_VALUE "figurative"
   /---/
-----------------------------------------------------------------------
<?xml version="1.0"?>
<THESAURUS>
<WM ID="55718">
  <POS VALUE="n"/>
  <VARIANTS>
    <LITERAL VALUE="job">
      <SENSE VALUE="2"/>
      <DEFINITION VALUE="what you should do for a living"/>
      <EI>
        <SI VALUE="1">
          <TK VALUE="08508615-n"/>
        </SI>
      </EI>
    </LITERAL>
    <LITERAL VALUE="work">
      <SENSE VALUE="1"/>
      <STATUS VALUE="New"/>
      <DEFINITION VALUE="what you do for a living"/>
      <USLS>
        <UL VALUE="sub">
          <ULV VALUE="Medicine"/>
        </UL>
      </USLS>
      <FEATURES>
        <FEATURE VALUE="connotation">
          <FV VALUE="figurative"/>
        </FEATURE>
      </FEATURES>
    /---/
  </VARIANTS>
</WM>
</THESAURUS>
```

Figure 4: An extraction from EWN import-export format (upper) translated into XML format (lower)

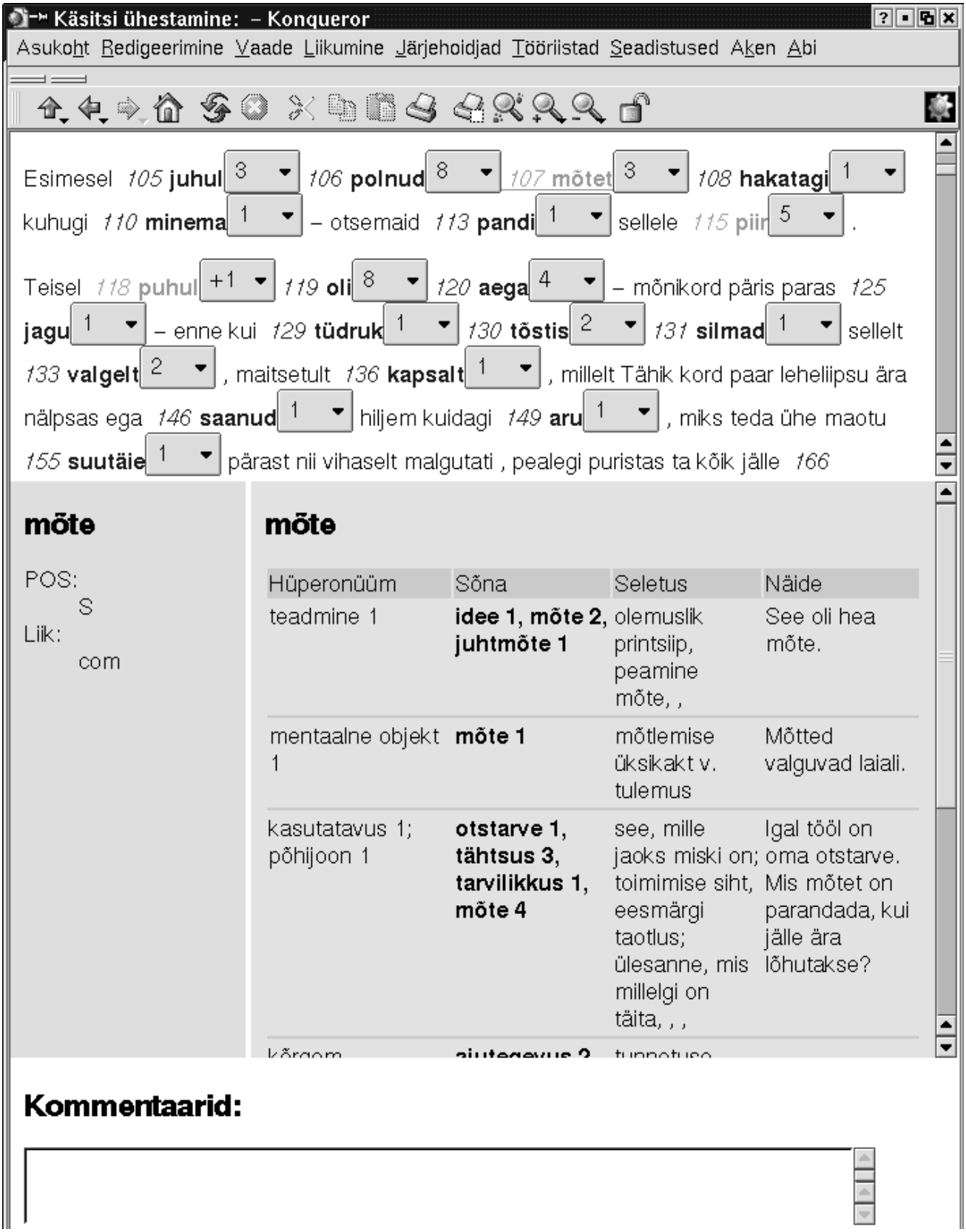


Figure 5: The user interface of the lexicographer's tool as seen in Konqueror browser. In upper frame, there is current text; in left part of middle frame (gray background), there is some morphological information (part of speech and class), in right part of middle frame there is semantic information from Estonian WordNet presented in a table; the lowest frame is for lexicographer's comments.

Approaches to Checking Subsumption in GermaNet

Piklu Gupta

Fraunhofer Integrated Publication and Information Systems Institute
Dolivostr. 15
D-64283 Darmstadt, Germany
gupta@ipsi.fraunhofer.de

Abstract

The paper describes different approaches for checking the subsumption relation in GermaNet using database queries and subsequent manual analysis. The work was carried out in an object-oriented tool environment hosting the GermaNet data. Finally there is a brief note comparing GermaNet coverage with that of Duden dictionaries.

1. Introduction

The context of the work presented here was a study for Bibliographisches Institut & F.A. Brockhaus (BIFAB), publishers of Duden dictionaries; the main purpose of the study was to subject GermaNet (Hamp and Feldweg, 1997; Kunze, 2000) to close scrutiny by examining semantic relations in terms of consistency and subsequently to compare coverage of GermaNet with Duden dictionaries. The relations we focused on were the generic hierarchical relation expressed by hyponymy/hyperonymy and its synonymous variant for verbs (troponymy/troponymyOf), since this is the fundamental relation in GermaNet (Kunze, 1999). The tool hosting GermaNet for this work was the TerminologyFramework system, briefly described below in subsection 2.1. Various approaches to investigating the subsumption relation were adopted:

- formal consistency checks using database queries and manual analysis of results
- manual inspection of the non-overlapping parts of a semantic field and the corresponding concept hierarchy in GermaNet
- manual inspection of subsumption links reachable from a 10% sample of the denotation strings of GermaNet which also belong to the single volume Duden dictionary (DUDEN, 2000b) as lexical entries.
- top concepts were identified and analysed.

The starting point for the manual tests was rigid or strict subsumption: concept A is subsumed by concept B iff all instances of A are also instances of B.

Duden made their dictionary material available to us in machine-readable format thus enabling us to also compare the coverage of GermaNet with both the 10 volume (DUDEN, 2000a) and the single volume Duden dictionaries.

2. Formal Consistency Checks with Queries in Terminology Framework

2.1. Terminology Framework

TerminologyFramework (henceforth TFw) is a general purpose tool for representation and maintenance of thesaurus-like structures, ranging from conventional thesauri to the published CyC upper ontology or lexical databases such as WordNet (Fischer, 1998). GermaNet was imported into a TFw application, using an identical schema previously developed for investigating WordNet (Fischer, 1997). This import generated an object-oriented representation of GermaNet including persistent storage. Its contents could be inspected with tools including a frame to slot to value list view and graphic view (described in Möhr and Rostek (1993)) and investigated by means of database queries. The import turns every synset into an object (known in TFw as a **concept**) and the synset elements are represented as **terms**, which are denotation objects with disambiguated denotation strings. One of the advantages of TFw is that it allows for computable relations such as the transitive closure of the subsumption relation.

2.2. Formal Consistency Checks

A broad range of formal checks for redundancy and consistency in WordNet had already been devised and described by Fischer (1997). We restricted ourselves to consistency checks with respect to the subsumption relation in GermaNet. Fischer's investigation employed three distinct queries relevant to this relation:

1. Are there opposed concepts where one subsumes the other?
2. Are there opposed concepts which have a common subconcept?
3. Are there examples where the commutativity of subsumption and opposedness does not hold?

We understand subsumption not only as a relation that holds directly but also indirectly between concepts (as a result of the transitivity of this relation), which means that these questions presuppose the availability of the transitive closure of hyponymy/hyperonymy in GermaNet. The 'opposed' relation is defined thus: two concepts are opposed (or synonymously 'antosemous') if at least two of their terms are antonyms. Therefore a further computable semantic (concept-concept) relation is induced from a lexical (term-term) relation and both computable relations are prerequisites for the check. If we consider the third query, we need to explain what is meant by commutativity of subsumption and opposedness. Fischer (1997) defines this as follows:

For each concept *c:* If *antosem(c)* is not empty, then the equation *hypernym(antosem(c)) = antosem(hypernym(c))* or set inclusion in one direction or the other should hold.

All three rules may be justified by a concept model with feature inheritance, assuming that opposed concepts necessarily have some kind of contradictory feature which must not be inherited simultaneously by a more specific concept, otherwise this would lead to an oxymoron (e.g. bittersweet). We use this example, however, to illustrate that it is by no means impossible for language to creatively violate this logical inheritance rule. These three questions are therefore best seen as a heuristic to detect on the one hand cases which entail errors and on the other hand cases which invalidate the generality of the rule.

We did not consider the last of the three questions concerning commutativity, but concentrated instead on the first two. The retrieval results are discussed below in subsections 2.3. and 2.4.

2.3. Does GermaNet contain opposed concepts where one subsumes the other?

This query posed to the classes of verb and adjective concepts returned no hits, but when posed to the class of noun concepts it returned three noun concept pairs, illustrated by the three figures below:

- *Ziegenbock* (male goat) and *Ziege(2)*[1] (goat, in the generic rather than female sense),
- *Subjekt(2)* (subject in the sense of a living being) and *Objekt(2)* (object in the sense of living being)
- *Titelverteidiger* (title holder) and *Herausforderer* (challenger)

Figure 1 illustrates the case of *Ziegenbock*. Here we see that the antonymy relation has been falsely assigned between the generic and the male form; there should be a link showing antonymy stretching from left to right in the figure, that is from the term *Ziegenbock* to the term *Ziege* of the concept *Ziege* in its female sense. This case appears to be the result of an incorrectly assigned pointer due to homographs; we can only speculate as to whether inappropriate tools or limited views used in linking concepts by the lexicographer are the source of the error here.

Figure 2 illustrates the case of *Subjekt (2)* and *Objekt(2)*. The opposed relation between *Subjekt(2)* and *Objekt (2)* induced by antonymy is clearly false. We suggest that another pair of concepts, 'namesakes' to the given pair – *Subjekt (1)* and *Objekt (1)*, both in the grammatical sense, should be linked as opposed concepts. The 'namesakes' relation is a computable TFw relation which links concepts with homographic denotation strings.

The case of *Titelverteidiger* (title holder) and *Herausforderer* (challenger), illustrated in Figure 3 below, leads to a different diagnosis. We maintain that the hyponym link

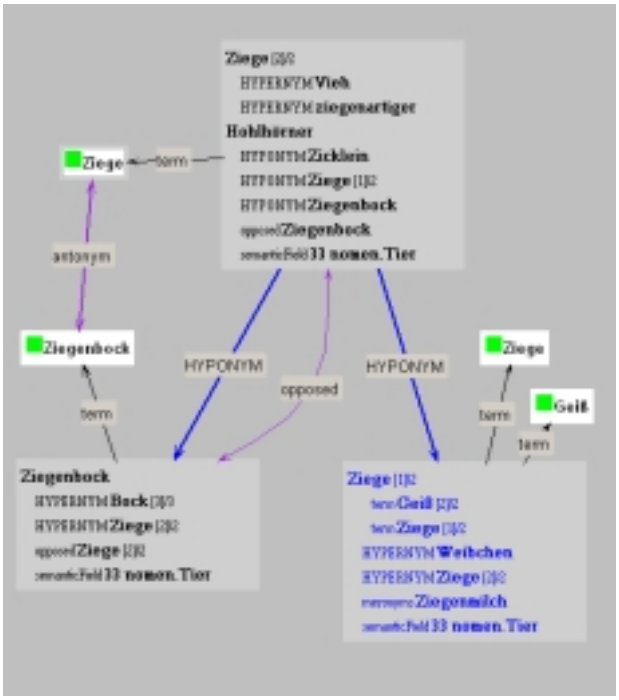


Figure 1: Faulty antonymy target

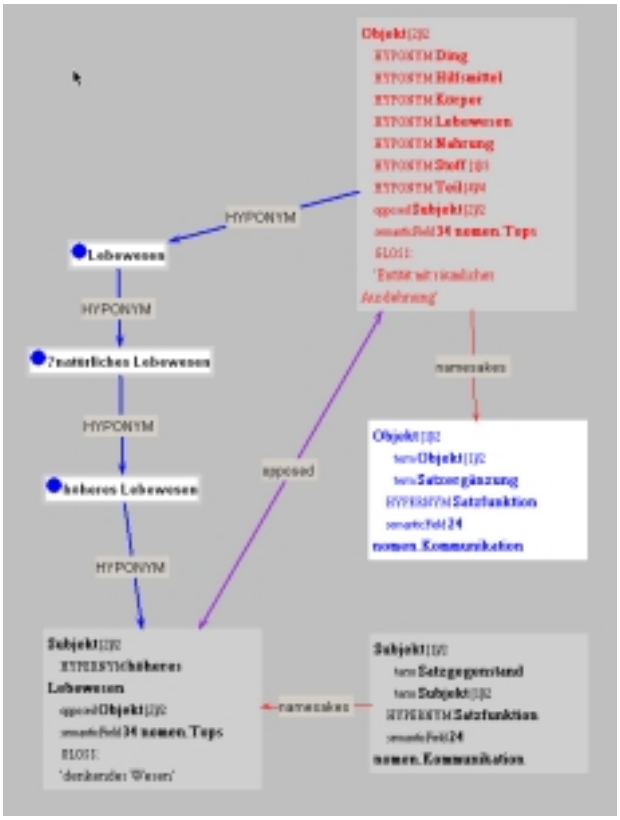


Figure 2: Faulty antonymy pair

between *Herausforderer* and *Champion* (champion) is incorrect, since not every champion is a challenger. [2]

[1] The number after the word denotes a homograph counter generated by TFw; the figures also show the number of homographs for each respective homographic string, separated by a '/'.

[2] Note that an antonym link is missing between the gender-inclusive forms *HerausforderIn* and *TitelverteidigerIn*.

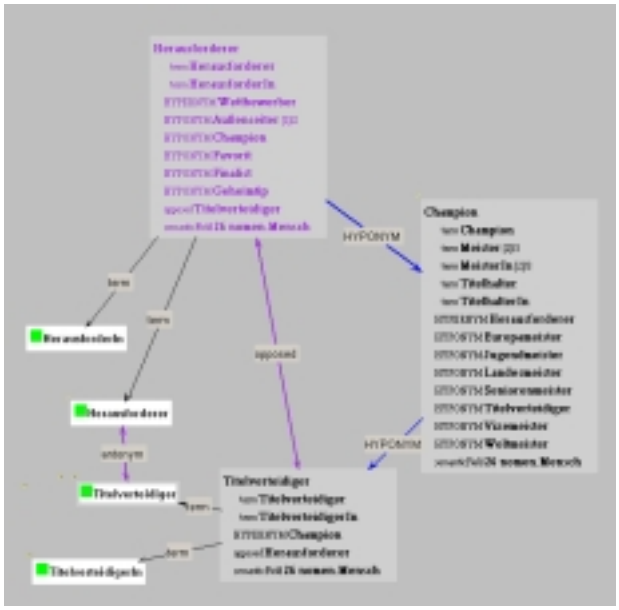


Figure 3: Faulty hyponym link between *Herausforderer* and *Champion*

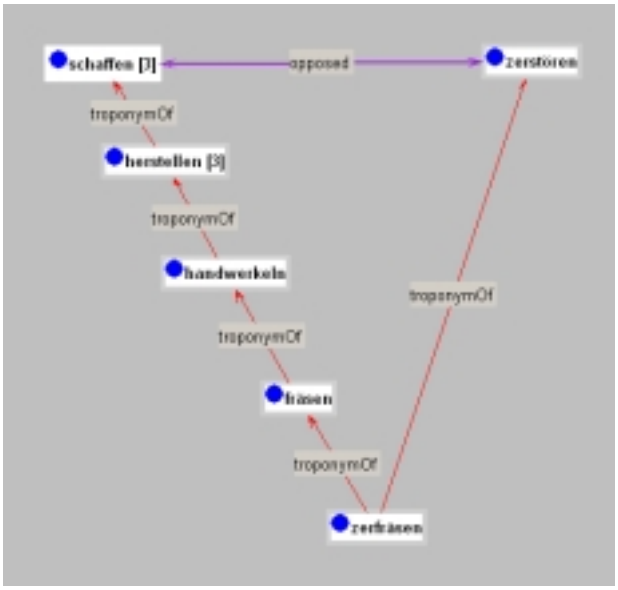


Figure 4: Strict versus defeasible hyperonymy

2.4. Does GermaNet contain opposed concepts with a common hyponym?

The query posed to the class of verb concepts returned two verb concept pairs, illustrated by the two figures below:

- *schaffen (3)* (in the sense of to create) and *zerstören* (to destroy); their common troponyms are *zersägen* (to saw up), *zerkochen* (to overcook or cook to a pulp), and *zerfräsen* (to mill to pieces)

- *nehmen (1)* (in the sense of to take something) and *geben (2)* (in the sense of to give something); there are 8 common troponyms including e.g. *tauschen* (to exchange something for another thing) and *dealen* (in the sense of dealing e.g. drugs).

This query returned results which are not indicative of incorrect pointer assignment but rather raise non-trivial questions about the nature of the subsumption relation or the antonymy relation in GermaNet.

The figure shows that *zerfräsen* is simultaneously a hyponym of verb concepts denoting creation and destruction. At first sight this seems counterintuitive. How can we account for this phenomenon? The hyponymy relation of *zerfräsen* and *zerstören* is obviously correct and is a rigid subsumption link. Looking at the left hand side of the figure, we check the link from *zerfräsen* to *fräsen*. This we deem to be acceptable as a rigid subsumption link, if we *fräsen* means to use a milling tool or mould in its neutral sense irrespective of its creative or destructive effect. If, however, we proceed from that concept node upwards to *schaffen (3)* we leave the neutral sense of *fräsen* and adopt a sense in which a creative or non-destructive use of the tool is implicit. It therefore follows that we have given the concept node *fräsen* two different meanings, and therefore according to the general WordNet philosophy we should split the node into 3: *fräsen(neutral)* , *fräsen (constructive)* and finally *fräsen (destructive)*, which already exists as *zerfräsen*.

Another possible remedy is to differentiate between strict and defeasible (non-strict) subsumption; the link between *zerfräsen* and *fräsen* would be strict whereas the link between *fräsen* and its direct superordinate *handwerken* or its indirect superordinate *schaffen (3)* is non-strict, i.e. in most cases the use of a milling tool or mould is constructive. Introducing a new subsumption relation type to the WordNet software, however, is likely to be difficult in contrast to TFw. This would entail checking all subsumption links for their type. Note that we cannot assume transitivity for the concatenation of strict and non-strict subsumption links.

A radically different diagnosis and remedy spring to mind when considering the case of Figure 5. At first sight the constellation appears to be acceptable, thus disproving the general validity of the rule. Our intuition may tell us that *tauschen* implies simultaneous acts of giving and taking and thus even the conjunction of the superordinates *nehmen* and *geben* seems plausible. On closer inspection, however, we see that a *tauschen* act implies the taking of **one item** in exchange for **another**, which means that the act of exchange consists of two simultaneous (or more probably) consecutive acts of giving X and taking Y where X and Y are not identical. The opposition of the concepts 'giving' and 'taking', however, obviously implies that the object of both is the same otherwise there would be no opposition. For example, teaching linguistics is not the 'opposite' of learning mathematics. What does the antonym or opposed link actually mean? (cf. Woods (1991, pp. 54ff)) If it means every act of giving is opposed to every act of taking, in the same way as every sweet object is opposed to every savoury object then the opposed link is faulty. If it means that for every giving act there exists a taking act which is opposed, then the rule implicit in the query does not have

general validity! This demonstrates the inconsistent use of the antonym/opposed link. Instead of the troponymOf links between *tauschen* and *nehmen* and *tauschen* and *geben* we propose a pair of 'entails' links, which would show that exchanging entails both giving and taking.

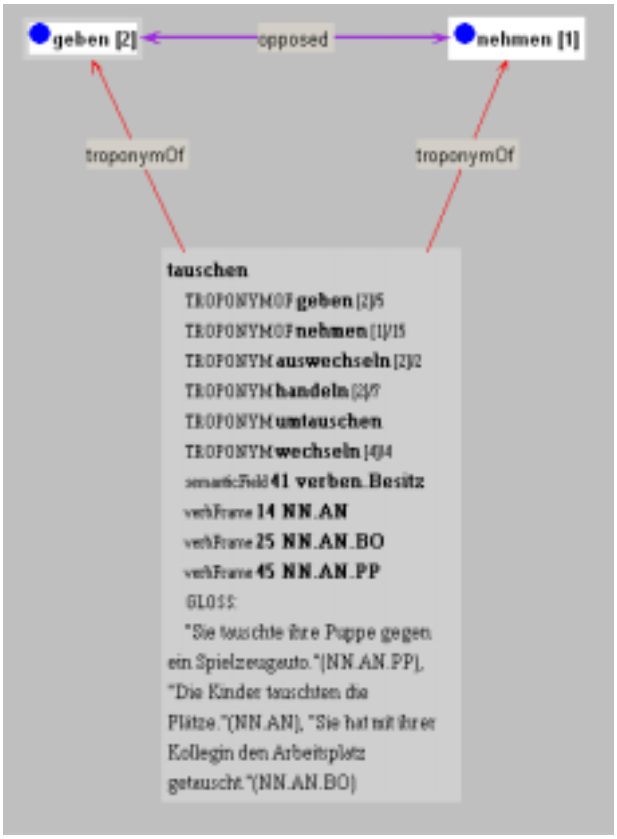


Figure 5: *Geben (2)* and *nehmen (2)* only opposed with a common object

Posing the query to the class of adjective concepts returned one concept pair, *farbig (2)* (in the non-racial sense of coloured) and *farblos* (colourless) with the common hyponym *falb* (dun, as applied to horses). This constellation contains a highly questionable hyponym link between *fahl* (pale) and *farblos* (colourless).

Posing the same query to the class of noun concepts also returned a single concept pair, *Vermögen* (property) and the non-lexicalised concept *?negativer Besitz* (negative ownership). In this case two highly questionable hyponym links exist, on the one hand between *Zins* (interest) and *Vermögen* or *Finanzen* (finances) and on the other between *Verzugszins* (interest payable on arrears) and *?negativer Besitz*.

In concluding this section, we note that retrieval results for both kinds of questions did not invalidate the implicit heuristic rules.

3. Semantic fields and hyponymy

According to GermaNet documentation (http://www.sfs.nphil.uni-tuebingen.de/lsd) , the division of GermaNet into semantic fields served an organisational purpose in that a field corresponds to a data file for editing by lexicographers. It was nonetheless interesting to investigate to what extent the semantic fields did in fact contain the expected content and to this end we looked at the hyponymy relation in cases where an available top concept corresponded to a semantic field label. Wherever this proved to be the case, we would expect all hyponyms to be members of that semantic field. Those elements **not** in the intersection of both sets demanded closer inspection, for instance with regard to which hierarchy they should actually belong to. This is another fruitful method for delimitation of the set of hyponym links to be checked manually.

Among others, the semantic field *nomen.Tier* (noun animal) was examined in tandem with the concept *Tier* (animal) and queries led us to obtain the following results:

- the concept *Tier* has 2049 hyponyms.
- the semantic field contains 2086 elements.
- only one concept *Pute* (turkey in its food sense) is an indirect hyponym of *Tier* but not a member of the semantic field *nomen.Tier* and instead belongs to the semantic field *nomen.Nahrung* (noun food). Here we have a clear-cut case of 'animal grinding' (Briscoe et al., 1995), in which a count noun (animal) becomes a mass noun (food). It might therefore be useful to assign a different kind of link which is applicable to the grinding operation.
- 38 concepts are in the semantic field but are not hyponyms of *Tier*; almost half of these consist of mythical beasts. The remainder include borderline cases such as single-celled beings, bacteria and microorganisms. If we maintain that mythical beasts such as *Einhorn* (unicorn) are animals irrespective of their real existence, then they should by rights also be hyponyms of *Tier*. There are also concepts such as *Männchen* (male animal) and *Weibchen* (female animal) which should properly be classed as animals.

4. Missing links

Formally speaking, top concepts are those which have no superordinates. GermaNet contains 500 such formal top noun concepts, but it should be noted, however, that of these 500 concepts only 125 are what we would term genuine top concepts in that they also have hyponyms, the remaining 375 are therefore isolated having neither superordinates nor hyponyms. For verbs there are 125 genuine tops and 94 isolated concepts and for adjectives there are 34 genuine tops and 246 isolated concepts. This points towards the transitional status of these concepts – GermaNet is, after all, a work in progress. The large number of remaining top concepts for nouns and verbs in particular is therefore arguably due to missing structure at this highest level. For example, a number of the genuine tops should either be hyponyms of other tops or hyponyms of new, more general concepts – *Wurstware* (sausages) is a top and is not a hyponym of *Nahrung* (food) as would be expected and tops such as *Arbeitszeit* (working hours), *arbeitsfreie Zeit* (leisure time) are not linked to the possible concept of time interval. Another approach to finding missing antonym links is to run

the third query mentioned above in subsection 2.2. and discussed for WordNet in Fischer et al. (1996, p. 253) and Fischer (1997, p. 28).

5. Manual evaluation of generic links in a sample of GermaNet

This section summarises results of an investigation of a sample of GermaNet with regard to the correctness of the hyponymy relation. The basis of the sample was a list of GermaNet synset elements which also appear in the single volume Duden dictionary as lexicon entries. Starting with the ninth list entry and subsequently every tenth entry was extracted from a list of adjectives, nouns and verbs thus providing us with a 10% sample of GermaNet. A total of 3511 hyperonym links were examined and classical tests for strict hyponymy were applied. We distinguished between correctly assigned, doubtful [3] and incorrectly assigned hyperonymy. Results were as follows:

- out of 519 verb denotation strings we derived 914 hyperonym links , 89% were deemed to have correctly assigned hyperonymy, 4% were doubtful and 7% were incorrect,
- out of 396 adjective denotation strings we derived 489 hyperonym links, 92% were correct, 2.5% were doubtful and 5% were incorrect,
- out of 1664 noun denotation strings we derived 2108 hyperonym links, 96.6% were correct, 1.2% were doubtful and 2.2% were incorrect,
- of a total of 2579 denotation strings for all 3 GermaNet word classes we derived 3511 hyperonym links, 94.1% were correct, 2.1% were doubtful and 2.1% were incorrect.

Some of the commonest errors were mistaken assignment of hyponym/troponym where a merge of concepts would be more appropriate because their terms are stylistic variants and therefore synonyms. For instance, the stylistic variants *pennen*, *knacken* and *ratzen* (to kip; colloquial for to sleep) are deemed to be troponyms of *schlafen* (to sleep) rather than as what Cruse (1986) regards as 'cognitive synonyms'. The assignment of hyponymy seemed on occasions to be based on morphological factors rather than semantic ones (e.g. *Fahrgast* (passenger) as a hyponym of *Gast* (guest).

6. Coverage of GermaNet compared with Duden dictionaries

It is a truism that both the single volume and 10 volume Duden dictionaries have wider coverage than GermaNet, with around 100,000 entries and 200,000 entries respectively so it is arguably more interesting to look at what is to be found in GermaNet but not in single or multi-volume reference works rather than to simply enumerate what is in the dictionary but not in GermaNet. GermaNet contained [4] a total of 41359 entry strings, of which 25798 appear in both GermaNet and the single volume Duden. A total of 15561 entry strings were to be found in GermaNet but not in the single volume Duden. 28862 entry strings appeared in both GermaNet and the 10 volume Duden and 12497 entry strings were present in GermaNet and not in the 10 volume Duden.

A number of groups in GermaNet and in neither of the Duden dictionaries can be identified as follows:

- gender-neutral terms denoting roles (e.g. *AntifaschistIn* (anti-fascist))
- very specific specialised language (e.g. terms from a biological taxonomy)
- selected compounds; compounding is highly productive in German and therefore criteria for their selection and inclusion are dependent on e.g. frequency, corpus evidence
- orthographic variants
- misspellings

GermaNet contains 1869 gender-neutral terms denoting roles which are not present in the form with an upper case 'I' in either the single or 10 volume Duden, but feminine forms are to be found if the upper case I is eliminated by a normalisation to lower case letters. GermaNet appears to contain an exhaustive biological taxonomy (despite claims for inclusion on the basis of corpus frequency), so on inspection of the 2049 hyponyms of *Tier* (animal) and the 189 hyponyms of *Pflanze* (plant), 1043 animal hyponyms and 1119 plant hyponyms are present that are not to be found in the 10 volume Duden. The difference between what is present in GermaNet and in dictionaries raises important questions for lexicographers – for instance, which criteria should be employed for inclusion of compounds, which can in any case never be completely covered due to the productivity of compounding. Also, how subjective frequency decisions made by lexicographers are and to what extent the use of balanced corpora can contribute to lexicography.

[3] Some of the links that we deemed in this analysis to be merely doubtful (such as the link between *tauschen* (to exchange) and the *geben* (to give) and *nehmen* (to take) pair) were deemed incorrect after the formal checks described in subsection 2.4..

[4] We used version 3.0, current as of 22.01.01

7. Acknowledgements

I am grateful to Dietrich Fischer for his insights and encouragement in the writing of this paper. Constructive discussion with him proved invaluable. I would also like to thank Lothar Rostek for initiating the study in the first place and for playing a major role in setting up the working environment and the scenario.

8. References

Ted Briscoe, Ann Copestake, and Alex Lascarides. 1995. Blocking. In Patrick Saint-Dizier and Evelyne Viegas, editors, *Computational Lexical Semantics*, pages 273–301. Cambridge University Press, Cambridge.

D A Cruse. 1986. *Lexical Semantics*. Cambridge University Press, Cambridge.

DUDEN. 2000a. *Das große Wörterbuch der deutschen Sprache*. Bibliographisches Institut & F.A. Brockhaus A.G., Mannheim.

DUDEN. 2000b. *Der Duden, 12 Bde., Bd.1, Duden Die deutsche Rechtschreibung, neue Rechtschreibung*. Bibliographisches Institut & F.A. Brockhaus A.G., Mannheim.

Dietrich H. Fischer, Wiebke Möhr, and Lothar Rostek. 1996. A modular, object-oriented and generic approach for building terminology maintenance systems. In Christian Galinski and Klaus-Dirk Schmitz, editors, *TKE '96:Terminology and Knowledge Engineering*, pages 245–258, Frankfurt a.M. INDEKS Verlag.

Dietrich H Fischer. 1997. Formal redundancy and consistency checking rules for the lexical database WordNet 1.5. In Vossen et al. (Vossen et al., 1997), pages 22–31.

Dietrich H Fischer. 1998. From Thesauri towards Ontologies? In Widad Mustafa el Hadi, Jacques Maniez, and Steven A. Pollitt, editors, *Structures and Relations in Knowledge Organisation:Proceedings of the Fifth International ISKO Conference*, volume 6, pages 18–30, Lille, France. Ergon Verlag.

Birgit Hamp and Helmut Feldweg. 1997. GermaNet - a Lexical-Semantic Net for German. In Vossen et al. (Vossen et al., 1997), pages 9–15.

Claudia Kunze. 1999. Semantics of Verbs within GermaNet and EuroWordNet. In V. Kordoni, editor, *Proceedings of the ESSLLI-99 Workshop on 'Lexical Semantics and Linking in Constraint-Based Theories'*, pages 189–200, Utrecht.

Claudia Kunze. 2000. Extension and use of GermaNet, a lexical-semantic database. In *Proceedings of LREC 2000 2nd International Conference on Language Resources and Evaluation*, Athens.

Wiebke Möhr and Lothar Rostek. 1993. TEDI: An Object-Oriented Terminology Editor. In Klaus-Dirk Schmitz, editor, *TKE '93:Terminology and Knowledge Engineering*, pages 363–374, Frankfurt a.M.

Piek Vossen, Gert Adriaens, Nicoletta Calzolari, Antonio Sanfilippo, and Yorick Wilks, editors. 1997. *Automatic Information Extraction and Building of Lexical Semantic Resources*. Association for Computational Linguistics, 12 July 1997.

William A. Woods. 1991. Understanding subsumption and taxonomy: A framework for progress. In John Sowa, editor, *Principles of Semantic Networks: Explorations in the Representation of Knowledge*, pages 45–94. Morgan Kaufmann, San Mateo, CA.

Evaluation of GermanNet: Problems Using GermaNet for Automatic Word Sense Disambiguation

Jahn-Takeshi Saito, Joachim Wagner, Graham Katz, Philip Reuter, Michael Burke, Sabine Reinhard

Institute for Cognitive Science
University of Osnabrück
49074 Osnabrück
Germany

Abstract

WordNets such as GermaNet have been frequently used as an inventory of word-senses for word-sense disambiguation tasks. In the work described here we evaluate the adequacy of GermaNet for this task. That is we attempt to determine the degree to which GermaNet provides an adequate inventory of senses for word-sense annotation of running text. Our findings were on the whole very encouraging. GermaNet provides an appropriate sense for 83 % of the content words in our texts. More interestingly, an error analysis showed that simple morphological processing could significantly improve coverage.

1. Introduction

The use of WordNet for sense tagging of English is by now an established research program (Miller, et.al 1994; Resnik 1998; Landes, Leacock & Tengi 1998). With the advent of WordNet-style lexical resources for languages other than English (Bloksma, Díez-Orzas & Vossen 1996) the application of these resources, to sense-tagging for these languages is a natural evolution. A number of questions arise in this context, however. While the original WordNet has been used with success for English, there is no guarantee that this experience generalizes to other WordNets for other languages. Both the language itself and the particular WordNet developed for it may present problems that were not present in the WordNet/English case. Our goal here was to evaluate how useful GermaNet is as a resource for word sense tagging for German.

Our task, then, was to annotate a corpus of German text using GermaNet and to determine how close to the ideal of providing an appropriate sense tag for all content words in the corpus GermaNet is. This is of interest both as an evaluation of GermaNet itself, and also because German and English differ in ways that, *a priori*, might indicate that German would be a difficult language to sense-tag (Hamp & Feldweg 1997). German has, for example, highly productive word-formation processes and a rich derivational morphology.

In form, however, our work was very similar to that done for English by Landes, Leacock & Tengi (1998) in that we simply set out to manually disambiguated words in a corpus, tagging each appearance of a content word in the corpus. In contrast to their work, we developed our own (German language) corpus and used GermaNet as our repository of word senses. Additionally, work was separate from the development of GermaNet and we did not have contact with GermaNet lexicographers.

2. GermaNet

GermaNet is a lexical-semantic net based on the WordNet example (Kunze & Wagner 1999a). It is intended to cover the basic vocabulary of German. Although GermaNet relies on the design principles and shares the same database structure as the Princeton WordNet (Miller 1990), it is build from scratch and features some modifications. In contrast to WordNet, GermaNet includes non-lexicalized *artificial* concepts to fill lexical gaps (e.g. to provide the missing antonym for *thirsty*) and to avoid unjustified co-hyponomy. Additionally, cross-classification of concepts, which is seldom used in WordNet, is an essential feature of GermaNet, and regular polysemy is integrated via a special relation between synsets. There are also some particular differences with respect to the way parts of speech are handled. Adjectives in GermaNet, for example, are hierarchically structured (in contrast to a clustering approach in WordNet). It wasn't clear that any of these differences affected the usefulness of GermaNet for sense-tagging, however.

More important was GermaNets coverage. Although GermaNet is comparable in size to WordNet, it is significantly smaller, as indicated in Table 1.

	GermaNet	WordNet 1.7
Noun	27824	74488
Verb	8810	12754
Adjective	5141	18523
Adverb	2	3612
Total:	41777	109377

Table 1. GermaNet vs. WordNet

Although GermaNet has been integrated into EuroWordNet (Kunze & Wagner 1999b), the version we used for our research was the stand-alone GermaNet.

3. The annotation task

As a preliminary to the development of an automatic sense tagger for German we hand-tagged eleven small German texts. We used these hand tagged texts to evaluate the applicability of GermaNet to large-scale

sense tagging applications. The procedure we used for annotation was fairly straightforward. We automatically lemmatized the words and tagged them for part of speech using the Stuttgart TreeTagger. To actually carry out sense tagging, we developed a software tool for presenting words in texts along with their GermaNet synsets, which was used by five annotators to annotate the texts. The texts were annotated on a word-by-word basis, with each token that had been tagged either as a verb, a noun or an adjective presented for word-sense tagging. For words that could not be annotated with GermaNet synsets, the problem that the word appeared to pose was noted by the annotator, if one was apparent. These error-annotations were used to classify the types of words that presented difficulties for sense-tagging using GermaNet synsets.

3.1. Corpus preparation

As there is not yet a standard representative German corpus, we choose to develop our own corpus. The corpus consisted of eight short excerpts from novels for children and young people and three articles taken from German newspapers. The total number of words in our corpus was 5625 and the individual subcorpora varied in size from 257 to 1021 words.

The entire corpus was both lemmatized and tagged for part of speech by the IMS TreeTagger (Schmid 1994). These lemmata were then used to automatically compile a list of GermaNet synsets for each token in the corpus that appeared in GermaNet. For each lemma, the complete set of GermaNet synsets associated with the lemma by GermaNet was stored alongside the lemma. The POS information was **not** used in this step for filtering, so as to exclude this as a source of error. As indicated in Table 2, GermaNet assigned a synset to more than 90% of the content words (noun, verb or adjective tagged words) in the texts. Strikingly, the percentage of content words not assigned an appropriate synset by GermaNet is lower for the newspaper corpora (about 80%) then for the childrens fiction corpora (about 85%).

3.2. Corpus Annotation

For purposes of annotation, the eight short-novel corpora were split up randomly into 24 pieces which were recombined into equal-sized subcorpora and distributed among our five annotators. The pieces were systematically permutated in order to minimize the influence of inter annotator differences. After annotation was complete the pieces were reordered, so that statistics could be obtained on a per corpus basis. At a later stage the newspaper subcorpora News 1, News 2, and News 3 were annotated. Although the annotation procedure was the same, these subcorpora were annotated by a single annotator.

The actual annotation was carried out as follows. The five annotators – all native speakers of German – were provided with a software tool and a set of files to be tagged. The software tool (see fig. 1) presented the annotator with each occurrence of a lemma for which GermaNet provided synsets.

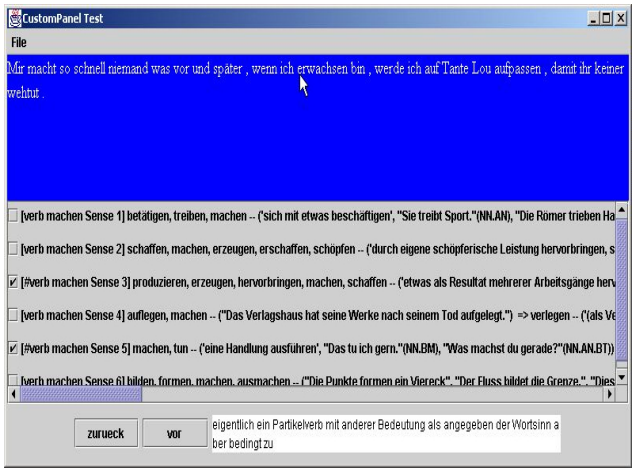


Figure 1. The TAZAN annotation tool

The annotator task was to mark the appropriate synset, if there was one. In addition to the textual context the word appeared in, i.e. the sentence, annotators were shown the set of synsets for the lemma and the basic characterization provided by GermaNet for these synsets. These contained brief descriptions of the synset, examples of typical uses of that sense of the word and an indication of where the synset was located in the GermaNet hierarchy. For verbs the syntactic frame associated with the sense was also indicated. The synsets were presented to the annotator grouped by POS. In choosing a synset, annotators also implicitly indicated what they took to be the correct POS for the word in contexts.

For the lemma *essen*, for example, the following information was presented, with three noun senses and one verbal sense.

[nomen essen Sense 1] Essen, Mahl, Mahlzeit --
('Einnahme von Speisen')
[nomen essen Sense 2] Gericht, Speise, Essen --
('Speise, die für eine Mahlzeit zubereitet ist') =>

Corpus	Word Tokens	Content words	Synset Assigned	Marked	Marked (of Assigned)
Fiction	4330	1770	1658 (93.7%)	1497 (84.6%)	90.3%
Newspaper 1	257	143	129 (90.2%)	124 (86.7%)	86.7%
Newspaper 2	474	206	179 (86.9%)	161 (78.2%)	89.9%
Newspaper 3	564	270	233 (86.3%)	205 (75.9%)	76.3%

Table 2: Quantitative Characterization of the Corpora and Annotation Results

Nahrung, Nahrungsmittel, Lebensmittel, Esswaren, Eßwaren*o, Essen, Speisen
[nomen essen Sense 3] Nahrung, Nahrungsmittel, Lebensmittel, Esswaren, Eßwaren*o, Essen, Speisen => Objekt -- ('Entität mit räumlicher Ausdehnung')
[verb essen Sense 1] essen, futtern*s, nehmen -- ('etwas zu sich nehmen', "Er isst kein Fleisch."(NN.AN), "Er futtert wie ein Scheunendrescher."(NN.BR), "Sie nimmt viel Flüssigkeit zu sich."(NN.AN.PP), "Die Kinder futtern fleißig Schokolade."(NN.AN.BM) "Heute abend werde ich warm essen."(NN.BM)) => verzehren -- ('Ein Lebensmittel essen oder trinken, Perspektive auf Lebensmittel', "Auf der Weihnachstfeier haben die Mitarbeiter zehn Kilo Fleisch verzehrt.", "Sie verzehrte ihr Gemüse ohne Appetit.")

The annotators were also encouraged to use the GermaNet browser to locate additional information about a synset if a decision was difficult.

To annotate, the annotator simply selected (via check box) the appropriate sense(s) for the word as used in the context presented. They were able to move freely forwards and backwards through the corpus and to change their choice of synset at any time. The task was not an easy one. To fully annotate even one of our 24 small subcorpora took our annotators approximately an hour of annotation time. Typically, however, our annotators divided up the task into a number of sessions.

Note that annotators were instructed to mark all synsets considered appropriate. That means that the annotator could mark more than one of the senses GermaNet assigned to the word or reject all of them. This means that words which were not assigned at least one GermaNet synset were not presented for tagging at all. As indicated in the sixth column of Table 2, this was typically around 10% of the content words.

3.3. Results of annotation task

The results of our annotation exercise are indicated in the final columns of Table 2. This column indicates the percentage of the total number of content words (NVA tagged words) for which an annotator marked at least one of the supplied senses as correct and the percentage of the total number of words assigned a synset by GermNet for which at least one of the synsets assigned was marked by an annotator as being appropriate. This is a raw measure of how well GermaNet could be used to sense tag our corpora. That is, in about 90% of the cases, if a word appears in GermaNet, then the annotators found that GermaNet provided an appropriate sense for the word as used in the corpus. While not disappointing, the numbers may seem low. In fact they are misleadingly low, as a significant proportion of these errors are not due to GermaNet at all. In section 4 we will discuss these error factors extensively.

3.4. Inter annotator agreement

An important question, however, was the degree to which the judgement of our annotators varied. We made provision for evaluating inter annotator agreement by having all the annotators tag one small subset of the short novel corpus. This subcorpus contains 431 tokens and was annotated by all five annotators. Only 170 of these tokens were assigned a list of synsets by the GermaNet. So there were 170 points the annotators could disagree on. To evaluate inter annotator agreement, we looked at whether for each of these 170 tokens any synset was marked or not by the annotators. The number of tokens that were not marked as having any acceptable GermaNet assigned synset is shown in Table 3. All five numbers are in the 95% intervall [35, 57] of the binomial distribution with n = 170 and p = 46.0 / 170 = 0.271.

Annotator	1	2	3	4	5
Token with no synset marked	40	44	56	50	40
Mean	46.0				
Variance	38.4				
Standard deviation	6.2				

Table 3: Basic statistics of annatation

It is not, of course, correct to infer from this that the annotators agree on which tokens to mark. To evaluate the more narrow question of whether our annotators agree on this we compared our annotators pairwise. Table 4 shows how many tokens can be counted in the union and intersection of two annotators' annotation records filtered for tokens that have no marked synset and in which each token was prefixed with a unique token ID. The size of the intersection gives the number of tokens that they agree on and the difference to the size of the union gives the number of tokens they disagree on. If, for example, annotator 1 and 2 completely agreed, the number of tokens would be max(40,44) = 44 in the union and min(40,44) = 40 in the intersection. If they disagreed as often as possible, the numbers would be 40+44 = 84 and 0. Table 4 gives these numbers, with the possible ranges in square brackets. The numbers seem to show quite good agreement.

A way of measuring inter annotator agreement is provided by Cohen's (1960) kappa statistic. This measure indicates the degree to which the observed agreement rate differs from chance, and is given by:

$$\kappa = \frac{P_a - P_e}{1 - P_e}$$

where P_a is the observed agreement rate and P_e is the expected chance agreement. Numbers above 0.80 are generally considered to give evidence for a good agreement, whereas numbers below 0.67 indicate poor agreement (Carletta 1996). Our κ values – indicated in the final column of Table 4 – are between or even above these standard values, indicating acceptable agreement.

We did not analyze agreement of polysemy judgements, that is, agreement on what sense should be assigned to which word (c.f. Veronis 1998), because they are irrelevant to our study. Furthermore, token counts per type are too small to get significant results. It is important to keep in mind that we were primarily interested in whether GermaNet is a rich enough lexical resource, not with whether the annotators agreed exactly on how to use it.

Annotator pair	Token in Union	Intersection	κ
(1, 2)	48 [44, 84]	36 [0, 40]	0.81
(1, 3)	58 [56, 96]	38 [0, 40]	0.71
(1, 4)	51 [50, 90]	39 [0, 40]	0.82
(1, 5)	45 [40, 80]	35 [0, 40]	0.84
(2, 3)	61 [56, 100]	39 [0, 44]	0.69
(2, 4)	54 [50, 94]	40 [0, 44]	0.79
(2, 5)	50 [44, 84]	34 [0, 40]	0.75
(3, 4)	62 [56, 106]	44 [0, 50]	0.75
(3, 5)	58 [56, 96]	38 [0, 40]	0.71
(4, 5)	55 [50, 90]	35 [0, 40]	0.70
all five	64 [56, 107]	30 [0, 40]	0.75

Table 4: Inter annotator agreement

4. Error analysis

In order to analyze the quality and extent of GermaNet's coverage, then, we chose to further examine those tokens for which GermaNet should provide a synset, but for which no sysnset was marked by our annotators. These are the cases in which GermaNet fails to do its job. Our goal was to quantify this failure and to assess its most likely causes.

We take it to be the case that in the ideal case GermaNet would associate an appropriate sense for all occurrences of nouns, verbs, and adjectives. Given a perfect POS tagger a perfect lemmatizer, a perfect GermaNet and a perfect human annotator, every NVA-tagged word in our corpus should be marked by the annotators with at least one synset. (Perhaps *exactly* one would be more ideal; in our study we ignored this however. We were concerned that GermaNet be rich enough, not that it be too rich.)

In practice, of course, the results are not perfect. In the following we will discuss the degree to which our results deviated from the ideal. As we saw in Table 2, the number of content words which could be assigned a synset at all by GermaNet ranges from just over 83% to just under 94%. In only about 90% of the cases was one of the synsets assigned to a word by GermaNet marked as being the correct one by our annotators.

In fact, however, a large proportion of this error was introduced not by GermaNet, but by TreeTagger, which we used to lemmatize and tag our texts for part of speech. While errors in POS tagging could lead to suboptimal performance, POS tagging errors were fairly rare in our texts (as Schmid (1994) shows the tagger employed can reach an accuracy of about 97.5%). Furthermore, the kinds of errors that would be problematic in our task (mistagging of prepositions, adverbs or articles as nouns, verbs or adjectives) are the least common type. So POS tagging did not contribute significantly to the errors. Lemmatization errors, however, contribute significantly to the error rate, since every incorrectly lemmatized word resulted directly in an error: When a word is not properly lemmatized it is impossible for the human annotator to choose the correct synset, since this synset is not an available choice, as we have looked up the wrong word in GermaNet.

In order to evaluate GermaNet, then, we needed to classify our errors, so as to determine which errors were the result of GermaNet design or coverage problems and which, like lemmatization errors, were epiphenomenal.

4.1. The error classes

For purposes of our evaluation we took any NVA tagged token in our corpus to which no GermaNet synset was assigned to be an error, and we assigned each error occurrence to one of the following error classes: **lemma, particle**, **collocation, compound**, **derivation**, **auxiliary**, and **other**. The classification of errors was carried out by a single annotator (JS) using a Java-implemented GUI-tool. Each error was assigned to exactly one of the error classes. These classes were chosen because either they were a type of error that was particularly common, or because they were a type of error that the GermaNet developers had suggested might cause problems.

The error classes are described as follows:

Lemma. As mentioned, when a word is not properly lemmatized it is impossible for the human annotator to choose the correct synset, since it is not available for choice. An example of this kind of error is when the particle *mal* in "Mal wieder hat er es getan" is lemmatized as *malen*, the verb 'to draw'.

Particle. German seperable verbs, such as *vorschlagen*, contain prefixes which significantly alter the meaning of a verb (*schlagen* – "hit"; *vorschlagen* – "propose"). These verbs should be lemmatized as a single lexeme. Unfortunately in many contexts the prefix is not concatenated with the verb, as in:

Er *schlug* einen Kompromiss *vor*.
"He proposed a compromise."

This presents difficulties for lemmatizers. Very often the lemmatizer does not link the particle verb's root and prefix leading to a wrong lemmatized form, omitting the prefix (e.g. *schlagen* instead of *vorschlagen*).

Auxiliary. The verbs *sein* and *haben* (as well as certain modals and others) are also problematic. These verbs can be used simply as syntactic operators – auxiliaries – on the one hand, or as main verb on the other. As auxiliaries, there is a sense in which they should not be sense tagged (since they are not "open class"). In this group we mark those cases in which such a verb is not tagged but is recognized as being used as an auxiliary.

Strictly, speaking both **particle** and **auxiliary** errors can be thought of as lemmatization errors of a very specific type, and cannot really be attributed to GermaNet. In contrast to these we distinguished three types of errors that can be attributed to word-formation processes:

Collocation. Many words are used in a very specific sense in combination with other words (*ins Wasser fallen* to mean "cancelled", for example). In those cases in which the word to be tagged was recognized as forming part of a collocation, it was assigned to this class. While it is arguably not the task of a lexicon to account for collocations and idioms, we were interested in assessing the degree to which these are problematic.

Compound. Compounding – the formation of a new word from two or more existing words (for example

Errors class	Corpus			
	Fiction	News 1	News 2	News 3
Lemma	12.3	5.3	13.3	10.8
Particle	5.9	0	4.4	4.6
Auxiliary	25.3	21.1	22.2	21.5
Compound	11.5	31.1	11.1	32.3
Derivation	5.2	10.5	4.4	6.1
Collocation	2.2	5.3	2.2	1.5
Other	36.8	26.3	42.2	23.1
Total errors	269	19	46	62

Table 5: Distribution of errors by class and corpus (in percent)

Montagsauto) is a productive word formation process in German (as in English). As the sense to be associated with the compound is a fairly arbitrary function of the meaning of the constituent words (cf. Fanselow 1981), it is in principle difficult to provide appropriate synsets for words formed this way.

Derivation. The generation of nouns from verbs (for example *Vorbereitung* from *vorbereiten*) and the generation of diminutive forms (for example *Hündchen* from *Hund*) are productive process in German. These are somewhat more regular and might be accounted for by a GermaNet with sophisticated morphological processing (like that suggested by Kunze (1999) for particle verbs).

Finally there are the errors that fit into none of these classes:

Other. All other forms of derivation are covered by the "other coverage" default error class. The major component of this class is simply the set of words which are simply missing form GermaNet, i.e, those that should be and could be listed, but are not.

4.2. Results of error analysis

In Table 5 we present the distribution of the different type of errors by error class in each of our small corpora. It is clear there was significant variation across the corpora as to which error classes were predominant. The variation was particularly evident in the case of **lemma** and **compound** errors. The most significant class of errors was the **auxiliary** class. These were fairly uniform, accounting for between a quarter and a fifth of all errors in each of the corpora. The surprising fact that we noted in section 3, that the newspaper corpora appear to be better handled by GermaNet than the fiction corpus, gets a simple explanation: lemmatization-related errors were more pronounced in the newspaper corpus. In fact, looking only at non lemmatization-related errors, we see that the childrens fiction is, as we might expect, less error prone than the newspaper articles.

The newspaper corpora evidenced significantly more errors that were due to the use of productive morphology. The **compound** errors were the most prominent, particularly in the newspaper corpora, although was significant variation here as well. Other **derivation** errors, however, had a relatively small share. **Collocations** though they appear in most corpora, also play a minor role.

In Table 6 the distribution of errors by POS is displayed. It is obvious why **particle** and **auxiliary** errors would be limited to verbs, as they are verb-specific error types. More interesting is the fact that errors that could be attributed to productive morphology were essentially limited to nouns and adjectives. Essentially only nouns were involved in **derivation** errors, while for adjectives (other than **lemma** errors) essentially only **compound** errors were present

Error class	Part of Speech		
	Verb	Noun	Adjective
Lemma	3.5	23.7	13.9
Particle	13.2	0	1.3
Auxiliary	58.8	0	1.3
Compound	0	31.6	8.9
Derivation	1.8	17.1	1.3
Collocation	3.5	2.6	0
Other	19.3	25	73.4
Total errors	114	76	79

Table 6: Distributions of errors in Fiction corpus by class and part of speech (in percent)

5. Conclusion

Our results were very encouraging. On average 92% of the words which were tagged as verbs, nouns or adjectives were provided with at least one sense by GermaNet, and more than 83% were provided with at least one sense that was judged as the correct sense by our annotators. One of the major sources of error was, in fact, external to GermaNet: On average 15% of the content words were incorrectly lemmatized, leading to incorrect lookup. Additionally we found that many of the potential sources of coverage failure suggested by Hamp & Feldweg (1997) were indeed evident: productive morphological processes such as derivation and compounding as well as collocative uses of words accounted for a nearly 25% of the errors we noted. Particle verbs also presented problems for our annotators, as in some cases the verb was not lemmatized with its separable prefix. Clearly a more sophisticated lemmatizer could have eliminated some of these errors. In other cases productive combinations with main verbs gave rise to forms which were not covered by GermaNet. For nouns a predominant source of errors was the existence of a large number of nouns that were clearly derived via productive rules of derivation from verbs. These could, presumably, be looked up on the verbal hierarchy. Words formed via compounds were also a significant source of noun and adjective errors. Words that could not be properly tagged because they were used as part of a collocation accounted for only minority of the errors overall, however.

We also found that the effectiveness of GermaNet as used for the word-sense disambiguation task as well as the kinds of errors that were found was highly dependent on the variety of text to be disambiguated. This suggests that it is crucial that in WordNet evaluation both domain and text type be standardized, and that a variety of types be used.

Finally, many of the types of errors that we found were clearly German-language specific. This finding suggests that language-specific issues are quite important when evaluating the effectiveness of a particular WordNet and that simple cross-WordNet evaluation will likely lead to a incorrect evaluation of the value or coverage of a particular WordNet. With respect to GermaNet, our results suggest that sense-tagging using GermaNet, while quite good as it is, could be significantly improved by integrating additional morphological processing into the tagger. In particular, methods for dealing with compound words and derived words could lead to significant improvements.

6. References

Bloksma, L., P. Díez-Orzas, and P. Vossen, 1996. User requirements and functional specification of the EuroWordNet project. EuroWordNet (LE-4003), Deliverable D001, University of Amsterdam.

Carletta, Jean, 1996. Assessing agreement on classification tasks: the kappa statistics. *Computational Linguistics*, 22(2), 249-254.

Cohen, J., 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20, 37-46.

Dutoit, Dominique, Laurent Catherin, and Andreas Wagner, 1998. Specification of German and French Wordnets. EuroWordNet (LE4-8328), Deliverable 2D002.

Fanselow, Gisbert, 1981. *Zur Syntax und Semantik der Nominalkomposition – Ein Versuch praktischer Anwendung der Montague-Grammatik auf die Wortbildung des Deutschen*. Tübingen: Niemeyer.

Hamp, Birgit and Helmut Feldweg, 1997. GermaNet - a lexical-semantic Net for German. In: P. Vossen et al. (eds.), *Proceedings of the ACL/EACL-97 workshop Automatic Information Extraction and Building of Lexical Semantic Resources for NLP Applications*, Madrid, pp. 9-15.

Kunze, Claudia and Andreas Wagner, 1999. Integrating GermaNet into EuroWordNet, a multilingual lexical-semantic database. *Sprache und Datenverarbeitung*.

Kunze, Claudia (ed.), 1999. Final wordnets for German, French, Estonian, and Czech. EuroWordNet (LE-8328), Deliverable 2D014.

Kunze, Claudia and Andreas Wagner, 1999. The German Wordnet. EuroWordNet (LE-8328), Deliverable 2D014.

Kunze, Claudia, 1999. Semantics of Verbs within GermaNet and EuroWordNet. In: V. Kordoni (ed.), *Proceedings of the ESSLLI-99 Workshop on 'Lexical Semantics and Linking in Constraint-Based Theories'*, pp. 189-200.

Landes, Shari, Claudia Leacock, and Randee I. Tengi, 1998. Building Semantic Concordances. In: Christiane Fellbaum, (ed.), *WordNet: an electronic lexical database*. MIT. Chapter 8, pp. 199-216.

Miller, G., M. Chodorow, S. Landes, C. Leacock, and R. Thomas, 1994. Using a semantic concordance for sense identification. In *Proceedings of the ARPA Human Language Technology Workshop*, San Francisco, Morgan Kaufmann, pp. 240-243.

Miller, George A. (ed.), 1990. WordNet: An on-line lexical database. Special issue of *International Journal of Lexicography*, 3 (4).

Resnik, Philip, 1998. *WordNet and Class-Based Probabilities.* In: Christiane Fellbaum (ed.), *WordNet: an electronic lexical database.* Cambridge: MIT Press, p. 239-263.

Schmid, Helmut, 1994, Probabilistic Part-of-Speech Tagging Using Decision Trees. *Proceedings of International Conference on New Methods in Language Processing*, September 1994.

Véronis, J., 1998. A study of polysemy judgements and inter-annotator agreement. Programme and advanced papers of the Senseval workshop, 2-4 September 1998. Herstmonceux Castle, England.

Vossen, Piek (ed.), 1999. WordNet1.5 in EuroWordNet format. Deliverable D032D033/2D014 Part B1.

Glosses in WordNet 1.5 and Their Standardization/Consistency

(The Exercise for Balkanet)

Karel Pala and Pavel Smrz

Faculty of Informatics, Masaryk University
Botanicka 68a, 60200 Brno, Czech Republic
{pala,smrz}@fi.muni.cz

Abstract

In this contribution we present an analysis of selected WN 1.5 glosses and dictionary definitions from other resources -- we examined what is the internal (semantic) organization of the glosses and dictionary definitions, i.e. how reliably and systematically they follow the standard principles of building dictionary definitions. The results following from the presented analysis should be applied in building glosses within Czech WordNet and hopefully they can serve as an exercise for other partners within Balkanet Project

1. Introduction

In this contribution we present an analysis of selected WN 1.5 glosses and dictionary definitions from other resources -- we examined what is the internal (semantic) organization of the glosses and dictionary definitions, i.e. how reliably and systematically they follow the standard principles of building dictionary definitions. The results following from the presented analysis should be applied in building glosses within Czech WordNet and hopefully they can serve as an exercise for other partners within Balkanet Project (2001).

When working on EuroWordNet (Vossen,1999) and now on Balkanet (2001) one has to have a look at the glosses in WN 1.5 and examine them quite closely since so far they have been regularly used as the references to the individual synsets or, in other words, as the descriptions of their senses. It is no secret that there are many reservations with regard to the glosses, especially to their properties from the lexicographical point of view. The main objections that can be heard are that the glosses are not consistent enough, that quite often they are nothing more than just examples and sometimes they are completely missing .

In writing dictionary definitions the following techniques are regarded as the standard ones:

1. definitions using *genus proximum* and the *distinguishers* (typically for nouns),
2. definitions using semantic components or features (verbs, e. g. **hurt***:*6 = cause pain),
3. definitions based on the relation of *troponymy* (*laugh – guffaw*),
4. definitions using synonymical explanations (typical for adjectives, e. g. *clever – smart*),
5. definitions based on collocational determination of the sense (e. g. *a bad student*, *bad debt*).
6. definitions based on the descriptions of events or situations (see e.g. definition of **bend**:1 (in Cobuild95, p.144) – *when you* **bend** *you move top part of your body downwards and forwards*,
7. definitions exploiting various kinds of *ad hoc* descriptions or explanations or just mere examples as e.g .**bring**:2 in WordNet 1.5 – *bring or fetch; "Could you bring over the wine?"*

2. Noun Glosses in WN 1.5

A large group of nouns denote the particular physical objects such as *table:1*, *chair:2*, etc. Thus we selected few "furniture" expressions and examined their glosses. It can be said that they "behave" in a relatively standard way, typically, these glosses follow the classical dictionary definition pattern, i.e. first part of the gloss consists of *genus proximum* (GP) and the second one represents the *distinguishers* (d1, d2, …, dn). Some slight inconsistencies can be observed: while **table***:*2 has as its genus proximum *a piece of furniture* (and other hyponyms of it as well), **chair***:*2 has as its GP *a seat for one person* (**seat**:2) and only then **seat**:2 displays as its GP *a piece of furniture.* Thus a question may be asked why the principle of GP is not followed strictly here. The good news perhaps is that the GP expressions in the whole WordNet can be checked and in our view corrected semi-automatically using the corresponding H/H trees. The same can hardly be applied to the distinguishers but we suggest to formalize this part of the gloss giving it a rigid structure in the form GP + *d1, d2, ..., dn.* More examples from WN 1.5 can be given, e.g.: **knife**:1 *cutting instrument + d1, d2,...* but in the corresponding H/H tree we find as the next upper node **edge tool**:1 – *any tool with sharp cutting edge*. The conclusion is obvious: we should try to avoid these inconsistencies in building Czech glosses and it can be seen that they can be checked semi-automatically as well. We examined also some other nouns like **digital computer**:1 or **house**:1 and it can be concluded that the situation with respect to their glosses is more or less the same.

The next point we are interested in is the semantic organization of the noun glosses or dictionary definitions in general, and how it is related to their syntactic structures. We can observe here quite a good parallelism between GP and the first noun group in the dictionary definition.

If we have look at the distinguishers it can be seen that they are expressed in several ways: as noun groups, relative sentences, adjectival phrases with complements or as prepositional groups. The closer examination, however, shows that the picture is more complicated and the

no of entries ("sentences" processed)	5935	100 %	
not applied	1207	20.3 %	
applied	4728	79.7 %	
from this:			
def1:	548	11.6 %	entry = one_word_synonym
def2:	2987	63.2 %	entry = (Ng | Pg)+
def3:	877	18.5 %	entry = (Ng | Pg)+ Ap (Ng | Pg)+
def4:	92	2.0 %	entry = Ng Sr
def5:	5	0.1 %	combination of def3 and def4
def6:	201	4.2 %	[kdo|co|někdo|něco] .*
def7:	17	0.4 %	[schopnost|neschopnost] .*

Table 1. Frequencies of the different definition types

corresponding surface syntactic structures are much richer (see below).

Thus it is our opinion that we should try to parse the dictionary definitions in order to discover the inventory of the syntactic structures that may correspond to the GP + d1, d2, ... dn scheme. For this purpose we again selected several typical "furniture"examples from SSJČ (1960) together with their English equivalents from NODE (1998). Angle brackets in Czech descriptions mark out the particular groups (and the grammatical cases in which they may occur).

stůl: <*kus*>$_{ng1}$ <*nábytku*>$_{ng2}$ <*tvořený*>$_{ap}$ <(*vodorovnou*) *deskou*>$_{ng7}$ <*na nohách*>$_{png6}$ <*nebo*>$_{conj}$ <*na podstavci*>$_{png6}$

table: ***a piece of furniture*** *with a flat top and one or more legs, providing a level surface on which objects may be placed, and which can be used for such purpose as eating, writing, working or playing games*

židle: <*přenosný kus*>$_{ng1}$ <*nábytku*>$_{ng2}$ <(*s opěradlem*)>$_{ng7}$ <*k sezení*>$_{png3}$ <*pro jednu osobu*>$_{png4}$

chair: ***a separate seat*** *for one person, typically with a back and four legs*

křeslo: <*pohodlné sedadlo*>$_{ng1}$ <*s opěradly*>$_{ng7}$

armchair: ***a large, comfortable chair*** *with side supports for a person's arm*

skříň: <*vyšší kus*>$_{ng1}$ <*nábytku*>$_{ng2}$ <*na ukládání různých předmětů*>$_{png4}$ <*nebo*>$_{conj}$ <*na věšení šatstva*>$_{png4}$

cupboard: ***a piece of furniture*** *with a door and usually shelves, used for storage*

blbec, **blb**: <*velký hlupák*>$_{ng1}$, <*pitomec*>$_{ng1}$, <*idiot*>$_{ng1}$

idiot: *a stupid* ***person***

student: <*posluchač*>$_{ng1}$ <*vysoké školy*>$_{ng2}$ <*nebo*>$_{conj}$ <*žák*>$_{ng1}$ <*střední školy*>$_{ng2}$

student: ***a person*** *who is studying at a university or other place of higher education*}

2.1. Syntactic structures of the dictionary definitions

The basic Table 1 has been obtained from the sample containing 10 000 noun dictionary definitions from SSJČ and show the main types of the syntactic patterns as they can be found within the noun dictionary definitions in SSJČ.

It can be said the definitions of the entries for whose no structure has been found usually can be intuitively classified as belonging to some of the groups 1-5. However, they may display very complicated structures (e.g. very complicated attributive noun groups), that prevent the parser (the particular rules in it) from recognizing them. There are only few entries that do not belong into any of the introduced groups/categories, for example *názor*, *že ...* (*the opinion that …*)

2.2. Czech WordNet

What also can be done is to check semi-automatically the heads of these noun groups against the corresponding nouns in Czech WordNet and to see how regularly they contain the hyperonymical expressions (such as **furniture** in our example group of selected furniture nouns) – this can be done by comparing them with the corresponding H/H trees in WordNet.

If we take the parsed syntactic structures of the processed dictionary definitions and extract their head noun groups representing (according to our parser) the GP pattern we obtain a list of expressions that are hyperonyms of the headwords in the dictionary definitions. The first part of this list is given below and it contains 30 most frequent (Czech) hyperonyms (sorted according to their frequency) from our sample of dictionary definitions. To confirm that they are hyperonyms we compared them with the corresponding expressions from Czech and English WordNet (the first number indicates the frequency in the Czech sample, then there is Czech literal with its sense number in Czech WordNet and its English equivalent with its respective sense number as well. The results of the comparison show that all the expressions extracted from the dictionary definitions are hyperonymical, thus in this way confirming our starting assumption that GP patterns can be processed and obtained from the dictionary definitions automatically. Then it is also possible to check for their consistency. The next goal is to try to recognize the distinguishers at least in a semi-automatic way though we are aware that this task is not going to be as easy as the former one.

173: kdo (who)
91: přístroj:1 (apparatus:1)
72: druh:1 (sort:2, kind:1)
63: zařízení:1 (installation:2)
56: část:1 (part:3)
52: člověk:1 (human:1, person:1)
42: souhrn:1 (aggregate:1, sum:1)
42: místo:1 (place:10)
37: nástroj:2 (instrument:2)
36: obor:2 (discipline:5)
35: nauka:1 (doctrine:1)
34: látka:2 (matter:1, substance:1)
28: přísluąník:1 (member:4)
27: skupina:1 (group:1)
26: způsob:2 (means:1, way:1)
22: jednotka:3 (unit:8)
21: něco (something)
21: činnost:1 (activity:1)
20: stav:1 (state:1)
20: součást:2 (component:1)
19: vlastnost:1 (quality:1)
18: místnost:1 (room:1)
18: hornina:1 (rock:1, stone:1
17: stroj:1 (machine:2)
16: útvar:2 (formation:5)
16: sloučenina:1 (compound:4)
16: schopnost:3 (ability:1)
16: pracovník:1 (worker:2)
14: oddělení:2 (department:1)
14: nedostatek:1 (deficiency:1)
14: názor:1 (opinion:1).

3. Verb Glosses in WN 1.5

At the first glance it can be observed that the verb glosses are less consistent and regular than noun ones. Also some glosses are missing more frequently (e.g. *write:*7). We have selected verb *to kill* and its hyponyms to see how reliable the glosses are and how they are built. If we take *kill:5 cause to die* we can immediately see that GP + d1,…, dn principle does not apply here. This is generally due to the fact that the semantic nature of verbs as the relational elements is different from the nouns and that is why they require other types of definitions.

With *kill:5* the analysis to the simpler semantic components is used (type 2 above), however the problem is that the respective semantic components are used rather spontaneously, they are not defined anywhere and they are in no way related to the Top Ontology which certainly represents a collection of the specific semantic components or features. It is very instructive to examine some of the hyponyms of *kill:5* and their glosses:

behead:1 cut the head of sb (synonymical explanation)

drown:3 kill by submerging in water (troponymy relation)

poison:5 no gloss at all

shoot:16 kill by firing a missile (troponymy relation)

stone:7 "adulterers should be stoned according to the Koran" (just the example)

strangle:1 squeeze the throat of sb (synonymical explanation)

sabre: in the sense of killing not found in BNC

overlay: in the sense of killing not found in BNC

The picture we can see is rather confusing: in the cases of *drown:3* and *shoot:16* the relation of troponymy is used as the defining principle in the gloss (different manners of killing), *behead:1* and *strangle1* are defined by synonymical explanations, however *strangle:1* is not defined correctly, to squeeze the throat of a person is not enough to kill him or her, thus the gloss is defective. Moreover, for *stone:1* the example is offered instead of the definition, though *to kill by stoning* certainly could have been used. To complete this certainly not consistent view we can only add that *poison:5* has no gloss assigned at all in WN 1.5 though again *kill by using poison* offers itself as an obvious solution. It may be interesting to note that *sabre*:4 given in WordNet 1.5 as a hyponym of *kill*:5 does not occur in British National Corpus at all.

3.1. The Possible Solutions for Verbs

One of the techniques that has to be considered with regard to the verbs is an appropriate semantic classification of verbs yielding the semantic classes of verbs. The information about the semantic class a verb belongs to can become a part of the gloss/definition and can make it more systematic. Though the criteria for establishing the semantic classes may be in a certain degree arbitrary on the other hand they may be compared with Genus Proximum principle that seem to work well for nouns.

Levin's (Levin, 1993) semantic classification of English verbs appears as an interesting solution – we have tried to develop a similar semantic classification of Czech verbs that can be applied here.

4. Adjective (and Adverb) Glosses in WN 1.5

The selected examples of the adjective synsets for *good* can well demonstrate the point.

good:8, dear:2 with or in a close relationship: "a good friend"

good:10 "good taste" (an example only)

good:12 resulting favorably: "it is a good thing that I wasn't there"

good:13, unspoiled:2 "the meat is still good" (an example only)

good:14 not forged: "a good dollar bill"

good:15 having desirable or positive qualities, esp. those suitable for a thing specified: "good news from the hospital", "a good joke", "a good secretary"

good:16 morally admirable

good:23, just:6 of moral excellence: "a genuinely good person"

good:18 appealing to the mind: "good music", "a serious book"

good:19 agreeable or pleasant: "good manners"

good:25, secure:12 financially sound: "a good investment"

good:26 in excellent condition: "good teeth"

good:27 well above average in performance: "a good student"

good:29, lucky:4 "it is good that nobody saw you" (an example only)

in good taste:1 no gloss, syntactically this case can be hardly classified as an adjective.

It can be seen that for adjective *good* the definitions of the type 4, 5, 6 are used. The most frequently used are the synonymical explanations (type 4 definitions) combined with the examples of typical collocations (type 6 definitions). Only *good:16* does not include a collocational example.

The presented examples also clearly demonstrate that many senses of *good* are very close to each other and it is not easy to discriminate them. It can be observed that *good:15* seems to cover/represent the main sense of *good* and that *good:18* or *:26* or *:*27 just stress some rather arbitrarily selected semantic features such as *in excellent condition* which can be certainly classified under *a positive quality*. The adduced examples convincingly show how the senses of *good* are split into the fine grained senses but at the same time the question has to be asked what can we gain by splitting senses in this way (quite typical for WN 1.5)? The hope is that the split senses can be integrated into the larger groups and in this way the number of senses can be reasonably reduced to obtain simpler and better applicable collection of the senses. In our view the appropriate sets of the semantic features have to be considered in combination with the collocational examples – in this way the operational classification procedures (relying on corpora) for reasonably large group of adjectives and adverbs can be obtained.

The obvious conclusion also is that it is necessary to pay the more detailed attention to the collocational examples (type 6 definitions, if they can be taken as such), to explore their behaviour in the corpora and on this ground to design the techniques of their semiautomatic handling.

5. The Conclusions for Standardization

The above analysis leads us to the following steps in the building glosses within Czech WordNet (with the hope that they can appear useful in the development of other WordNets as well):

- to use the different types of definitions for the different parts of speech in a systematic way, i.e.. GP + d1, d2,…, dn mostly for nouns, semantic components and troponymy relations for verbs and synonymical explanations combined with collocational examples for adjectives,
- to use the semantic classification of Czech verbs and integrate it appropriately into the glosses,
- to examine in a more detailed way the GP + d1, d2,…, dn definitions for nouns and to check whether the distinguishers can be inherited systematically within H/H trees,
- to examine whether the distinguishers can also capture the relation of meronymy/holonymy and in the positive case to find out how frequent it is,
- to explore systematically the collocational examples using corpus data and integrate them systematically into the adjective glosses,
- the ultimate goal of the mentioned steps is to obtain the glosses for the particular synsets that would be as systematic, formal and consistent as possible.

We have tried to show how the indicated solutions may work for the selected collections of Czech synsets and in this way they may help to standardize the glosses used in Czech WordNet..

6. Bibliography

Balkanet Project, 2001, www pages http://www.ceid. upatras.gr/Balkanet/

Collins Cobuild English Dictionary, ed. by J. Sinclair, London, Harper Collins Publishers, 1995.

Havránek B. et al., Slovník spisovného jazyka českého (SSJČ, Dictionary of Written Czech), Academia, Praha, 1960.

Levin, B., English Verb Classes and Alternations, The University of Chicago Press, Chicago, 1993.

New Oxford Dictionary of English, ed. by P. Hanks, Oxford University Pres, Oxford, 1998.

Vossen, P., EuroWordNet 1, 2, Final Report, University of Amsterdam, CD ROM, 1999.

Žáčková, E.: Partial Parsing (of Czech), Ph.D. Thesis, Masaryk University, Brno, 2002.

Standardizing Wordnets in a Web-compliant Format: The Case of GermaNet

Claudia Kunze, Lothar Lemnitzer

Seminar für Sprachwissenschaft
Universität Tübingen
Wilhelmstr. 113, 72074 Tübingen, Germany
{kunze,lothar}@sfs.uni-tuebingen.de

Abstract

Following the success of the Princeton WordNet, a range of wordnet initiatives have been launched, either monolingual or multilingual. The variety of wordnets which have a common core architecture but also their language-specific peculiarities calls for a common standard to enhance interoperability, to merge of different lexical resources and to define a common application programme interfaces. At the same time, the drive for the "semantic web" and the resp. need for ontologies calls for XML- and RDF-binding of at least the common core architecture of wordnets. The GermaNet group therefore wishes to contribute to the standardization of wordnet architectures by presenting the data model of GermaNet, an XML binding of this data model and some proposals for a common terminology.

1. Introduction

Have you ever tried to use your razor or your hair dryer in another country than that where you bought this device? Even in Europe you might be caught in a situation where the plug of your device and the socket in your hotel room are incompatible. You might end up buying an expensive adapter at the reception desk of your hotel. Missing standards can be a burden or even an obstacle to further development.

Avoiding a waste of time and money is one incentive of undergoing the effort of negotiating a standard, which in itself can be a time-consuming task.

The situation of the European traveller might be comparable to that of a language engineer who wants to:

- Use wordnets of various languages in a multilingual application environment
- Adapt an application which uses a wordnet in one language to another language and wants to adapt an available wordnet for that language, too
- Couple a dictionary management or visualization tool for a wordnet in one language with the wordnet of his / her language (see Pavelek and Pala, this volume)

It is therefore in our opinion worth the effort to discuss the following issues. In what manner are the WordNet architecture, the EuroWordNet architecture and the architecture of any individual wordnet are related? Is there a common core architecture? Do we really mean the same if we use the same concepts and terms to describe our resources? Do we perhaps refer to the same concepts though we are using different terms?

The GermaNet development group wants to contribute to this discussion. First of all, we describe the features which GermaNets shares with other wordnets, in particular the Princeton WordNet (section 2). We will present the data model of GermaNet in an application neutral graphical form, using the Entity Relationship model (section 3), as well s an XML binding of the GermaNet data model (section 4). In section 5 we will show a way of integrating the Interlingual Index of the EuroWordNet architecture into the GermaNet architecture. We will explicate the terminology we use and relate it to other wordnet terminologies, the Princeton WordNet and the Czech word net in particular (section 6). Finally, we will raise compatibility issues and suggest solutions to at least some of them (section 7).

The task we are facing is not exciting nor is it easy. Anyway, our motivation to solve it should be clear to all developers of wordnets: Think of the plug and the socket!

2. GermaNet: its standard core and its peculiarities

2.1. General Remarks

The fundamental lack of electronic lexical-semantic resources for German (see Hamp & Feldweg (1997)) was the major motivation for constructing GermaNet a few years ago. Therefore, a first project (SLD) created an on-line thesaurus covering the German basic vocabulary. GermaNet adopted the design principles and the database technology from the Princeton WordNet. However, GermaNet includes principle-based modifications on the constructional and content-oriented level which we will describe later on.

GermaNet currently covers some 40,000 synsets with more than 60,000 word meanings, modelling nouns, verbs, adjectives and adverbs (see Kunze (2001)). Within the EuroWordNet project, GermaNet was integrated into the polylingual EuroWordNet database (see Vossen (1999), Wagner and Kunze (1999)). We followed the merge approach, i.e., a wordnet is built independently from WordNet and the synsets are linked to the Interlingual Index (ILI) by creating the appropriate relations. The merge approach preserves language-specific patterns with differing hierarchical structures in comparison to the WordNet structure.

2.2. Major differences to WordNet

In spite of its general similarity with and compatibility to WordNet, we can state the following differences for GermaNet:

- we are using **artificial**, i.e. non-lexicalised **concepts**, which have been introduced to fill

- lexical gaps, to balance the taxonomical structure more adequately and to avoid unjustified co-hyponymy;
- in GermaNet, **adjectives** are ordered hierarchically as opposed to Princeton's grouping by the satellite approach;
- we pursued a uniform treatment of **meronymy** within GermaNet, whereas WordNet has established three different pointers for *Part, Member* and *Substance*;
- within GermaNet, the **causation relation** can be encoded between all parts of speech, not only between verbs and adjectives;
- due to emphasizing the syntax-semantics-interface for disambiguation tasks we accounted for over one hundred verbal **subcategorisation frames**. These frames are more elaborate than the WordNet frames, and, furthermore, for each verb reading we provide a typical **example**.

These differences and their technical impact on compatibility for the XML conversion are outlined in more detail below.

3. The data model of GermaNet

We visualize the data structure by graphic means using the Entity-Relationship Model (Chen, 1976).

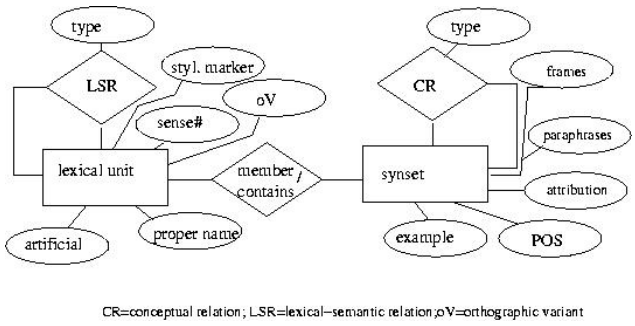


Fig. 1: Entity-Relationship graph of the GermaNet data model

The graph in figure 1depicts:

- the **objects**, synsets and lexical units, which are represented as *rectangles*,
- the **attributes** of these objects, represented as *circles*,
- the **relations**, represented as *diamonds*. In GermaNet, like in WordNet, we distinguish:
  - **conceptual** relations (CR) which hold between instances of the synset object (e.g. hyperonymy) from
  - **lexical-semantic** relations (LSR) which hold between instances of the lexical unit object (e.g. antonymy).

From an Entity-Relationship model, one can formally derive the conceptual structure of a relational database in a normalized form (Seesing, 1993). One can also, however not as unambiguously, derive a DTD or schema for an encoding of the data which is in line with the XML standard.

4. An XML Binding of the Data Model

We have converted the GermaNet data into a set of XML-encoded documents which conform to two *Document type definitions* (DTDs). One DTD represents the objects (synsets and lexical units) and their attributes, the other represents the relations between these objects.

In the following, we will describe both DTDs. The first DTD represents the data model of the objects and their attributes. It is recorded completely in fig. 2.

```
<!-- DTD for Germanet objects -->
<!-- Version 1.9, March 2002 -->>
<!-- Copyright: Sem. f. Sprachwissenschaft der
Universität Tübingen -->

<!ELEMENT synsets      (synset)+>
<!ELEMENT synset       ((lexUnit)+, attribution?,
frames?, paraphrases?, examples?)>
<!ATTLIST synset       id            ID     #REQUIRED
                       wordClass     CDATA  #IMPLIED
                       lexGroup      CDATA #IMPLIED>
<!ELEMENT lexUnit      (orthForm)+>
<!ATTLIST lexUnit      id            ID     #REQUIRED
                  StilMarkierung  (ja|nein)     "nein"
                  sense  CDATA          #REQUIRED
                  orthVar        (ja|nein) "nein"
                  artificial     (ja|nein)  #REQUIRED
                  Eigenname      (ja|nein) #REQUIRED >
<!ELEMENT orthForm     (#PCDATA)>
<!ELEMENT paraphrases  (paraphrase)+>
<!ELEMENT paraphrase   (#PCDATA)>
<!ELEMENT examples     (example)+>
<!ELEMENT example      (text, frame*)>
<!ELEMENT frames       (frame)+>
<!ELEMENT attribution  (#PCDATA)>
<!ELEMENT text         (#PCDATA)>
<!ELEMENT frame        (#PCDATA)>
```

Fig 2: The GermaNet objects DTD

Description: Documents which conform to this DTD contain a set of *synsets*. Every *synset* consists of at least one *lexical unit*. *Paraphrases* may be given to characterize the meaning of the synset and an *attribution* as well as *examples* may be added to illustrate the use of its member lexical units. For verb synsets, subcategorization *frames* are given. The individual lexical units are characterized by a set of attributes, e.g. *sense* number and stylistic marker (*StilMarkierung*). A concept can be represented by a string which does not correspond to a lexical unit in the German vocabulary. Such a unit will be marked as *artificial*. The content model of most atomic elements is set to *#PCDATA*, therefore minimizing data type restrictions. It is up to the lexicographers to fill the elements with appropriate data.

```
<!-- DTD for GermaNet relation files.-->
<!-- Version 1.4, März 2002 -->>
<!-- Copyright: Sem. f. Sprachwissenschaft der
Universität Tübingen -->

<!ELEMENT relations (lex_rel | con_rel)+>
```

```
<!ELEMENT lex_rel (locator+, arc+)>
<!ATTLIST lex_rel name (antonymy | pertonymy |
participleOf) #REQUIRED
            dir (one | both) #REQUIRED
            sense          CDATA #REQUIRED
            xmlns:xlink CDATA #FIXED
'http://www.w3.org/1999/xlink'
             xlink:type (extended) #FIXED 'extended'>
<!ELEMENT con_rel (locator+, arc+)>
<!ATTLIST con_rel name (hyperonymy | meronymy |
holonymy | entailment | causation | association)
#REQUIRED
            dir (one | both) #REQUIRED
            xmlns:xlink CDATA #FIXED
'http://www.w3.org/1999/xlink'
            xlink:type (extended) #FIXED 'extended'>
<!ELEMENT locator EMPTY>
<!ATTLIST locator xlink:type (locator) #FIXED 'locator'
              xlink:href CDATA #REQUIRED
              xlink:label CDATA #REQUIRED>

<!ELEMENT arc EMPTY>
<!ATTLIST arc xlink:type (arc) #FIXED 'arc'
        xlink:from CDATA #REQUIRED
        xlink:to CDATA #REQUIRED
        xlink:actuate (onRequest) #FIXED 'onRequest'
        xlink:show (other) #FIXED 'other'>
```

Fig. 3: The GermaNet relations DTD

Description: Documents which conform to this DTD contain a set of relations which are either conceptual or lexical relations. These relations are characterized by their type (attribute: name) and they are marked as either symmetrical or directed (attribute: dir). They are realized as links according to the XLink specification: a link consists of two nodes (locators, specified through the IDs of the synsets or lexical units) and one or two arcs, depending on whether the relation is directed or symmetrical. The attributes of the 'arc' element specifies the processual behaviour whenever a link is traversed.

5. Extensions of the Data Model and DTD

5.1. Cross-lingual extension with EuroWordNet

Within a European project, the wordnets of several languages, including German, have been integrated into the polylingual architecture of the EuroWordNet database. This has been achieved by linking the language-specific concepts to the Interlingual Index (ILI) of EuroWordNet (Vossen, 1999). The ILI has the following features:

- It is an unordered list of synsets, so-called ILI-records;
- Each ILI-record has a unique identifier, consisting of a categorial marker and a sense ID;
- The ILI-records have basically been derived from the Princeton WordNet; some new ones have evolved from the project;
- The ILI does not account for structural relations between the records. The structural relations are provided by the language-specific wordnets being linked to the ILI.

An example of the ILI and its satellites is shown in fig. 4

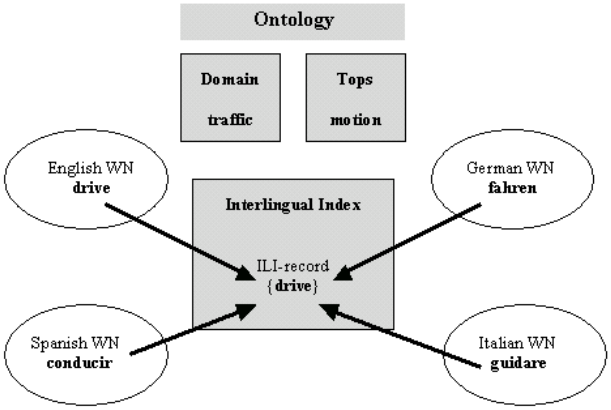


Fig 4: Partial architecture of the EuroWordNet database

From fig. 6, one can derive that there is no direct connection between the wordnets of the various languages. Mappings between language-specific wordnets are mediated by the Interlingual Index.

The following inventory of equivalence relations for connecting synsets of an individual wordnet to the ILI is provided by the EWN specification:

- EQ_SYNONYM
- EQ_NEAR_SYNONYM
- EQ_HAS_HYPERONYM
- EQ_HAS_HYPONYM
- EQ_INVOLVED
- EQ_ROLE
- EQ_IS_CAUSED_BY
- EQ_CAUSES
- EQ_HAS_HOLONYM
- EQ_HAS_MERONYM
- EQ_HAS_SUBEVENT
- EQ_IS_SUBEVENT OF
- EQ_BE_IN STATE
- EQ_IS_STATE_OF

Furthermore, the relations between a wordnet synset and an ILI element are directed. The wordnet synset is the source and the ILI element is the target of this link.

Given these characteristics, we extend the GermaNet relations DTD in the following way:

- Introduce an additional element for this new class of links ("equivalence link")
- Characterize the link as directed
- Define an attribute with the closed set of types which characterize ILI links in the EuroWordNet architecture
- Define two locators for the link, one of which must have an identifier designating a GermaNet synset, the other an identifier designating an ILI element
- Define an arc between these two locators and specify the application semantics of the link during traversal of this arc.

The result of this procedure is shown in fig. 5.

```
<!-- DTD for GermaNet relation files – extended,
interlingual version.-->
....
<!ELEMENT relations (lex_rel | con_rel | eq_rel)+>
…
<!ELEMENT eq_rel (locator+, arc+)>
<!ATTLIST eq_rel name (EQ_SYNONYM|
EQ_NEAR_SYNONYM| EQ_HAS_HYPERONYM|
EQ_HAS_HYPONYM| EQ_INVOLVED| EQ_ROLE|
EQ_IS_CAUSED_BY| EQ_CAUSES|
EQ_HAS_HOLONYM| EQ_HAS_MERONYM|
EQ_HAS_SUBEVENT| EQ_IS_SUBEVENT OF|
EQ_BE_IN STATE| EQ_IS_STATE_OF) #REQUIRED
                dir (one | both) #FIXED 'one'
                xmlns:xlink CDATA #FIXED
'http://www.w3.org/1999/xlink'
                 xlink:type (extended) #FIXED 'extended'>
…..
```

Fig. 5: Extended interlingual relations DTD

A core of GermaNet synsets has been linked to the Interlingual Index (ILI). In the process of linking these synsets have got a separate ID. We could have used the IDs as a key to those synsets. The fact however that only one third of the synsets is linked to the ILI led us to the decision to employ our own scheme of IDs, which are processed on conversion of the data. We will provide a mapping from ILI link IDs to the IDs generated by our programs.

6. Terminology

In this section we want to compare the terminology we use with those employed by other wordnet development groups. The documents we refer to are the description of the Princeton WordNet (Miller, Fellbaum) and the description of the Czech WordNet (Pavelek and Pala, this volume).

Uncontroversially the *synset* is the central object of every wordnet. A synset consists of one or many members. In the RDF binding of WordNet these members are called *word forms*. Some wordnet development groups call them *synonyms*. We decided to use neither because:

- *Word form* denotes a concrete linguistic entity, in many times inflected and found in texts, whereas the members of synsets are lexical abstractions which are represented by one form, the so called base form.
- *Synonym* is a genuinly relational term. A lexical sign can be a synonym only in relation to some other lexical sign.

In contrast, we use the tem *lexical unit* to establish a distinct kind of object which has its own attribute-value pairs. Furthermore, the term is also used with traditionally organized lexical resources and can therefore facilitate a merge of different kinds of lexical resources.

Lexical units are organized in synsets by the central relation of *synonymy*. It is however not clear to us wether all groups employ the same definition of *synonymy* and the same set of operational tests. On the other hand, the linking of synsets with a narrow definition of *synonymy* to synsets with a wider definition of it – in interlingual relations – might cause severe problems in multi-lingual application environments. We believe that the reliability of equivalence relations between synsets is worth testing.

Lexical units are represented by "literal strings" (we are using the term *orthographical form*) and sense numbers.

Part of speech plays a central role as a feature of synsets in that it divides the set of concepts into subsets. Most wordnets comprise nouns, verbs and adjectives. There is a strong tendency therefore to stick to these parts of speech even if they do not prove adequate for all languages (see Kahusk, this volume, for a more detailed discussion).

Most wordnets provide a textual description of synsets. In WordNet and in the Czech word net they are called *glosses*, whereas we are using the term *paraphrase.* The WordNet RDF *glossary* however seems to comprise paraphrases and examples, which are two different data types in GermaNet. This point needs clarification. We are not against using the term *gloss* if it is well defined.

Again there is little difference in the kinds and types of relations within wordnets. There are *conceptual relations* between synsets and *lexical-semantic relations* between lexical units. Some wordnet development groups however (see Pavalek and Pala, this volume; Vider, this volume) use the tem *semantic relations* instead of *conceptual relations.* The Czech wordnet developers are using *literal relations* to signify what we call *lexical-semantic relations.*

In addition, EuroWordNet 1 defined a set of interlingual relations between synsets on which at least the members in this project phase agreed. Furthermore this project provided a proposal for a set of intralingual relations which at least some of the new members of the wordnet society in Europe have taken over.

The Estonian wordnet applies a much richer set of semantic relations than e.g. WordNet(see Vider et al., 1999). Furthermore the developers are in need of a set of subtypes for the EWN relation *derived / has_derived / derived_from.*

Furthermore there are differences between the architecture of WordNet (at least the RDF binding), GermaNet and the Czech wordnet.

In WordNet (the RDF version), synsets, glosses and relations (or to be precise, the hyperonymy relation) are organized in different files. In GermaNet (the XML version) the synsets and synset related features are organized separately from the relations, which are called *links* in orientation to the Xlink standard. In the Czech wordnet synsets and relations are organized in one data structure. Glosses are stored in a different file for the simple reason that the English WordNet glosses are used until Czech glosses will be generated (see Pavelek and Pala, this volume). This however, seems to be a minor, merely technical point. At least, GermaNet offers a data structure comparable to the Czech wordnet.

There are only a few if any information types other than the the ones mentioned which are shared by a larger number of wordnets. Subcategorization frames seems to be one candidate. However it might be even more difficult to come to an agreement about the status and the information provided by this data type. This and other information should be treated as particular to any individual wordnet.

7. Compatibility issues

In this section, we will raise several compatibility issues and show how they can be solved within the XML framework We will elaborate on six types of structural differences between WordNet and GermaNet:

1. Objects or relations might have different extensions in both nets, as is the case with the CAUSE relation. In WordNet, this relation holds exclusively between verbs and adjectives. In GermaNet, synsets of all word classes are in the domain of this relation. True compatibility would require a finer granularity of the CAUSE relation in GermaNet. This could be realised by adding an attribute to it. The values of this attribute would lead to at least two subsets of items: one which is extensionally identical with the WordNet CAUSE relation and one which characterises the GermaNet-specific extension.
2. The granularity of a relation differs. For example, WordNet divides the generic part-whole relation into three sub-relations: part (e.g. *arm,body*), member (e.g. *director, staff*), substance (e.g. *glass, glass plate*). Other values might be added to this list. GermaNet, in contrast, uniformly applies the generic relation. We recommend for WordNet or any other wordnet which applies this architecture to add an attribute to a truly generic part-whole-relation which divides the instances into three classes. In GermaNet, this attribute might get a value ANY, until a more fine-grained specification is implemented.
3. There are a few attributes specific to GermaNet, e.g. *StilMarkierung* (=stylistic marker) as an attribute of lexical units. For instance, the German concept *schlafen* (=sleep) has *ratzen*s, pennen*s, knacken*s, pofen*s* as hyponyms which are stylistically marked. These attributes can be INCLUDED in GermaNet and EXCLUDED elsewhere. The same holds for language-specific features of other word nets, e.g. features like *katharevousa* and *demotiki* in Greek.
4. An attribute which is equivalent in both wordnets specifies a different set of values. This holds for the *verb frame* attribute. The German verb frames which are implemented in GermaNet are a closed class. For type checking, it could have been more elegant to define an attribute with a fixed set of values. For compatibility reasons, however, we voted for an element group "frames" with frames as its elements and #PCDATA as data type
5. The adjective domain in GermaNet differs fundamentally from that in WordNet. The domain is ordered hierarchically in GermaNet, whereas WordNet applies an associative similarity relation which groups adjectives in equivalence classes. At present, we do not see any easy solution which would preserve compatibility in this case.

8. Conclusion

We presented the GermaNet data model and an XML binding for it in order to contribute to the difficult process of establishing a standard for at least the core architecture of wordnets. On the way to a standard both conceptual and terminological issues arise. With respect to visualization tools and the semantic web we decided to choose XML in general, and two DTDs in particular, to present our view of the GermaNet architecture.

9. Acknowledgments

The work we report here is in part the outcome of some student projects. We would therefore like to thank Iris Vogel (Heidelberg) and Holger Wunsch (Tübingen) for their valuable contributions. Research on GermaNet was funded by the Land Baden-Württemberg.

10. References

Chen, P. P.-S., 1976. The Entity-Relationship Model - Towards a Unified View of Data. *ACM TODS 1* No. 1 (March 1976):9-36.

Fellbaum, C., 1998. *WordNet: An Electronic Lexical Database*. Cambridge, Mass.: MIT Press.

Hamp, B. and Feldweg, H., 1997. GermaNet - a Lexical-Semantic Net for German. In: *Proceedings of the ACL/EACL-97 workshop on Automatic Information Extraction and Building of Lexical Semantic Resources for NLP applications*. Madrid, July 7-12, 1997.

Kahusk, Neeme, 2002. A Lexicographer's Tool for Word Sense Tagging According to WordNet. In: *Proc. of the LREC Workshop on Word Net Structiure and Standardization and how these Affect Wordnet Applications and Evaluation, Las Palmas, 28 May 2002.*

Kunze, C., 2001. Lexikalisch-semantische Wortnetze. In K.-U. Carstensen et al. (eds.): *Computerlinguistik und Sprachtechnologie: eine Einführung*. Heidelberg; Berlin: Spektrum, Akademischer Verlag, S. 386-393.

Kunze, C. and Wagner, A., 2001. Anwendungs-perspektiven des GermaNet, eines lexikalisch-semantischen Netzes für das Deutsche. In Lemberg, I. & B. Schröder & A.Storrer (eds.): *Chancen und Perspektiven computergestützter Lexikographie*. Tübingen: Niemeyer. Lexicographica Series Maior 107. S. 229-246.

Lemnitzer, L. and Kunze, C., 2002. Adapting GermaNet for the Web. *Proceedings of the first Global WordNetConference*, Central Institute of Indian Languages. Mysore, India, 2002, pp. 174-181.

Miller, G. et al., 1990. *Five papers about on WordNet.* CSL-Report, Vol. 43. Cognitive Science Laboratory, Princeton University.

Pavelek, Tomas and Pala, Karel, 2002. WordNet Standardization from a Practical Point of View. In: *Proc. of the LREC Workshop on Word Net Structiure and Standardization and how these Affect Wordnet Applications and Evaluation, Las Palmas, 28 May 2002.*

Seesing, Paul R., 1993. *Basic Systems Analysis Tools for Computer Users* (http://www.open.org/~prslkg/syintro.htm)

The Semantic Web Community Portal (URL http://www.semanticweb.org/)

The Semantic Web Community Portal – Library (URL http://www.semanticweb.org/library)

Vider, Kadri, Paldre L., Orav, H and Õim, H, 1999. The Estonian Wordnet. In: Kunze, C.. editor, *Final Wordnets for German, French, Estonian and Czech*. EuroWordNet (LE-8328), Deliverable 2D014.

Vossen, P., ed., 1998. *EuroWordNet: A Multilingual Database with Lexical Semantic Networks*, Kluwer Academic Publishers, Dordrecht.

Vossen, P., 1999. EuroWordNet. Building a Multilingual Database with Lexical-Semantic Networks for the European Languages. *Proceedings of EUROLAN'99, 4th European Summer School on Human Language Technology*. Iasi, Romania. July 19-31, 1999.

Wagner, A. and Kunze, C., 1999. Integrating GermaNet into EuroWordNet, a Multilingual Lexical-Semantic Database. *Sprache und Datenverarbeitung, SDv* Vol. 23.2/1999:5-20.

Xlink 1.0. (URL=http://www.w3.org/TR/2001/REC-xlink-20010627/)

XML 1.0. (URL=http://www.w3.org/TR/1998/REC-xml-19980210/)

WordNet Standardization from a Practical Point of View

Tomas Pavelek, Karel Pala

Faculty of Informatics, Masaryk University
Botanicka 68a, 60200 Brno, Czech Republic
{xpavelek,pala}@fi.muni.cz

Abstract

This article deals with standardization of WordNet data in practice. The format of WordNet databases mentioned here comes as a result of connection with a tool VisDic which enables browsing and editing electronic readable dictionaries. Later in the article, a possibility of embedding WordNet in a common dictionary using this format is described. Finally, a short overview of VisDic tool is presented.

1. Motivation

When we thought about the improvements of WordNet databases we knew that it would be necessary to make a tool which easily enables editing synsets, their relations and links to other wordnets. There has been a tool which met the requirements - Polaris (Louw, 1998). But Polaris was a good WordNet editor just at a first glance. It displayed many functions, but unfortunately also several serious disadvantages: It was a closed project, it was aimed to WordNet databases only and it used its own format for representation of synsets – Import/Export format (Louw, 1998).

What is needed is a program that would enable users to search also in other databases and sources like monolingual and bilingual dictionaries, dictionaries of synonyms or corpora. Thus we needed to find a format for description of possibly any type of lexical resources. The XML format fits perfectly for these purposes. In the next section, we describe the details of this format and motives that led us to the design mentioned below.

2. Format of WordNet Data

A representation of WordNet data comes from the idea that the WordNet database is a dictionary consisting of entries which correspond to the individual meanings. The meaning is described by a set of words. A meaning can also have a gloss consisting of a free text definition of the meaning. The meaning is further derived from synset relations connecting them. There are two types of synset relations (Miller 1993, Vossen 1999):

Internal Language Relations that connect synsets in the range of one language, e.g. hyperonymy, hyponymy, meronymy, holonymy relations.

External Relations which connect synsets among more languages, e.g. EQ_SYNONYM, EQ_HYPERONYM, EQ_HYPONYM.

At this point, it is necessary to make clear how to represent the relations effectively in the computer. The most effective way is to assign a key to each synset which uniquely identifies it. Then the synset can be easily referred by others just by specifying its key. But in WordNet, there already is a value which can be understood as a key, particularly, it is the Interlingual Index (ILI). Then, all relations can be represented just by their names and ILI of the target synset. Moreover, ILI immediately defines the EQ_SYNONYM relation.

The format is further extended by an information about the part of speech of each synset. The next extension divides words in the synset to a literal part and sense number part. The reason lies in the fact, that about 22% of words (e.g. *page*) have more meanings and then it is useful to distinguish them by a sense number.

Fig 1. shows the selected parts of the just described synset representation (VisDic definition). Each row contains a specific information about a synset represented by a tag. The first column contains a level of the tag in a structure. Every tag belonging to a specific level N can be understood as a part of the nearest upper tag having a level N-1. The second column contains a name of a tag. The third column contains its minimal number of repeating in a structure and the fourth column its maximal number of repeating in a structure (–1 means infinity). The fifth column contains the following information about the type of a tag:

N – the tag contains a normal text value

K – the tag contains a key value uniquely identifying the synset, this key can be used by all L, R, and E tags whose definitions follow

L – the tag contains a link to another synset, it is representing a semantic relation

R – is similar to L, but it is not necessary to store the tag, because it can be reversibly inferred by a tag stored in the sixth column

E – the tag contains an information stored in another dictionary, a name of an external tag is contained in the sixth column and a name of a dictionary in the seventh column.

Fig. 2 shows the corresponding DTD. At the first sight you can see the difference between these two descriptions. VisDic definition does not contain any information about attributes. Therefore, all tags are understood as elements from a DTD point of view. On the other side it is not crucial to specify which information should be understood as an attribute and which as an element, because all elements which have not any children should be considered as attributes from a low level processing. VisDic definition can be thus understood as a description of XML which comes from a binary representation of a database, it exactly describes a tree structure of XML, while DTD defines more data types and is more readable for humans. The difference between VisDic definition and DTD is comparable to a difference between C and Prolog programming languages.

VisDic tool that will be described later uses the simplified VisDic definition of an XML database.

```
0 SYNSET               1      1 N
  1 ILI                1      1 K
  1 POS                1      1 N
  1 GLOSS              0     -1 E WORD_MEANING.GLOSS    wn/ili/wn_ili
  1 SYNONYM            1      1 N
    2 LITERAL          1     -1 N
      3 SENSE          1      1 N
  1 BE_IN_STATE        0     -1 L
  1 STATE_OF           0     -1 R SYNSET.BE_IN_STATE
  1 CAUSES             0     -1 L
  1 IS_CAUSED_BY       0     -1 R SYNSET.CAUSES
  1 HYPERONYM          0     -1 L
  1 HYPONYM            0     -1 R SYNSET.HYPERONYM
  1 HOLONYM            0     -1 L
  1 MERONYM            0     -1 R SYNSET.HOLONYM
  1 SUBEVENT           0     -1 L
  1 IS_SUBEVENT_OF     0     -1 R SYNSET.SUBEVENT
  1 ANTONYM            0     -1 L
  1 INVOLVED           0     -1 L
  1 ROLE               0     -1 R SYNSET.INVOLVED
  1 XPOS_NEAR_ANTONYM  0     -1 L
  1 XPOS_NEAR_SYNONYM  0     -1 L
  1 EQ_HOLONYM         0      1 L
  1 EQ_MERONYM         0      1 R SYNSET.EQ_HOLONYM
  1 EQ_HYPERONYM       0      1 L
  1 EQ_HYPONYM         0      1 R SYNSET.EQ_HYPERONYM
```

Fig 1. VisDic definition of synset representation (selected tags)

```
<!ELEMENT SYNSET  (POS,GLOSS,SYNONYM+)>
<!ELEMENT POS     (#PCDATA)>
<!ELEMENT GLOSS   (#PCDATA)>
<!ELEMENT SYNONYM (#PCDATA,SENSE)>
<!ELEMENT SENSE   (#PCDATA)>
<!ATTLIST SYNSET  ILI               ID #REQUIRED>
<!ATTLIST SYNSET  BE_IN_STATE       IDREFS>
<!ATTLIST SYNSET  STATE_OF          IDREFS>
<!ATTLIST SYNSET  CAUSES            IDREFS>
<!ATTLIST SYNSET  IS_CAUSED_BY      IDREFS>
<!ATTLIST SYNSET  HYPERONYM         IDREFS>
<!ATTLIST SYNSET  HYPONYM           IDREFS>
<!ATTLIST SYNSET  HOLONYM           IDREFS>
<!ATTLIST SYNSET  MERONYM           IDREFS>
<!ATTLIST SYNSET  SUBEVENT          IDREFS>
<!ATTLIST SYNSET  IS_SUBEVENT_OF    IDREFS>
<!ATTLIST SYNSET  ANTONYM           IDREFS>
<!ATTLIST SYNSET  INVOLVED          IDREFS>
<!ATTLIST SYNSET  ROLE              IDREFS>
<!ATTLIST SYNSET  XPOS_NEAR_ANTONYM IDREFS>
<!ATTLIST SYNSET  XPOS_NEAR_SYNONYM IDREFS>
<!ATTLIST SYNSET  EQ_HOLONYM        IDREF>
<!ATTLIST SYNSET  EQ_MERONYM        IDREF>
<!ATTLIST SYNSET  EQ_HYPERONYM      IDREF>
<!ATTLIST SYNSET  EQ_HYPONYM        IDREF>
```

Fig 2. DTD of WordNet Database (selected tags)

3. Advantages of VisDic definition

Looking at Fig. 3 we can see the example of two synsets stored in a typical database: {psychological feature:1} (P) and {cognition:1, knowledge:1} (C). Notice the following facts:

HYPERONYM tag of C contains exactly the same ILI value (*00012517-n*) as is present in ILI tag of P. This implies, that P is a hyperonym of C. A searching of the hyperonym is reduced just to a looking up a single value *00012517-n*, which **reduces the time for searching**.

There is no HYPONYM tag. An information that C is a hyponym of P is already present in the fact, that P is a hyperonym of C. Searching for all hyponyms of P is then converted to looking up all synsets having their HYPERONYM value the same as ILI value of P. In most cases, the synset has only one hyperonym, but it can have tens of hyponyms. Then, using reversible tags as HYPONYM **reduces the size of a database**.

If we look at Fig. 1, we can see that the gloss is present in the external file called *wn/ili/wn_ili*. There is a good reason for that. There are wordnets that do not have their own glosses at the time. Until these glosses will be added, it is good to use glosses which already exist even in another language. Therefore this external link points to a special file, where all English glosses are stored. All wordnets then can point to this place and automatically load a gloss when necessary - the format allows to **link dictionaries**.

If it is necessary the user can add its own tag, such as gloss in his language, to his own WordNet and the changes take effect immediately without a need of any further processing, such as a recompilation of the WordNet database - the format is **easily extensible**.

```
<SYNSET>
  <ILI>00012517-n</ILI>
  <POS>n</POS>
  <SYNONYM>
    <LITERAL>psychological feature
      <SENSE>1</SENSE>
    </LITERAL>
  </SYNONYM>
</SYNSET>
```

```
<SYNSET>
  <ILI>00012878-n</ILI>
  <POS>n</POS>
  <SYNONYM>
    <LITERAL>cognition
      <SENSE>1</SENSE>
    </LITERAL>
    <LITERAL>knowledge
      <SENSE>1</SENSE>
    </LITERAL>
  </SYNONYM>
  <HYPERONYM>00012517-n</HYPERONYM>
</SYNSET>
```

Fig. 3. Two synsets represented in VisDic definition

4. WordNet Embedded in Another Dictionary

There are very few relations which cannot be represented in VisDic format definition: DERIVATION, ANTONYM (for literals only), IS_DERIVED_FROM, HAS_DERIVED, PERTAINS_TO, IS_PERTAINED_TO, HAS_INSTANCE and BELONG_TO_CLASS. The reason is, that these relations do not connect two synsets, but in the most cases two literals. Synsets are uniquely identified by their ILI values, but there is no way to refer to their parts – particularly literals. Although most of WordNet data do not contain these relations it is necessary to think about how they can be represented.

One possible solution of this problem would be to specify the second key (in VisDic definition say the type of a tag K2), which makes a unique identification of each literal and sense pair. The literal relations can be then labelled by L2 or R2 type of a tag. The difference between synset links and literal links should be then distinguished (synset links have L and R type of tags). The corresponding part of VisDic definition of synonyms from Fig.1 should be then replaced by data represented by Fig.4.

```
1 SYNONYM                1      1 N
  2 LITERAL              1     -1 N
    3 ID                 1      1 K2
    3 SENSE              1      1 N
    3 IS_DERIVED_FROM    0     -1 L2
    3 HAS_DERIVED        0     -1 R2 SYNSET.SYNONYM.LITERAL.IS_DERIVED_FROM
    3 DERIVATION         0     -1 L2
    3 PERTAINS_TO        0     -1 L2
    3 IS_PERTAINED_TO    0     -1 R2 SYNSET.SYNONYM.LITERAL.PERTAINS_TO
    3 ANTONYM            0     -1 L2
```

Fig. 4. Additional information in VisDic definition for literal relations

It looks quite well but now consider, that we would like to link WordNet database to another common dictionary. Most of common dictionaries are sorted by words which correspond to the literals rather than the word meanings. Word meanings in WordNet typically contain more literals. Therefore, if we would like to refer to the word *cognition* from an example in Fig. 3 we should use a SYNSET.SYNONYM.LITERAL.ID tag instead of SYNSET.ILI tag, because ILI value comprises both *cognition* and *knowledge* literals.

Now think about the real situation, when both WordNet and the other dictionary are being edited. While ILI values of synsets are strictly given, the literals' ID's are very often modified. E.g., when a user deletes the literal from a synset, adds another literal to this synset and finally realizes that the first one was correct and replaces it back, the ID will not be the same. Therefore, during every simple change in a synset, it is necessary to update all the references to the literal. It is possible to maintain ID numbers within WordNet as a compact dictionary, but it is hard to keep consistent more different dictionaries. In our view, this is one of two reasons why this approach should not be followed.

The second reason is that common dictionaries usually contain more information about a specified word, that WordNet does. Except for a simple information such as origin of the word, morphological data (genitive form, plural form), typical collocations, etc., every verb in a common dictionary can display its valency, which may represent quite a complicated structure. From that point of view it is much easier to store relations between literals in the other dictionary. Each word description then can contain an identification of a synset (given by ILI) which specifies which word meaning it belongs to.

Fig. 5 shows a VisDic definition for a simple common dictionary with a link to a WordNet database stored in the ENTRY.SYNSET tag. The format is followed by an example of two entries of this dictionary. Notice that all literal relations are stored in this dictionary instead in WordNet itself (especially ANTONYM relation, for example). The external tag ENTRY.SYNSET allows to work with the corresponding WordNet synset, as if it were included in the common dictionary. In the first synset it is linked via 06193747-n value, in the second one, the value of external synset is 05847495-n.

```
1 ENTRY                   1       1 N
  2 ID                    1       1 K
  2 HEAD                  1      -1 N
  2 PLURAL                0      -1 N
  2 IS_DERIVED_FROM       0      -1 L
  2 HAS_DERIVED           0      -1 R ENTRY.IS_DERIVED_FROM
  2 DERIVATION            0      -1 L
  2 PERTAINS_TO           0      -1 L
  2 IS_PERTAINED_TO       0      -1 R ENTRY.PERTAINS_TO
  2 ANTONYM               0      -1 L
  2 SYNSET                0       1 E SYNSET wn/en/wn_en
```

```
<ENTRY>
  <ID>00000001</ID>
  <HEAD>man</HEAD>
  <PLURAL>men</PLURAL>
  <ANTONYM>00000002</ANTONYM>
  <SYNSET>
    <POS>n</POS>
    <SYNONYM>
      <LITERAL>adult male
        <SENSE>1</SENSE>
      </LITERAL>
      <LITERAL>man
        <SENSE>4</SENSE>
      </LITERAL>
    </SYNONYM>
    <ILI>06193747-n</ILI>
    <HYPERONYM>05850734-n</HYPERONYM>
  </SYNSET>
</ENTRY>
```

```
<ENTRY>
  <ID>00000002</ID>
  <HEAD>woman</HEAD>
  <PLURAL>women</PLURAL>
  <ANTONYM>00000001</ANTONYM>
  <SYNSET>
    <POS>n</POS>
    <SYNONYM>
      <LITERAL>adult female
        <SENSE>1</SENSE>
      </LITERAL>
      <LITERAL>woman
        <SENSE>3</SENSE>
      </LITERAL>
    </SYNONYM>
    <ILI>06434591-n</ILI>
    <HYPERONYM>05847495-n</HYPERONYM>
  </SYNSET>
</ENTRY>
```

Fig. 5. Example of a common dictionary embedding the WordNet data

5. VisDic

VisDic is a program tool which allows to browse and edit common dictionaries, corpora and also databases like WordNet. All of these resources are based on elementary structures – common dictionaries consist of entries, while WordNet is made of synsets.

The user can view more dictionaries at the same time. Each has its own sub-window consisting of three parts. The topmost one is a query box. The middle one contains all found entries and the last displays a view of a specified entry. This window is represented by a graphical item called notebook which allows to view the entry in more ways. VisDic window with two active dictionaries can be seen in Fig. 6.

The query consists of an XML tag specification, = character and a value specification, e.g. if a user likes to find all the nouns in WordNet, he has to type *SYNSET.POS=n*. One of tags can be understood as the default one. Then the tag specification and = character can be omitted, e.g. if *SYNSET.SYNONYM.LITERAL* is defined as the default tag for WordNet database all the occurrences of a word form, say *side*, can be found by typing just *side*. Queries can be grouped by logical OR (*||*) or AND (*&&*), the value can be prefixed by ^ character, which means to find all entries beginning with the value phrase, or suffixed by *$* character, which means to find all entries ending with the value phrase.

The more complete description of VisDic can be found in (Pavelek, 2002).

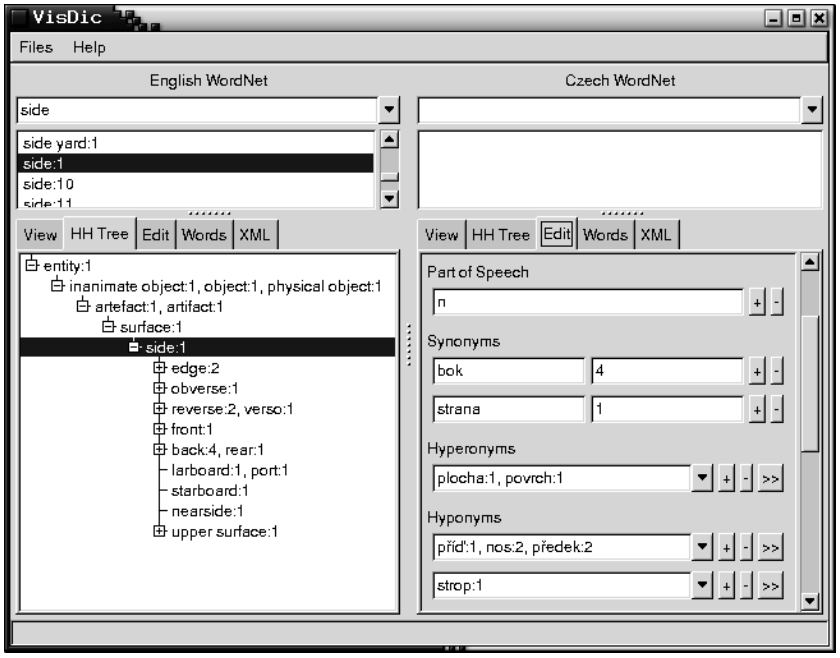


Fig 6. VisDic

6. Conclusions

The suggested format fully corresponds to the XML format as it is used for the data representation. Although we do not use a proper DTD specification fulfilling the requirements of the standard DTD in XML, the presented definition is quite similar to it and though it does not use some features that XML offers in general we think that it is well suited not only for wordnets, but also for other lexical resources as well, such as explanatory dictionaries, bilingual dictionaries, dictionaries of synonyms, corpora, etc. The format used within VisDic tool enables a user to browse and edit easily any type of database stored in it.

The conclusion that can be drawn from this exercise is the following:

The standards can be arrived at either from top (this is not our case) or from the bottom which is the solution presented here. The experience seems to show that real standards develop from the practical use shared by many users. Then the modifications from the top can be applied and adopted if the users can agree upon them.

7. References

Louw M., 1998. *Polaris User's Guide*. Lernout & Hauspie, Antwerp, Belgium

Miller G.A., Beckwith R., Fellbaum Ch., Gross D., Miller K., 1993. *Introduction to Wordnet: An On-line Lexical Database*. Princeton University

Pavelek T., Pala K., 2002. *VisDic - A New Tool for WordNet Editing*. 1st International WordNet Conference, Mysore, India.

Vossen, P., 1999. *Final Report on EuroWordNet, CD ROM*. Amsterdam University

Methodological issues in building the Romanian Wordnet and consistency checks in Balkanet

Dan Tufiş *, Dan Cristea♥

* RACAI-Romanian Academy
13, "13 Septembrie", Bucharest 5, Romania
tufis@racai.ro
♥University A.I. Cuza
16, Berthelot, Iaşi 6600, Romania
dcristea@infoiasi.ro

Abstract

The requirements in building a multilingual ontology of the EuroWordNet kind are frequently conflicting and if not considered in the first stages of the project, later harmonizing might be extremely difficult if possible at all. To ensure as early as possible usability, the incrementally developed lexical stock of each individual wordnet, should cover the most frequent vocabulary of the language. On the other hand, given that this is a multilingual lexical resource, special care should be addressed to the compatibility problems. Specifically, there are two main compatibility issues to be considered: there should be a cross-language conceptual coverage, meaning that each monolingual lexicon should globally deal with the same conceptual areas or domains and the interpretation of the defined relations should be the same in any monolingual ontology considered by the multilingual harmonized ontology. This is why, drawing as much as possible from the EuroWordNet lessons, we decided to address these issues at the very beginning phase of the BalkaNet project.

1. Introduction

BalkaNet [1] (Stamou et al, 2002) is an EC funded project (IST-2000-29388) that aims to develop in accordance with EuroWordNet philosophy a core multilingual resource for the following Balkan languages: Greek, Turkish, Romanian, Bulgarian, Czech and Serbian. As in EuroWordNet, the monolingual lexical ontologies are projected onto an interlingual set of concepts (ILI), the correspondences being established by means of complex equivalence relations (eq-synonymy, eq-near-synonymy, eq-has-hyperonym, eq-has-hypernym etc).

The requirements in building a multilingual ontology of the EuroWordNet kind are frequently conflicting (Rodriguez et al, 1998) and if not considered in the first stages of the project, later harmonizing might be extremely difficult if possible at all. To ensure as early as possible usability, the incrementally developed lexical stock of each individual wordnet, should cover the most frequent vocabulary of the language. On the other hand, given that this is a multilingual lexical resource, special care should be addressed to the compatibility problems. Specifically, there are two main compatibility issues to be considered: there should be a cross-language conceptual coverage, meaning that each monolingual lexicon should globally deal with the same conceptual areas or domains and the interpretation of the defined relations should be the same in any monolingual ontology considered by the multilingual harmonized ontology. This is why, drawing as much as possible from the EuroWordNet lessons, we decided to address these issues at the very beginning phase of the BalkaNet project.

The first part of the paper will address the approach we took for the selection of the initial lexical stock to be included into the Romanian core wordnet so that to observe multilingual design criteria and cross-language compatibility issues. The synsets (in two or more languages) that are mapped onto the same ILI concept are implicitly semantically linked. The nature of these cross-lingual semantic links, which we call *translational links*, depends on the links between the ILI concept and the synsets in the monolingual wordnets. One way to check consistency of the ILI projection of the individual wordnets is comparing the translation links with the translation equivalents licensed by a parallel corpus. This issue will be discussed in the second part of the paper.

2. An overview of the language resources

The Romanian wordnet started, as in the case of other languages in this project, from scratch. However, in order to ease the work and make the process as reliable as possible we built on various valuable language resources and several tools we developed for their exploitation. In the following there is a brief account of these building blocks, each of them being largely described elsewhere.

2.1. Corpora

Within the Multext-East and TELRI European projects (Erjavec et al. 1997), (Dimitrova et al., 1998), (Tufiş, Bruda, 1997), (Tufiş et al. 1997, 1998, 1999) there were created one 7-language heavily annotated parallel corpus based on Orwell's famous novel "1984" and one 25-language heavily annotated parallel corpus based on Plato's "The Republic". The annotation initially used was TEI compliant, but it was later on converted into CES (Ide, 1998). These are two relatively small corpora (about 110,000 tokens in each language) but given the accuracy of tagging and interlingual sentence alignment (hand validated) they were extremely useful for various applications ranging from building language models for morpho-syntactic tagging (Tufiş, 1999) and document classification (Tufiş et al., 2000) to automatic sense

[1] Further information can be obtained from the project's web site http://dblab.upatras.gr

discrimination (Erjavec et al., 2001). Besides the multilingual corpora we constructed two other much larger monolingual corpora: a literary corpus based on various novels (containing about 1,500,000 tokens) and a journalistic corpus (containing more than 100,000,000 tokens). Both corpora were automatically tokenized, tagged and lemmatized.

2.2. Lexicons and dictionaries

One delivery of the Multext-East project was a large wordform lexicon (more than 450,000 entries) containing triples <wordform, lemma, morpho-syntactic_code>. The encoding used in this lexicon is compliant with the Eagles recommendations for morpho-syntactic annotation and is largely documented in (Tufiş et al. 1997).

The reference dictionary we used for our analysis is The Explanatory Dictionary of Romanian (DEX,1996), work of the Romanian Academy Institute of Linguistics. This most authoritative lexicographic source for contemporary Romanian was partially digitized and converted into a lexical database (XML encoded) by RACAI under the European Project CONCEDE (Tufiş et al.1999). This core XML-dictionary has been extended to the full content of the printed dictionary by a follow-up project funded by Romanian Academy.

Another extremely useful lexical resource we relied on was the Romanian Dictionary of Synonyms-RDS (Seche, Seche 1997), which was transposed into electronic form by the NLP group at the University A.I. Cuza din Iaşi. The electronic form of RDS has been converted into an XML format so that the same query interface we developed for DEX works also with RDS.

From the multilingual parallel corpora mentioned before and using our translation equivalents extraction program (Tufiş, Barbu 2000, 2001a, 2001b) we constructed a bilingual Romanian English dictionary (also XML-encoded). This bilingual lexicon has been hand validated and extended with new entries from several public domain sources.

Finally, an extremely valuable resource was the ILI of the EuroWordNet, exported in XML format by means of the VisDic editor produced by the Masaryk University of Brno (Pavelek and Pala, 2002).

All these resources have been integrated by means of a series of tools developed for the purpose of the BALKANET project. They are user-friendly and allow for editing and mapping the Romanian synonymy series in RDS to the sense definitions in DEX and ILI records from EuroWordNet. The output of these tools is further subject to primary local consistency checks (such as detecting word sense appearing in more than one synset) and generated as an XML-encoded file appropriate for import in VisDic. We will provide a brief overview of these tools in Section 5.

3. Lexical stock selection

In order to ensure practical utility for the core wordnets to be delivered by the BALKANET project and to facilitate further extensions towards as large as possible coverage for the languages concerned, the project consortium decided to start the development process with a common set of concepts likely to be lexicalized in all the project languages. This special set of concepts, called *Base Concept* Set, was selected from the EuroWordNet interlingual index for reasons convincingly argued in (Vossen, 1998). The Base Concept Set contains 1310 concepts, each of them being attached a gloss and a Top Ontology Description (see Vossen, 1998). All project partners developed in a harmonized way the synsets in their languages corresponding to the Base Concepts. After this step, the monolingual wordnets will be further developed in a top-down approach starting with the synsets already mapped onto the Base Concepts.

Let us give a few definitions for some notions that will be used in the following.

When we place ourselves in a monolingual environment we speak about *senses*, *meanings* and *synsets.* A word has one or more *senses.* A sense refers to one *meaning.* In EuroWordNet the senses of a word are numbered according to their frequency and a sense of a lemma is denoted by appending the sense number to orthographic form of the lemma in case. A set of such numbered senses (eg. action2 activity1 activiteness1) referring to the same meaning is called a synset, which itself stands as a denotation of the common meaning of the senses in the synset. A meaning has a gloss that obviously applies for all senses in a corresponding synset.

When we want to abstract away from one language, we speak about the *concepts* referred to by the *word meanings*. So, we may speak about concepts with or without the reference to a specific language. Therefore, in trying to establish cross-lingual dependencies, via an interlingual index, it is convenient to refer to the entities used for this purpose as *concepts.* A concept is a language independent cognitive construct, which in EWN is always lexicalized at least in one language. A concept is further refined in terms of basic semantic distinctions (semantic features, sometimes referred to as semantic fields) so that one could speak about concept clustering along the basic semantic features.

According to these definitions we will use the term *Base Meaning* to refer to a basic (language specific) meaning in terms of which other word meanings can be defined and *which is directly mapped on a Base Concept*.

In EuroWordNet, and thus in BALKANET, ILI is defined as an unstructured collection of concepts represented by records of the form (<ILI-index> <ontological description> <gloss> {<domain>}). The initial ILI has been constructed from Wordnet1.5 and thus the gloss of each concept has been imported directly from the English synset referring to the meaning conceptualized in ILI.

According to the aims of the project regarding the interlingual coverage, language representativity, maximum usage of the core wordnet and scalability we started a series of quantitative analysis on a very large corpus made of several novels and a collection of journalistic texts, collected from the web. The corpus (containing more than 100 million words) was automatically tagged, lemmatized and the content words of interest (common nouns, verbs, adjectives and adverbs) were counted and sorted according to their frequency. We extracted this way, a list of more than 30,000 Romanian lemmas. Based on the frequency in the running texts, this list was divided into three parts, corresponding to the first 10,000 most frequent lemmas (I), the next most frequent 10,000 lemmas (II) and rest of the lemmas (III).

In deciding which is the most important subset of a lexical stock for a language, the frequency in running texts

is considered by many lexicographers to be a very subjective criterion. Among the strongest arguments they would come with is the volume and representativity of the texts included into the corpus subject to the quantitative analysis. With more and more texts available on the net, the size of the data is not anymore a significant issue, but the representativity remains a systematic complain. The exact definition of what representative texts should be included into a corpus for quantitative data analysis is a long-standing debate and we won't get into this. Considering that our data consisted, almost entirely, of journalistic texts, the representativity issue could certainly be raised. The Frequency Dictionary of Romanian Words– FDRW (Julliand et all., 1965) published long time ago, based on a balanced corpus of 500,000 words of Romanian literature, legal texts, poetry and journalism contains a list of most frequent 5,000 lemmas. In spite of being quite contested, it is still used by many Romanian linguists as a reference. The comparison we made revealed that most of the 5000 words in FDRW were also in our list, although not with the same frequency ranges.

As frequency in running texts is a disputable criterion for deciding what words should be encoded into a core dictionary/thesaurus/ontology we considered that this criterion should be complemented with others, less controversial in the world of traditional lexicography.

Among the criteria one could find pleas for, we opted for two that we could easily turn into operational selectors. The one is the number of senses a headword would have in a reference dictionary. The second one is the number of word definitions that use the headword in case. A third criterion, not considered yet, might be the number of derivatives of a given headword (this last criterion is preferred by most Romanian etymologists).

In this phase of the BALKANET project we concentrated our attention to the Romanian nouns and the experimental data reported below refers to nouns. Since the technical procedures do not depend on the specific part of speech, the same would apply for verbs, adjectives and adverbs.

Considering only the first two frequency ranges described above (the first most 20,000 words in the journalistic corpus) we extracted from our Explanatory dictionary more than 8000 entries for nouns and nominal compounds (accounting for almost 35,000 senses) so that the definitional productiveness DP (the number of sense definitions a noun participates in) was at least 3. The list was sorted according to the definitional productivity.

Noun	**Definitional productivity**	**Number of definitions**	**FRECV$_{range}$**
acțiune	2279	13	I
persoană	1979	9	I
parte	1882	94	I
formă	1286	21	I
obiect	1204	16	I
fapt	1044	11	I
apă	743	29	I
• • •	• • •	• • •	• • •
rasism	3	1	II

Table 1: scoring the headword candidates

For all these nouns we extracted EN translations from our translation equivalence dictionary. The procedures for automatic extraction of translation equivalents from parallel corpora as well as the sense discrimination procedure are largely described in (Tufiş&Barbu, 2001a,b), (Erjavec et al, 2001). As the translation equivalents found by our extractor are limited by the available parallel corpora we have, provisions were made for automatic updating of the Ro-En dictionary with web resources.

All pairs containing an English word (or a synonym of it) in the English synsets corresponding to the base concepts were also associated with the corresponding top-ontology description. Practically for all English words corresponding to the base concepts there were found translations in our translation lexicon and these translations appeared in the upper top of our 8000-noun list. Those few EN nouns not translated in our lexicon were given manual translations. Because our translation equivalence lexicon is based on sense equivalence in context, transferring the ontological description from one EN word to its equivalent translation was considered to be a legitimate option. Thus, at the end of this step we collected a list of Romanian nouns associated with one or more English translations out of which at least one was present in the base concept list. Each such an association was further enriched with additional information extracted from other resources:

a) the RO word was attached with all its definitions extracted from the Explanatory Dictionary of Romanian;

b) the EN word was attached with its entry in the WordNet1.5

The Romanian Dictionary of Synonyms (RDS), digitized and encoded as an ACCES database by University A.I. Cuza of Iaşi, was used to extract the synonymy series for the selected RO words. In RDS some members of the synonymy series are provided with usage information (old, regionalism, specific area of usage, domain, etc). Preliminary discussions lead to the idea to eliminate all the words marked as such (based on the assumption that we would like to construct a lexical stock for general use in contemporary Romanian). However, if later on this filtered out words (together with their usage information) would be necessary, their recovery was ensured. The synonymy series were taken as possible Romanian synsets and added to the RO-EN associations described above.

We have thus assembled the basic linguistic material that the lexicographer should use in making the decisions (linking) necessary for building the noun subset of the core Romanian wordnet. All this information is currently available in a java-based editor, showing in different frames, the following information (see figure 1):

- the list of the base concepts (upper-left frame), identified by the ILI record and an English word in the synset mapped on this concept (ex. *life_3_03941565-n*)
- the synset (life_3 living_1), its gloss and top-ontology description, possible translations and association boxes (right-upper frame)
- the numbered sense definitions from the Explanatory Dictionary of Romanian for the selected translation (left-lower frame);
- synonyms of the selected Romanian translation word (right-lower frame)

- pop-up menus for selecting the relevant sense numbers and the equivalence relation to the ILI concept.

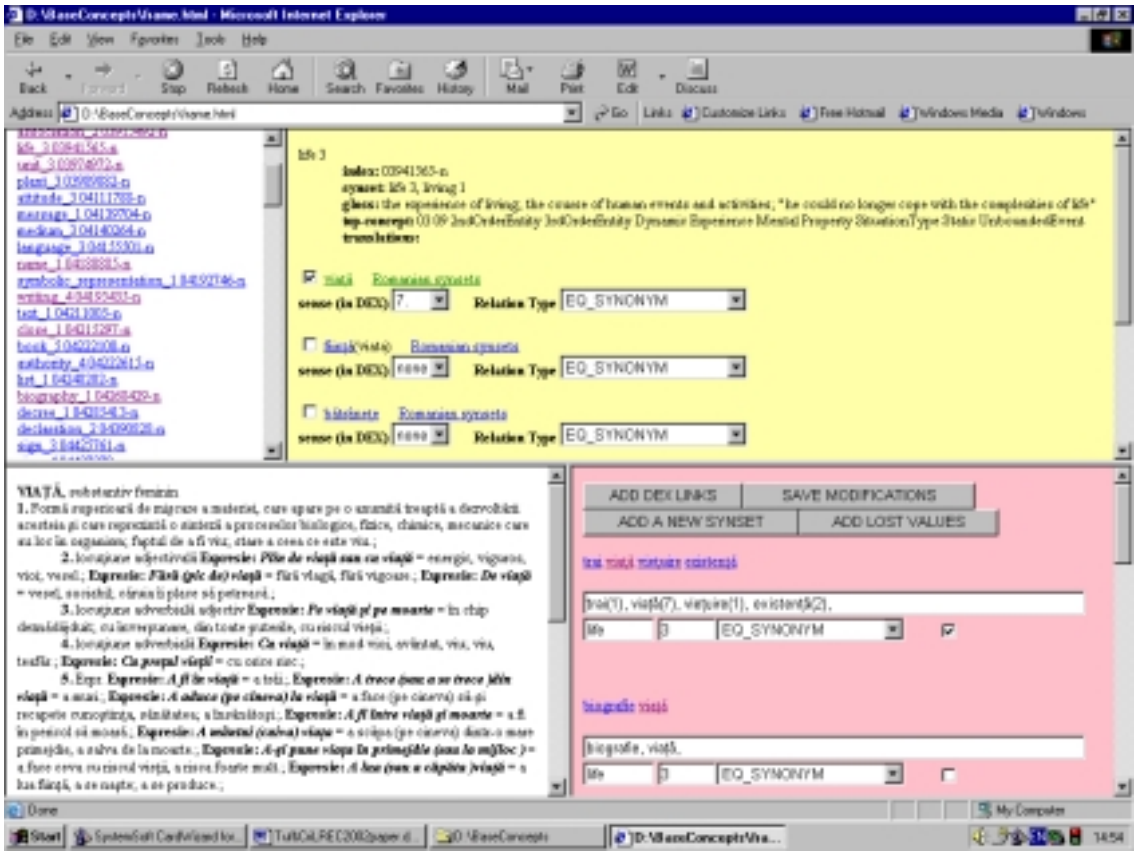


Figure 1: The editor for building synsets for the base meanings

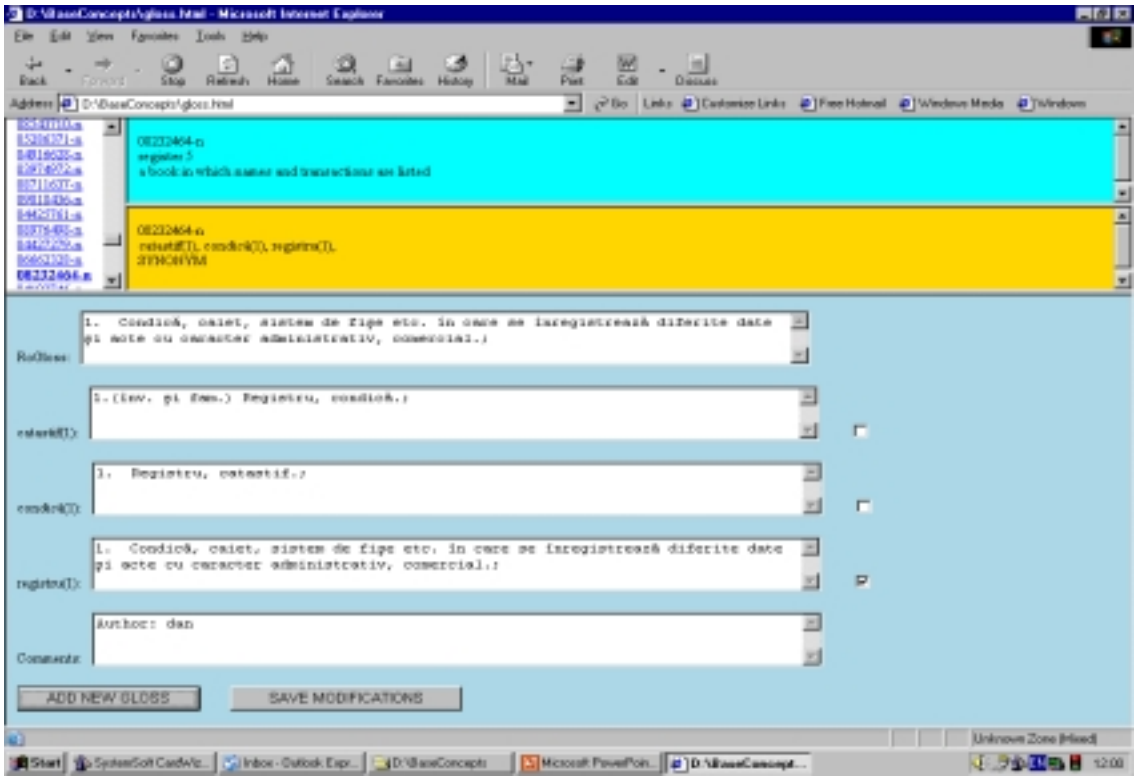


Figure 2: The editor for gloss assignment

The editor has been instantiated into 10 differently populated copies, each containing a different set of base concepts. Each incarnation of the editor has been given to a different expert who was in charge of building his/her set of Romanian synsets and map them onto the appropriate base concepts. When this building phase was finished we performed a few simple error-checking such as:

- all literals appearing in a synset should have attached a sense number
- no sense (literal and sense number) should appear in two or more synsets
- each synset should have an equivalence relation to a unique base concept.

Once the synsets were constructed and mapped onto base concepts, the second phase was to add a Romanian gloss to each Romanian synset. In the vast majority of cases, the definitions extracted from DEX corresponding to the senses in a synset were different in wording so, the lexicographers had to chose the best definition, closest to the definition of the corresponding base concept. The Figure 2 shows that the base concept 08232464-n corresponding to the 5th sense of the English word *register* (a book in which names and transactions are listed) corresponds in Romanian to the synset (catastif_1 condică_1 registru_1). The selected senses for the three Romanian words have in DEX different definitions. By checking the box to the right of the third definition (lower frame in Figure 2) the lexicographer decided that the definition given to *registru_1* is the one to be attached to the synset.

It is worth mentioning that during the gloss assignment phase it became apparent that several synsets were not correct, requiring modifications. In some cases, the Romanian Explanatory Dictionary includes under the same definition two senses that are differentiated in ILI as two distinct concepts. In such cases, the general strategy was to split the Romanian definition and attach the relevant part as a gloss.

4. A proposal for cross-lingual validation of the ILI mapping

As we said before, one of the main objectives of the BALKANET project (which adopted a merge model approach) is to ensure as much as possible overlap between the concepts lexicalized in the concerned languages. A significant overlap may be hampered either by conceptually different lexical stocks for the different languages or by inconsistent projection of the monolingual concepts onto the ILI concepts. In order to ensure conceptual similarity for the lexical stocks across various languages, the development of the monolingual ontologies started in two different, but convergent ways: the minimalist one was to provide direct translations of the EuroWordNet Base Concept Set; the second way (language-centric) was to produce a ranked list of most important (according to prescribed lexical criteria) words in each language and to include in the monolingual wordnets at least those words, the meanings of which would cover the Base Concept Set. Irrespective of the approach taken towards ensuring lexical stock similarity across languages, we had to consider means for automatic check of the correctness of the mapping of the monolingual synsets over the ILI concepts. To this end we will describe in some details a proposal for an automatic consistency checking.

Our approached is based on the notion of translation equivalence over bitexts, on bilingual lexicons automatically extracted from parallel corpora (Tufiş, Barbu, 2001 a,b) and on sense disambiguation (Erjavec et al., 2001).

The parallel corpus we used in our experiments is the "1984", based on Orwell's famous novel, developed in the MULTEXT-EAST project, further cleaned up in the TELRI and CONCEDE projects. The corpus contains professional translations of the original novel in 6 languages (Bulgarian, Czech, Estonian, Hungarian, Romanian and Slovene), all aligned at the sentence level to the English original. Each monolingual part of this 7-language parallel corpus is segmented, tagged and lemmatized and also carefully hand validated.

From the 6 (integral) bitexts (CEE language texts aligned to the EN original) there were extracted bilingual lexicons (XX-EN, with XX one of the six CEE languages) and furthermore a 7-languages lexicon with EN as a hub. By removing all the non 1-1 alignments in the bitexts and using the EN sentence Ids as anchors, a partial (about 92% of the whole text) 7-lingual 1-1 alignment (EN-BG-CZ-EE-HU-RO-SI) was computed. The 7-language aligned corpus allows for extracting any of the 21 possible (partial) bitexts. A number of 104 nouns appearing in the English part of the multilingual corpus (altogether 3316 instances were hand annotated and used as a gold standard for our sense clustering algorithm (Erjavec, 2001).

As BG, CZ and RO are languages of the BalkaNet project from the present data, our methodology could be used for checking the ILI-mapping consistency for any of the RO-EN, RO-CZ, RO-BG, EN-CZ, CZ-BG and BG-EN pairs of wordnets. In the current phase of the project we are able to consider only the interlingual mapping of the base concepts. Let us generically denote the language pairs subject to checking as XX-YY. The basic methodology is as follows:

1) From the XX-YY bitexts we extracted the XX-YY lexicon (http://www.racai.ro/~tufis/BilingualLexicons/AutomaticallyExtractedBilingualLexicons.html). The bilingual lexicon contains not only the translation pairs but also, for each entry the aligned sentences that licensed the translation equivalence relation. This lexicon is purged so that it contains only words that have (in the respective monolingual wordnets) at least one sense mapped on a base concept set. Put it otherwise, any pair (W_{XX} *translated as* W_{YY}) of the purged lexicon has the property that W_{XX} or W_{YY} or both have at least one sense in the language-specific base meaning set.

2) Let it be (W_{XX} W_{YY}) a translation equivalent. Let us denote with S_{WXX} the synsets in language XX containing the W_{XX} word (actually one sense of it) and S_{WYY} the synsets in language YY containing the W_{YY} word (actually one sense of it). Starting in the XX monolingual wordnet from the synsets in S_{WXX}, via ILI, one ends in the YY monolingual wordnet with the XX-synsets having translation links to YY-synsets. Let us call this set as S'_{WYY}. S_{WYY} and S'_{WYY} should have at least one synset in common. Please note that if the intersection of the two sets of synsets is non-empty, the described procedure ensures semantic tagging of the (W_{XX} W_{YY}) pair with one or more ILI-concept tags. If the intersection contains exactly one synset, its corresponding ILI record-number

could be used to semantically tag both W_{XX} and W_{YY}. With intersection containing more synsets, we still are able to reduce the semantic ambiguity of the considered words. In case the intersection is empty, we might have one of the following possible explanations:

2.1) (W_{XX} W_{YY}) is not a valid translation pair; by checking the sentences that licensed the extraction of this translation pair one could confirm or refute this possibility; please note that an error here might be due to the extraction algorithm or to a problematic human translation (for instance it is not uncommon that even professional translators would sometimes translate one word by a non-eq-synonym for various reasons like contextual semantic gaps or stylistic preferences)

2.2) (W_{XX} W_{YY}) is a valid translation pair and the two words share a meaning assigned to a concept which is not in the base concept set.

2.3) the interlingual mapping of the W_{XX} and W_{YY} is "wrong"; being "wrong" might be a real mapping error in the XX or YY language (or in both) or it might be motivated by a lexical gap in one of the languages concerned (or both); the lexicographer might have overcome the lexical gap by using a complex equivalence relation (not the eq-synonym); in the second case, one might get insights on possible concept clustering at the ILI level (creating so-called *soft-concepts*).

We claim that this procedure allows us to estimate both the cross-lingual coverage and the correctness of the interlingual mapping of the two considered monolingual wordnets. The procedure allows not only for estimation but also for pinpointing the incomplete or missing synsets as well as inconsistencies in mapping the synsets onto ILI concepts and gives hints on soft-concept clustering.

4.1. Condiments, spices, sauces and other ingredients

Let us consider the fragments of the Ro-Wordnet and WN1.5 shown in the Figure 3. The arrows represent hyponymy relations in the two wordnets. The gray heavy lines represent translational links between the synsets in the two languages, meaning that the respective synsets are mapped onto the same ILI concept. The heavy dashed line represents a translational link that is reported as wrong during the cross-validation of the two wordnets. The reason for this comes from the violation of what we called the *hierarchy preservation principle*. The inconsistency is signaled because in language RO the hierarchical relations (hyponym) between $^{M}mirodenie_{RO}$ H $^{M}condiment_{RO}$ as well as $^{M}ketchup_{RO}$ H $^{M}sos_{RO}$ are not verified in language EN by the equivalent pair meanings ($^{M}spice_{EN}$ $^{M}condiment_{EN}$) and ($^{M}ketchup_{EN}$ - $^{M}sauce_{EN}$)(in EN they are sisters). If the structuring in WN1.5 is taken to be the Truth, this example shows that *the hierarchy preservation principle* is not true. On the other hand, if it would be reasonable to consider that WN1.5 is amendable (for instance making $^{M}mustard_{EN}$ and $^{M}ketchup_{EN}$ direct hyponyms of $^{M}sauce_{EN}$) then the *hierarchy preservation principle* might be a very powerful consistency check.

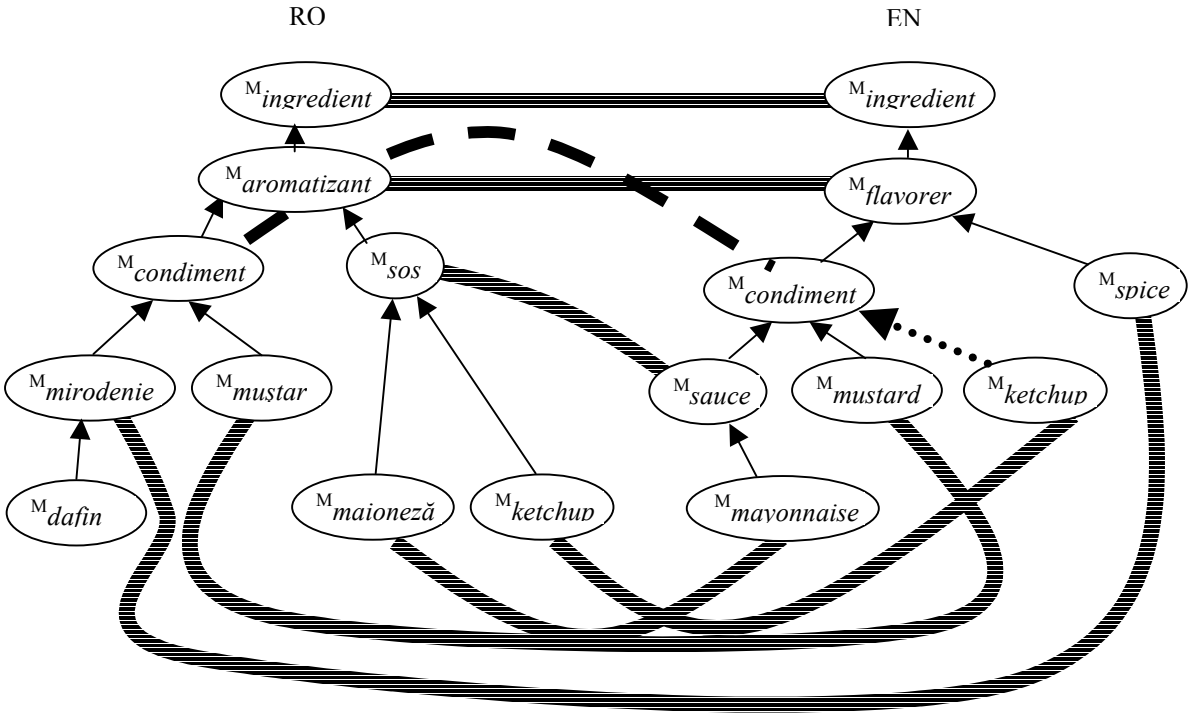


Figure 3: Translational links and consistency checks

5. Conclusions and further work

The approach on consistency checking based on translation equivalents in multilingual parallel corpora has some methodological similarity with (Resnik et al., 1999) on the multilingual corpus built up from many translations of the Bible. Speaking about useful sense distinctions (for machine translation for instance) Resnik (personal communication) identifies *strong sense distinctions* of one word in a source language as those that are lexicalized as

different words in the target languages. When some senses carried by a source word are found in a target word the distinction between them is called a *light sense* distinction. In the area of machine translation trying to disambiguate among light distinctions is not a very productive enterprise and therefore being able to identify, for a given pair of languages, which are the strong/light sense distinction might be extremely useful for machine translation. Our approach could be used to enhance the strong/light dichotomy with a third dimension: *fuzzy sense* distinction. This term is strongly related to that of *soft concept* used in EuroWordNet for clustering different ILI concepts that are lexicalized in two or more languages by words considered to be legitimate translations of one another.

In the next phase of the project, in order to extend the monolingual Romanian wordnet up to the level of the promised size, our strategy will be language-centric meaning that the new entries will be the top ranked words selected from our noun/verb/adjective/adverb lists sorted as described in the section 3.

6. References

Bloksma L., Diez-Orzas and Vossen P. (1996) The User Requirements and Functional Specification of the EuroWordNet-project *EWN-deliverable D.001*, LE-4003

DEX (1996). Coteanu, I., Seche, L., Seche, M. (coord.). Dicţionarul Explicativ al Limbii Române, Ediţia a II-a, *Univers Enciclopedic*, Bucureşti, 1996

Erjavec T., Ide N., Tufiş D.(1997) Encoding and Parallel Alignment of Linguistic Corpora in Six Central and Eastern European Languages" in Michael Levison (ed) *Proceedings of the Joint ACH/ALL Conference* Queen's University, Kingston, Ontario, June 1997 (also on http://www.qucis.queensu.ca/achallc97)

Erjavec T., Ide N., Tufiş, D.(2001) *Automatic Sense Tagging Using Parallel Corpora*, in Proceedings of the 6th Natural Language Processing Pacific Rim Symposium, Tokyo, Japan, 27-29 November, pp. 212-219, 2001

Ide, N. (1998) *Corpus Encoding Standard: SGML Guidelines for Encoding Linguistic Corpora* First International Language Resources and Evaluation Conference, Granada, Spain. See also http://www.cs.vassar.edu/CES/.

Julliand, A., Edwards P.M.H, Julliand I. (1965). The Frequency Dictionary of Rumanian Words. *Mouton & CO.,* London-The Hague-Paris, 1965

Miller G.A., Beckwidth R., Fellbaum C., Gross D., Miller K.J. (1990) "Introduction to WordNet: An On-Line Lexical Database" 1990 In International Journal of Lexicography, Vol. 3, No. 4 (winter 1990), pp. 235-244

Pavelek T., Pala K. (2002) *VisDic : A new Tool for WordNet Editing* in Proceedings of the 1st International Wordnet Conference, Mysore, January 21-25, 2002

Resnik, P. (1999) Disambiguating Noun Groupings with Respect to WordNet Senses, in S. Armstrong, K. Church, P. Isabelle, S. Manzi, E. Tzoukermann and D. Yarowsky (eds.), *Natural Language Processing Using Very Large Corpora.* Kluwer Academic Publishers, 1999, pp. 77-98.

Resnik, P., Broman Olsen M., Diab M.(1999) The Bible as a Parallel Corpus: Annotating the `Book of 2000 Tongues', *Computers and the Humanities*, 33(1-2), pp. 129-153, 1999.

Resnik P., Yarowsky D. (2000) Distinguishing Systems and Distinguishing Senses: New Evaluation Methods for Word Sense Disambiguation, *Natural Language Engineering* 5(2), pp. 113-133.

Rodriguez, H., Climent, S., Vossen, P., Bloksma, L., Peters, W., Alonge, A., Bertagna, F., Roventini, A.(1998) The Top-Down Strategy for Building EuroWordNet: Vocabulary Coverage, Base Concepts and Top Ontology. In Piek Vossen (ed.) *EuroWordNet: A Multilingual database with lexical semantic networks*, Computers and Humanities*,* Vol. 32, Nos 2-3, 1998

Seche L., Seche M.(1997) *Dicţionarul de sinonime al limbii române.* Univers Enciclopedic, Bucureşti, 1997

Stamou S., Oflazer K., Pala K., Christoudoulakis D., Cristea D., Tufiş D., Koeva S., Totkov G., Dutoit D., Grigoriadou M..(1997) BALKANET A Multilingual Semantic Network for the Balkan Languages, in *Proceedings of the International Wordnet Conference*, Mysore, India, 21-25 January 2002

Tufiş D., Şt. Bruda (1997) Structure Markup in CES and Preliminary Statistics on Romanian Translation of Plato's "Republica", *Proceedings of International Seminar on Encoding,* Ljubliana, February, 1997, also in *TELRI News*, nr. 5, May, 1997.

Tufiş, D. Tiered Tagging and Combined Classifiers In F. Jelinek, E. Nöth (eds) *Text, Speech and Dialogue*, Lecture Notes in Artificial Intelligence 1692, Springer, 1999

Tufiş D., Barbu A.M., Pătraşcu V., Rotariu G., Popescu C. (1997). Corpora and Corpus-Based Morpho-Lexical Processing, in Tufiş D., P. Andersen (eds.) *Recent Advances in Romanian Language Technology*, Editura Academiei, 1997.

Tufiş, D., Rotariu, G., Barbu, A.M. (1999) TEI-Encoding of a Core Explanatory Dictionary of Romanian. In Kiefer, F. and Pajzs J. (eds.) *Papers in Computational Lexicography*, Hungarian Academy of Sciences, 1999, pp. 219-228

Tufiş D., Popescu C., Roşu R.: Automatic classification of documents by random sampling in *Proceeding of the Romanian Academy*, Series A, vol 1, no. 2, p. 18-28, 2000

Tufiş, D. (2000). Blurring the distinction between machine readable dictionaries and lexical databases. R*esearch Report, RACAI-RR56*, 1999

Tufiş, D. (2001) Romanian wordnet of BALKANET: selecting the lexical stock. R*esearch Report, RACAI-RR68*, October 2001

Tufiş, D, Cristea D. (2001) *Methodological issues in selecting the candidate concepts to be included into the Romanian Wordnet.* Progress Report on BALKANET project. November 2001

Tufiş D., Barbu A.M.(2001a) *Computational Bilingual Lexicography: Automatic Extraction of Translation Dictionaries*, in International Journal on Science and Technology of Information, Romanian Academy, ISSN 1453-8245, Vol.4, No.3-4, 2001, pp.325-352

Tufiş D., Barbu A.M.(2001b) *Extracting multilingual lexicons from parallel corpora*, in Proceedings of the ACH-ALLC conference, New York, 12-17 June, 2001.

Vossen P. (ed.) (1998) "A Multilingual Database with Lexical Networks", Kluwer Academic Publishers, Dordrecht

Semi-automatic Development of the Hungarian WordNet

Gábor Prószéky*, Márton Miháltz**

*MorphoLogic, Késmárki u. 8, H-1118 Budapest, Hungary
proszeky@morphologic.hu
**Eötvös Lóránd University, Institute of Informatics, Budapest, Hungary
mmarcy@inf.elte.hu

Abstract

Construction of the Hungarian WordNet began in 2000. The project presently focuses on the nominal part. Our principal approach is to use Princeton WordNet as the basic structure, to which Hungarian nouns are attached. We are applying two methods to accomplish this: manual disambiguation for the more abstract levels, and automatic methods, including heuristics developed by earlier projects, in order to attach the remaining more specific senses. Results from these methods are integrated into a core structure, which will be enriched using further electronic linguistic resources.

1. Introduction

The construction of the Hungarian WordNet has started from scratch, unlike many components of the EuroWordNet project (Vossen, 1999), whose creators could rely on already existing lexical resources (Kunze et al., 1998). We employed the initial hypothesis that nominal hierarchies in English and Hungarian should be similar, at least for certain domains. This enabled us to attach Hungarian nominal entries of a Hungarian-English bilingual dictionary to Princeton WordNet 1.6 (WN) synsets. In this way, the English nominal hierarchy of WN serves as a skeleton structure to support the construction of the core Hungarian nominal WordNet. This approach was also used in the initial stage of the construction of the Spanish and Catalan WordNets (Farreres et al., 1998; Atserias et al., 1997). Furthermore, examining the Hungarian nominal taxonomies extracted from a Hungarian monolingual dictionary, we have found that hierarchies for the specific nominal domains (nouns denoting objects) tend to be similar to those found in WordNet.

Linking Hungarian words to WN synsets is accomplished in two ways. First, a software environment has been created to support the manual disambiguation of Hungarian nouns against WordNet. This is a top-down procedure advancing from the abstract to more specific levels in the WordNet hypernym structure, resulting in the manual construction of the more abstract levels of the core Hungarian WordNet.

Secondly, various heuristics, mostly developed in earlier projects, are applied to produce sets of candidate links between Hungarian nouns and WN synsets automatically. These methods rely on information found in the bilingual and monolingual dictionaries, plus the information already available from the parallel manual disambiguation procedure. Finally, results of all the different methods are manually checked and integrated.

Attaching further nominal entries from a larger bilingual dictionary, a thesaurus, and entries and definitions (serving as glosses) from a monolingual dictionary will enrich the resulting skeleton Hungarian WordNet structure, connected to the English WN.

In the following section, we will give a review of the electronic resources we use in our work. Section 3 gives an overview of the various automatic and manual methods used in the project. Integration of the information from different sources and the possibilities for further extensions are also discussed. Finally, our conclusions are provided in Section 4.

2. Acquiring taxonomies from various dictionaries

We have several electronic resources at our disposal: English-Hungarian bilingual dictionaries, a monolingual Hungarian explanatory dictionary, a Hungarian Thesaurus, and, of course, WordNet 1.6. MorphoLogic's English-Hungarian bilingual electronic dictionary contains entries for 17,801 Hungarian nouns with 12,440 English translations included in WordNet. The dictionary has been converted into a database of English-Hungarian word pairs with symmetrical translation relations (Prószéky et al., 2001). The entries of the Hungarian side constitute the basic set used for the various attachment procedures (see: Section 3).

A significantly enlarged version of the *English-Hungarian Dictionary* (Országh–Magay, 2001) will be used for further improvement of the Hungarian WN structure. It contains over 150,000 Hungarian entries, with English translations covering more than 80% of WordNet's entries. An electronic version of the Hungarian explanatory dictionary *Magyar Értelmező Kéziszótár* (ÉKSz) (Juhász et al., 1972) has been converted into XML format. This dictionary contains 42,942 nominal entries, corresponding to 64,146 definitions. 31,023 of them are annotated with usage codes, representing either the semantic domain (sport, medicine, science, religion etc.), or the language usage (technical, slang, vulgar, intimate, etc.). Through the smaller bilingual dictionary, 10,507 headwords have English translations in WN. We also have at our disposal a Hungarian electronic thesaurus. The *Magyar Szókincstár* contains 25,500 entries with synonyms and 14,400 entries with antonyms. Entries are linked to separate sets of synonyms for various senses. Most of the synonym and antonym words are annotated with language usage labels.

To help the construction of the Hungarian nominal WN, information is acquired from the monolingual dictionary in several ways. First, programs were developed to parse each dictionary definition and extract semantic information. In 83% of all the definitions, genus words were identified, which can be accounted for as hypernym approximations of the corresponding headwords. For example, the following ÉKSz entry will tell us that the *koala* is a kind of *mammal*:

koala: *marsupial* ***mammal*** *resembling a bear, native in Australia*

In about 1,700 cases, the identified genus word was either a group noun, or a word denoting a "part" relationship. Let us consider as an example the ÉKSz entries for *alphabet* and *face:*

alphabet: *The* ***set*** *of letters used for…*

face: *The* ***part*** *of the head that…*

Using morpho-syntactic information, the meronym or holonym word (in the example above: *letter, head*) could be identified instead of a genus word. This method provided holonym/meronym word approximations for 2.7% of all the headwords (only distinguishing between "part" and "member" subtypes of holonymy, as opposed to the 3 types represented in WN (Miller, 1990)). A further 13% of the definitions consisted only of a single noun. These are synonyms for the corresponding sense of the headwords, which are mostly rare variants or compounds.

These simple methods provided us with hypernym, holonym and synonym words for 98.9% of all the nominal dictionary entries. Such information extracted from machine-readable dictionaries can be used to build hierarchical lexical knowledge bases (Copestake, 1990), or semantic taxonomies (Rigau et al., 1998). The extracted genus word approximations can yield a hierarchical taxonomy of the nominal dictionary entries, organized by hypernym relations, providing a very versatile resource for the construction of our Hungarian nominal WN. However, in order to get hypernym relations between senses, the identified genus words have to be disambiguated, which means the hypernym sense must be separated from the senses corresponding to the genus word.

We are experimenting with several heuristics, relying on the work by Rigau et al. (1997) and Copestake (1990) to achieve an automated process of genus word disambiguation. About 70% of the genus terms are monosemous in the monolingual dictionary. In these cases the hyponym senses are attached to them directly.

Another heuristic utilizes the usage codes available for about 30% of the candidate senses Semantic codes, if available, can be tested for compatibility between the hyponym and the candidate hypernym senses. The pragmatic codes are also put to use: senses annotated as slang, vulgar etc. are more unlikely to be used as genus terms.

A third heuristic assigns the first sense occurring in an entry, relying on the fact that senses are ordered by usage frequency, and the most used senses are more likely to be used as hypernyms.

A fourth heuristic tries to measure semantic similarity among definitions by means of determining the number of lemmas shared by both definitions.

A fifth heuristic will rely on the conceptual distance formula, which measures semantic similarity between concepts using WordNet as a hierarchical knowledge base (Rigau et al., 1997). Application of the conceptual distance formula is discussed in more detail in Section 3.2.2.

Each heuristic will assign a score for the candidate senses, and the ones bearing the highest score will be linked to the hyponym senses. As work is still in progress for the disambiguation, it is early to report on the precision of the algorithm. Moreover, considering reports on previous works, it is likely that further manual and automatic assortment and/or verification of the resulting hierarchies will be necessary in order to attain a well-structured taxonomy (Rigau et al., 1998).

Some sample subsections of the resulting taxonomies were examined in order to investigate semantic similarities and differences between the parallel structures of the Hungarian hierarchy and WordNet. The most frequent difference originates from the fact that the hypernym trees in WN are quite detailed, often having 7-9 levels, while the Hungarian hierarchies tend to be more shallow, usually consisting of only 3 or 4 levels. The situation seems to be similar to previous projects constructing lexical hierarchies from machine readable dictionaries, for example in the Czech WordNet project (Pala & Ševeček, 1999).

Based on the samples examined, besides the lexical gaps on both sides, the two hierarchies seem to differ most at the higher, most abstract levels, where the Hungarian taxonomies are often unelaborated or confusing, and containing circular references. Nevertheless, we have not found evidence strongly contrasting our basic hypothesis, and our approach of attaching Hungarian nouns to the WN hierarchy seems maintainable for the initial stage of our work.

On the other hand, these facts have encouraged us to start linking Hungarian nouns manually, starting from the topmost WN levels, and to apply automatic linking procedures for the more specific senses.

3. Manual and semi-automatic procedures

We are using both manual and semi-automatic techniques to achieve the task of linking Hungarian nouns to the WN synsets. The manual methods provide a framework of top-down construction of the Hungarian nominal WordNet. The automatic methods rely on the bilingual and monolingual dictionaries, and on the extracted semantic information, applying heuristics developed for the construction of the Spanish and Catalan WordNet (Farreres et al., 1998; Atserias et al., 1997). We chose to test these methods because the resources available to the Spanish and Catalan Research Group are closest to our available resources, considering the participants in the EuroWordNet project (Vossen et al., 1999).

The result of these methods will be evaluated manually, based on random samples. Then all the possible intersections of the sets of results produced by the different methods will also be evaluated, and only the results obtained by the combination that produces the highest accuracy will be considered. We follow this approach, described by Atserias et al (1997), in order to ensure the precision of the core Hungarian WordNet structure.

3.1. Manual disambiguation with the help of the web

A set of Internet-based software tools has been developed for manual disambiguation of the Hungarian nominal entries against WN. The use of the Internet makes it possible for our contributing experts to work independently.

For the users, the system offers a web page, over which the expert can answer questions provided by the central server maintaining the database. (Figure 1) xperts are exposed to dialog boxes: if the word in question does mean the concept outlined below by English synonyms and a definition, then the human expert is supposed to press the Yes button (Nagy, 2001).

3.2. Semi-automatic methods based on heuristics

There are three kinds of automatic linking methods, each relying on different kinds of resources.

The first group of heuristics relies on information found in the bilingual dictionary and the structure of WN, while the second type relies on the genus information extracted from the monolingual dictionary. These constitute heuristics described by Atserias et al. (1997), plus a technique of our own.

The third method relies on the links already produced by the manual linking procedure and the taxonomy acquired from the monolingual dictionary.

3.2.1. Methods relying on bilingual dictionaries

Of the 17,800 Hungarian nouns forming the initial set, about 7,000 have translations in English, each belonging to only one synset in WordNet. These nouns are classified into four groups, based on the nature of the Hungarian-English translation relationships (one-to-one, one-to-many, many-to-one or many-to-many). Then, for every noun in each class a hypothetical link is produced to the unique synset containing the translation(s). Atserias et al. (1997) report on different kinds of precision for the four classes, ranging from 85% to 92% correct connections. Based on preliminary investigations, the average amount of correct links produced seems to be somewhat lower in our case. This is probably owing to the fact that the bilingual dictionary often either refers to senses not found in WordNet, or provides translations that correspond to hyponym senses of the Hungarian noun.

For the Hungarian nouns with polysemous translations in WordNet, the *Variant Criterion* and the 4 *Structural Methods* are being applied. These heuristics try to find common information between the English translations and WN. The *Intersection Criterion*, for example, will assign a Hungarian word to a synset if the synset is shared by at least two of the word's translations. In the Spanish experiments, precision is reported to be between 58% and 85% for these criteria (Atserias et al., 1997).

3.2.2. Methods relying on monolingual dictionaries

The ÉKSz explanatory dictionary contains *Latin* equivalents for about 1,600 nominal entries. These are mostly names of animal and plant species, taxonomic groups, diseases and chemical substances. Since WN 1.6 is very elaborate on Latin translations for such nouns, this provides for a reliable way for the linking of the Hungarian nouns. This method produced links for a small set of about 1,200 Hungarian nouns and corresponding definitions to WN, with the rate of correct connections estimated over 90%.

The second type of our automatic methods that utilize the monolingual dictionary relies on the extracted genus information (see Section 2). Following Atserias et al. (1997), we are applying the *Conceptual Distance formula* for the English translations of each headword-genus, or headword-holonym word pair we identified in the dictionary. The Conceptual Distance formula, introduced by Agirre et al. (1994), selects those two closest concepts in WN which represent the two input words. In the case of headword-genus pairs, the hypernym structure of WN is used as a semantic network for the heuristic, while for the ÉKSz headwords with holonym/meronym word approximations, the structures determined by WN's holonym links are used.

The application of the Conceptual Distance formula not only produces candidate links for the Hungarian words, but can also be used as a heuristic in the sense disambiguation of the Hungarian genus words, thus contributing to the construction of the Hungarian nominal taxonomy (Rigau et al., 1997).

3.2.3. Using information from the manual disambiguation procedure

After the semantic taxonomy is extracted from the EKSz dictionary, it can be used in conjunction with the already available information gained from the previous steps and WordNet's structure to support the manual processing. The order of the manual disambiguation of Hungarian words nouns follows top-down order (starting with abstract senses) of the English WordNet's hierarchy. Thus, once a Hungarian word is linked to a WordNet sense, hyponym words of its various senses can be disambiguated automatically against WordNet synsets, making use of the parallel structures of WordNet and the Hungarian taxonomy.

For example, let us suppose that the Hungarian word *állat* (`animal') has already been linked (either manually or automatically) to the WordNet synset {*animal, animate being, beast, brute, creature, fauna*}. *Állat* has 3 different senses in the Hungarian taxonomy, one of which has a hyponym pointer to (a sense of) the word *ló* ('horse'). The word *ló* has 3 English translations in the bilingual dictionary, which belong to 8 different synsets in WordNet. In order to determine which of those 8 synsets should *ló* be linked to, Conceptual Distance (see Section 3.2.2) is calculated between {*animal, animate being,…*} and the 8 candidate synsets. The candidate synset {*horse*, *equus caballus}* will show the smallest distance from the hypernym synset {*animal, animate being,…*}, thus, *ló* (with the sense determined by the hypernym *állat*) can be linked to {*horse*, *equus caballus*} (Figure 2).

A threshold condition will also be built into the algorithm, which will prevent links to existing but incorrect WordNet senses (i.e. in cases where a Hungarian word has a hyponym sense that does not have an equivalent meaning in WordNet).

3.3. Further steps

After the linking of the Hungarian entries of the bilingual dictionary to the WordNet semantic nodes is complete, further methods can be applied to enrich the resulting skeleton structure.

One way is with the aid of the *Magyar Szókincstár* thesaurus. With semantic disambiguation to decide which sense of a word the synonyms express, synonyms can be added to the Hungarian-English synsets. Antonyms to Hungarian words can also be added (antonymy is a lexical relation, therefore pre-existing WordNet antonym links cannot be used).

Daudé et al. (1999) describes a method for mapping multilingual hierarchies to WordNet using the relaxation labeling algorithm. Mapping the extracted Hungarian taxonomy to the Hungarian core structure using WN would provide the Hungarian WordNet with glosses, in addition to further synonymy and holonymy links.

4. Conclusion

In this paper we have described several methods we are using for the creation of the Hungarian nominal WordNet. A combination of automatic and manual methods is used. The manual method relies on human experts, who are allowed to work independently, constructing the higher levels of the hierarchy. Automatic methods relying on the bilingual and monolingual dictionaries are used to link a basic set of Hungarian nouns to WordNet. A third group of methods, which depend on taxonomies extracted from the monolingual dictionary, supplements this process. Our approach relies on the assumption that WordNet's semantic structure should provide us with an ample framework supporting the initial phase of our work.

5. References

Agirre, E., X. Arregi, X. Artola, A. Díaz de Ilarazza, and K. Sarasola, 1994. Conceptual Distance and Automatic Spelling Correction. In *Proceedings of the workshop on Computational Linguistics for Speech and Handwriting Recognition.*

Atserias, J., S. Climent, X. Farreres, G. Rigau and H. Rodríguez, 1997. Combining multiple methods for the automatic construction of multilingual WordNets. In: *Proceedings of the International Conference on Recent Advances in Natural Language Processing*, Tzigov Chark.

Copestake, A., T. Briscoe, P. Vossen, A. Ageno, I. Castellon, F. Ribas, G. Rigau, H. Rodriguez, A. Samiotou, 1994. Acquisition of Lexical Translation Relations from MRDs. In *Journal of Machine Translation*, 3.

Copestake, A., 1990. An approach to building the hierarchical element of a lexical knowledge base from a machine readable dictionary. In *Proceedings of the First International Workshop on Inheritance in Natural Language Processing.*

Daudé J., L. Padró and G. Rigau, 1999. Mapping Multilingual Hierarchies using Relaxation Labelling. In *Proceedings of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC'99)*.

Farreres, X., G. Rigau and H. Rodriguez, 1998. Using WordNet for building Wordnets. In*: Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, Montreal.

Juhász, J., I. Szőke, G. O. Nagy, M. Kovalovszky (ed.), 1972. *Magyar Értelmező Kéziszótár*. Budapest: Akadémiai Kiadó.

Kunze, C., A. Wagner, D. Dutoit, L. Catherin, K. Pala, P. Ševeček, K. Vider, L. Paldre, H. Orav, H. Oim. 1998. *First WNs for BCs in French, German, Czech and Estonian*. EuroWordNet Deliverable 2D007.

Miller, G. A., 1990. Nouns in WordNet: a Lexical Inheritance system. In *International Journal of Lexicography* 3 (4), 1990: 245-264.

Nagy, D., 2001. *Computer Aided Methods for Lexical Database Compilation (Hungarian Nominal WordNet)*. Master's Thesis, Budapest University of Technology and Economics.

Országh, L., T. Magay 2001. *Angol-magyar akadémiai nagyszótár*. Budapest: Akadémiai Kiadó.

Pala, K., and P. Ševeček, 1999. *The Czech WordNet*. EuroWordNet (LE-8328) Deliverable 2D014

Prószéky, G., M. Miháltz and D. Nagy, 2001. Toward a Hungarian WordNet. In *Proceedings of the NAACL 2001 Workshop on WordNet and Other Lexical Resources*, 174–176.

Rigau, G., J. Atserias and and E. Agirre, 1997. Combining Unsupervised Lexical Knowledge Methods for Word Sense Disambiguation. In: *Proceedings of the 35th Annual Meeting of the ACL*. Madrid, Spain.

Rigau, G., H. Rodriguez and E. Agirre, 1998. Building Accurate Semantic Taxonomies from Monolingual MRDs. In *Proceedings of COLING-ACL '98*. Montréal, Canada.

Vossen, P. (ed.), 1999. *EuroWordNet General Document.* EuroWordNet (LE2-4003, LE4-8328), Part A, Final Document.

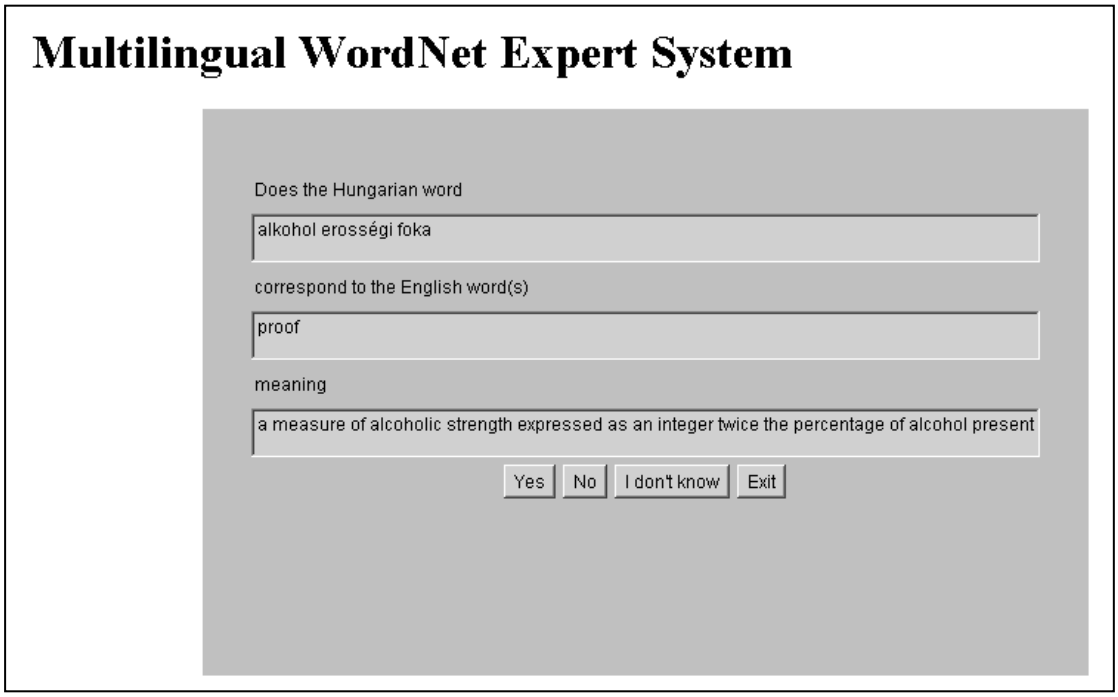


Figure 1. Disambiguation dialogue

```
Synonyms/Hypernyms (Ordered by Frequency) of noun ló
1 sense of ló
Sense 1
Equus caballus, ló                                          (horse)
      => emlős                                              (mammal)
          => állat                                          (animal)
              => valami                                     (entity)

Hyponyms of noun ló
1 sense of ló
Sense 1
Equus caballus, ló                                          (horse)
      => harci mén                                          (steed)
      => amerikai félvad ló, musztáng                       (mustang)
      => versenyló                                          (racehorse)
```

Figure 2. Sample hypernymy/hyponymy hierarchy

Viewing Semantic Networks as Hypermedia

Dimitris Avramidis[*†], **Maria Kyriakopoulou**[*†], **Giorgos Kourousias**[†], **Sofia Stamou**[*†], **Manolis Tzagarakis**[†]

[*]Computer Engineering and Informatics Department
University of Patras
GR-265 00, Rion Patras, Greece
{avramidi, kyriakop, stamou}@ceid.upatras.gr

[†]*Research Academic* Computer Technology Institute
Riga Feraiou 61,
GR-262 21, Patras, Greece
{avramidi, kyriakop, stamou, gk, tzagara}@cti.gr

Abstract

The analogy of a semantic network to hypertext has long been recognized, and a semantic network has been considered as a logical model of hypertext – especially for those hypertexts with typed nodes and links. Moreover, wordnets form the most representative type of semantic networks in the field of Natural Language Processing and semantics in particular. It is obvious that hypertext and wordnets share many common points regarding their fundamental principles and the objectives towards which they both aim. In particular, they are both targeted towards capturing relations that possibly exist between objects and thus providing information of the underlying objects via various types of links used for describing the relations. In this respect, we strongly believe that if semantic networks are viewed beyond strictly linguistically constraints and applications, the results could only be beneficial.

1. Introduction

Hypertext[1] has always been closely related to the idea of freedom to associate, making it to be considered as an alternative means of structuring information. This new promising field provides its users (namely, authors and readers) with effective ways of presenting and exploring information. For authors, hypertext systems offer a high degree of flexibility for connecting pieces of information and presenting it as an assembled collection in an information network. For readers, hypertext provides tools for navigating in these information networks and for exploring them freely. Therefore, hypertext can be a precious dialogic means, facilitating the organization of information according to the user needs.

On the other hand semantic networks form a highly structured linguistic resource enabling a flexible navigation through the lexical items of a language. Wordnet forms a kind of conventional dictionary where semantic information of the terms it contains is represented. The main structural entities of wordnets are language internal relations through which words are linked based on their semantic properties. The main contribution of wordnets in lexicography is the systematic patterns and relations that exist among the meanings that words can be used to express. In this respect wordnets as a particular type of semantic networks resemble much hypermedia as far as the structural organization of information is concerned.

The paper is organized in the following way. Section 2 provides a brief overview of structure in semantic networks. In section 3, we reason about the ability of hypertext to structure information. Section 4 focuses on the similarities that hypertext and wordnets share, claiming that semantic networks can be viewed as hypertext. Finally, section 5 refers to the benefits that these two research areas may have if they are seen as a whole.

[1]Initially, hypertext dealt only with the manipulation of text. Nowadays, one can shape information structures containing pictures, video, sound, etc. Hypermedia – a contraction of the words *Hypertext* and *Multimedia* – is a name invented to stress this change of emphasis.

2. Structure in Semantic Networks

Wordnets form the most representative type of semantic networks in the field of Natural Language Processing and semantics in particular. Motivated by theories of human knowledge organization, wordnet emerged as a highly structured language repository, where words are defined relatively to each other. Unlike machine-readable dictionaries and lexica in book format, wordnet makes the commonly accepted distinction between conceptual-semantic relations, which link concepts and lexical relations, which link words (Evens, 1988). Thus, despite their resemblance to typical thesauri, wordnets in general clearly separate the conceptual and the lexical levels of language, and such a distinction is reflected via semantic-conceptual and lexical relations that hold among synsets and words respectively. Wordnets form semantic dictionaries that are designed as networks, partly because representing words and concepts as an interrelated system seems to be consistent with evidence for the way speakers organize their mental lexicons (Miller, 1998; Kay, 1989).

Wordnets' hierarchical structure allows a searcher to access information stored in lexical chains along more than one path, semantics being among them. Conceptual structures are modelled as a hierarchical network enabling a graphical representation of the lexicalized concepts when the latter are denominated by words (Priss, 1998). The theoretical analysis shows dependencies among semantic rela-

tions, such as inheritance of relations from sub-concepts to super-concepts. Therefore, related senses grouped together under the same lexical chain form preliminary conceptual clusters. Words belonging to the same lexical chain are connected via language internal relations, each one denoting the type of relation that holds among the underlying word meanings. Some of the language relations are bi-directional in the sense that if a link holds between term A and B then a link also holds between term B and term A. However, bi-directionality of the relations strongly depends on the language particularities and semantic properties of the underlying word meanings.

In order to account for particularities in lexicalized concepts, tags are assigned to each lexical relation denoting specialized semantic characteristics of a word's meaning. Tags can be viewed as a means of semantic constraints posed upon semantic relations that link word meanings rather than word forms. Moreover, tags provide information about which of the semantic properties represented in a lexical chain are inherited to its components. In this respect, words represent an atomic and unbiased level of individuality that becomes meaningful via anchoring of semantic relations. As Hasan (Hasan, 1984) pointed out, any word in a chain can be related to multiple other words in that chain. All lexical relations form a graph where cycles are disallowed since after all they contribute very little of any new information.

Summarizing, the structure of lexical data within wordnets is what differentiates the latter from traditional lexicographic aids (both dictionaries and thesauri). The motivation behind construction semantic networks in the form of a graph relies on the fact that lexical data becomes meaningful only via predefined linguistics structures. Navigation through the content of wordnets becomes feasible via language internal relations, which form the main notion around which structure is defined.

3. Hypermedia Principles of Structure

The term of hypertext cannot be explicitly defined since one can approach it by different directions. More specifically, there are those who claim that hypertext can be viewed as an interaction paradigm, referring to the manipulation of "pointing at a link and clicking it" in order to follow it. Additionally, there are others maintaining that "hypertext deals with the organization of information", regarding not only data but also structure as first-class user abstractions. Finally, there is another user group that considers "structure more important than data", making hypertext more structure-based technology than data-dependent.

Adopting the "primacy of structure over data" (Nürnberg et al., 1997), hypertext can be seen as a technology well suited to exploring different kinds of representational structures (Marshall, 1987). Viewing different parts of information as objects, users, often referred to as readers, can navigate through it in a more effective and convenient fashion. Additionally, authors can manipulate information according to their needs (Kyriakopoulou et al., 2001). Therefore, hypertext can be regarded as an informal mechanism, which describes the attributes of these objects and captures relationships that possibly exist between them. Such a characteristic made hypertext become known as an alternative way of structuring information.

Autonomous units of data (e.g. text, images, etc.) can be connected non-linearly creating a structure that has the form of a graph. Apparently, such type of organization and representation of information benefits not only the readers but also the authors, each one by their own point of view. More specifically, readers can retrieve the information they want in the right order serving more easily their particular needs, whereas authors can organize their ideas more efficiently by creating relationships (links) between parts of data (nodes). Thus, hypertext can be a precious dialogic means that offers more flexibility and the freedom of choice to the users according to their preferences, the level of comprehension, and other determined factors.

The analogy of a semantic network to hypertext has long been recognized (Conklin, 1987), and a semantic network has been considered as a logical model of hypertext – especially for those hypertexts with typed nodes and links. As it is widely known, a semantic network is a knowledge representation scheme consisting of a directed graph in which conceptual units are represented as nodes, and relations between the units are represented as links. The graph becomes semantic when each node and link is assigned a particular type, making it meaningful. The essential idea of semantic networks is that the graph-theoretic structure of relations can be used for inference as well as understanding (Lehmann, 1992). In this paper we claim that semantic networks may be profitably viewed as hypertext.

Trying to model different user needs in hypertext, the notion of domain appeared, defining special structural abstractions with specific properties as well as a set of behaviors. The role of structural abstractions is to capture and generalize the knowledge of different problem domains, whereas behaviors are described as computation over structure which is considered as a crucial parameter for the semantic of hypertext structure (Leggett and Schnase, 1994) (see table 1). For example, the idea of taxonomic domain was coined by biologists wanting support for the task of creating taxonomies of the species they were researching (Nürnberg et al., 1996). Similarly, within the last decades, various domains, such as navigational (Halasz, 1987), spatial (Marshall et al., 1994), argumentation (Conklin and Begeman, 1987), etc., have emerged. Since semantic networks and hypertext are closely related, the former ones may be considered as a new domain. The issue in hypertext upon the introduction of a new domain is not to express the domain structure using some general model of structure, but to provide users with domain specific structure to directly work with.

Taking the aforementioned into consideration, it is inferred that the need for domain existence in hypertext is essential. Towards the better exploitation of the properties provided by a particular domain, tools can be developed in order to utilize these specific structures. In this way, users can have the opportunity to work with these tools in order to perform syntactic and/or semantic checks, and maybe to perform structural computations that are only relevant within the domain. Therefore, semantic networks can possibly take advantage of these features improving the infor-

Domains	Structural Abstractions	Behaviors
Navigational	node, link, anchor	follow link, generic links
Taxonomic	taxonomy, taxon, specimen	open taxon, compare, auto generate, detect double categorizations
Spatial	item, space, implicit structure	spatial parse
Argumentation	issue, position, evidence	support link, oppose link, circular argument detection
Wordnet	synset	?

Table 1: Example domains in hypertext.

mation management and graph organization.

4. Approaching Wordnet via Hypermedia

Hypertext and wordnets share many common points regarding their fundamental principles and the objectives towards which they both aim. In particular, they are both targeted towards capturing relations that possibly exist between objects and thus providing information of the underlying objects via various types of links used for describing the relations. Therefore, the main characteristic of wordnets and hypertext systems is the ability to create associations between semantically related information items. On the one hand, these associations imply purposeful and important relationships between associated materials, whereas on the other hand the emphasis upon creating associations stimulates and encourages habits of relational thinking of the user (Landow, 1987).

Relations form the notion around which both semantic networks and hypertext are organized. In the case of semantic networks, relations are denoted explicitly between the lexical units they contain via predefined lexical links, and capture information on the semantic properties of words. In the case of hypertext, although the notion of association can be met in all hypertext domains, the navigational domain with the use of *links* is more closely related to it. Consequently, lexical relations form the fundamental entity of semantic networks the same way as associations in hypertext form the basic structural element around which domains are modeled.

In both cases, information objects (either lexical or not) are heavily structured in order to enable users of wordnets or hypertext navigate through the information they contain successfully. Structure is achieved via internal links, which form the basis on which information is stored and expressed. However, links in semantic networks and hypertext are until recently viewed as two distinct elements and no attempt has been made towards comparing the two. We report on the similarities that exist between hypertext relations and semantic links in an attempt to model the latter in hypertext systems.

In order to support this linking activity in an effective way, hypertext researchers have created a flexible link structure incorporating different levels of functionality. More specifically, in hypertext one can create single or bi-directional links, binary or n-ary links, links to links, automatically activated links, etc. Similarly, links in wordnet are bi-directional and there is generally no restriction on the number and types of links they could be included in it as long as the relatedness between the information items is properly and adequately expressed. Bi-directionality of links indicates that if an object A is somehow related to an object B then object B is again related via the same or another relation to the object A.

However, since bi-directionality might not always be the case in wordnets, special tags need to be attached to the relations to denote their single direction. Namely, tags are being used on semantic network relations to indicate that a lexical item is related to another via a particular type of link but not vice versa. Tags are attached to each link separately and act like constraints on the information provided by the link. However, in the case of hypertext, due to the existence of many specialized domains, the notion of tags is used implicitly.

Furthermore, besides creating associations among semantically related information items, another characteristic shared between hypertext and semantic networks is inheritance. This feature implies that properties of the father are inherited to the children. More specifically, the notion of generalization and specialization forms the principle on which relations are expressed. Specialization and generalization define a containment relationship between a higher-level entity set and one or more lower-level entity sets. Specialization is the result of taking a subset of a higher-level entity set to form a lower-level entity set, whereas generalization is the result of taking the union of two or more disjoint (lower-level) entity sets to produce a higher-level entity set.

Inheritance in wordnets is described via the *H/H tree* that is the complementary hypernymy/hyponymy relations. This type of relationship between objects result in viewing wordnets like tree-structured sources of information, and thus not allowing circular loops. As far as hypertext is concerned, these organizational structures exist in the taxonomic domain under the respective terminology of *supertaxon* and *subtaxon*. The subtaxon is associated with the supertaxon via an "is-a" relationship, inheriting all the characteristics that the latter might have. In particular, the user can classify objects (known as specimens) into sets according to their features, search within the members of a set to find relationships or discreet subsets, and create new sets from the already existing ones.

Finally, what should be stressed is that semantic networks and hypertext, despite the characteristics they have in common, they also have quite a few differentiations, mainly stemming from their applications and usage. What we at-

tempted in this paper is to explore the usefulness of both wordnets and hypertext systems beyond the limitations imposed by the applications at which they are targeted. What we claim is that by treating wordnet, as a new domain of hypertext would result in a better understanding of the language structure and consequently human memory and way of thinking. After all, any application is targeted towards human beings and aims at providing a clear description of how information is stored and thus how it should be interpreted. In this respect we strongly believe that if semantic networks are viewed beyond strictly linguistically constraints and applications, the results could only be beneficial.

5. Discussion

As it has been already mentioned, the technology of hypertext is not mainly used for the organization of information but can be considered as a significant means of structuring information. Viewing semantic networks as hypermedia, the power of hypertext is enforced even more, making us infer that any kind of information can be structured under the fundamental characteristics of hypertext. Furthermore, some special structural characteristics of semantic networks can be effectively exploited by hypertext community, resulting in the extension of already existing domains, such as taxonomic, navigational, etc. More specifically, tags might be such a characteristic, providing the hypertext users with the ability to pose semantic constraints upon relations, enabling the distinction among different types of whichever kind links.

On the other hand, taking advantage of the structural characteristics of hypertext while developing semantic networks can prove quite beneficial for both the lexicographic and linguistic communities. In particular, hypertext provides ways of organizing information stored in such systems in a meaningful way so that navigation through the stored data is facilitated. By adopting structures implied by the hypertext community in other applications such as lexicography, the potential and performance of the latter can be greatly improved. When it comes to the storage of lexicographic data the need for efficient structures becomes apparent due to the large amount of information that has to be handled and especially due to the dynamic nature of the underlying information. Moreover, even if behaviors exist in wordnets, they haven't been explicitly defined so far, resulting in less comprehensive usage of the underlying data.

Language forms the mean through which communication is achieved and as such its processing undergoes through various structural decisions that need to be taken prior to storing and incorporating lexicographic data in applications. In this paper we attempted a preliminary comparison among structural characteristics of semantic networks with hypertext and as a conclusion we claim that the abovementioned areas share a few common points in terms of data representation, storage and navigation. What we imply is that semantic networks and hypertext are by no means equivalent in terms of structure. Conversely, what we suggest is that by tracing points between the two and by adopting structural characteristics of other domains can only be beneficial for both sides.

6. References

Jeff Conklin and Michael L. Begeman. 1987. gIBIS: A Hypertext Tool for Team Design Deliberation. In *Proceedings of the ACM Conference on Hypertext*, pages 247–251, Chapel Hill, North Carolina, United States. ACM Press.

Jeff Conklin. 1987. Hypertext: An Introduction and Survey. *IEEE Computer*, 20(9):17–41.

Martha W. Evens, editor. 1988. *Relational Models of the Lexicon: Representing Knowledge in Semantic Networks*. Cambridge University Press, Cambridge, England.

Frank G. Halasz. 1987. Reflections on Notecards: Seven Issues for the Next Generation of Hypermedia Systems. In *Proceedings of the ACM Conference on Hypertext*, pages 345–365, Chapel Hill, North Carolina, United States. ACM Press.

Ruqaiya Hasan. 1984. Coherence and Cohesive Harmony. In James Flood, editor, *Understanding Reading Comprehension*, pages 181–219. IRA.

Martin Kay. 1989. The Concrete Lexicon and the Abstract Dictionary. In *Proceedings of the 5th Annual Conference of the UW Center for the New Oxford English Dictionary*, pages 35–41, Waterloo, Ontario, Canada.

Maria Kyriakopoulou, Dimitris Avramidis, Michalis Vaitis, Manolis Tzagarakis, and Dimitris Christodoulakis. 2001. Broadening Structural Computing Systems Towards Hypermedia Development. In *Proceedings of the 3rd International Workshop on Structural Computing*, pages 131–140, Århus, Denmark. Springer-Verlag.

George P. Landow. 1987. Relationally Encoded Links and the Rhetoric of Hypertext. In *Proceedings of the ACM Conference on Hypertext*, pages 331–343. ACM Press.

John J. Leggett and John L. Schnase. 1994. Viewing Dexter with Open Eyes. *Communications of the ACM*, 37(2):76–86.

Fritz W. Lehmann. 1992. Semantic Networks in Artificial Intelligence. In Fritz W. Lehmann, editor, *Semantic Networks*, pages 1–50. Pergamon Press Ltd.

Catherine C. Marshall, Frank M. Shipman, and James H. Coombs. 1994. VIKI: Spatial Hypertext Supporting Emergent Structure. In *Proceedings of the 1994 ACM European Conference on Hypermedia Technology*, pages 13–23, Edinburgh, Scotland. ACM Press.

Catherine C. Marshall. 1987. Exploring Representation Problems Using Hypertext. In *Proceedings of the ACM Conference on Hypertext*, pages 253–268, Chapel Hill, North Carolina, United States. ACM Press.

George A. Miller. 1998. Nouns in Wordnet. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 23–46. MIT Press.

Peter J. Nürnberg, John J. Leggett, Erich R. Schneider, and John L. Schnase. 1996. Hypermedia Operating Systems: A New Paradigm for Computing. In *Proceedings of the the 7th ACM Conference on Hypertext*, pages 194–202, Bethesda, Maryland, United States. ACM Press.

Peter J. Nürnberg, John J. Leggett, and Erich R. Schneider. 1997. As We Should Have Thought. In *Proceedings of*

the 8th ACM Conference on Hypertext, pages 96–101, Southampton, United Kingdom. ACM Press.

Uta Priss. 1998. The Formalization of Wordnet by Methods of Relational Concept Analysis. In Christiane Fellbaum, editor, *WordNet: An Electronic Lexical Database*, pages 179–196. MIT Press.

Requirements for Domain-Specific WordNets

Koutsoubos Ioannis-Dimitrios, Christodoulakis Dimitris

Computer Engineering and Informatics Department,
University of Patras
Research Academic Computer Technology Institute
koutsoub@ceid.upatras.gr
dxri@cti.gr

Abstract

This paper addresses the need for domain-specific resources in NLP applications. The motivation for this work emerged from the current limitations of WordNet when the latter is adopted in a domain-specific applications and environments. Moreover, we report on methods and techniques for extending and tuning WordNets for domain-specific usage. We envisage a unifying WordNet structure, that will be easily extendable and customizable and also has the ability of incorporating other lexical and semantic resources with minimum effort. Finally, we discuss on the advantages of a unified WordNets structure over various types of applications that require extensive usage of NLP applications.

1. Introduction

Lexical resources used in natural language processing have evolved from handcrafted lexical entries to machine-readable lexical databases and large corpora. Much effort is being applied no the creation of electronic lexicons and electronic linguistic resources in general. However, the above resources are expensive to build, and instead of creating new ones from scratch, it is preferable to adjust and extend existing ones.

One linguistic resource of great interest is WordNet (FellBaum, 1998). WordNet is a general-purpose concept ontology, which has been developed a Princeton University, and resembles the way that humans store and organize information in their memory. It can be used both as an on-line dictionary or thesaurus for reference purposes, and as a taxonomic lexical database. WordNet is a resource of high quality and is freely available over the Internet, thus it has rapidly become one of the most widely used tools in language engineering, research and development.

However, as many technical words or word meanings cannot be found in general semantic databases such as WordNet, Natural Language Processing (NLP) in specific domains requires specialized semantic lexica. A major difficulty in using WordNet or any other general NLP resource in a specific domain is that much of the specialized semantic attributes (terminology, semantic relations, domain-specific relations etc) of the domain is not present. In this paper we describe the requirements of domain-specific wordnet development. First, we describe shortly the application usage of wordnet. We then explain the need for domain-specific NLP resources. Next we present techniques and methodologies that are used till now for the development of domain-specific wordnets. Finally, we present our approach towards developing domain-specific wordnets. Finally, we outline some early conclusions regarding the necessity for building domain specific WordNets and their usefulness in various applications.

2. WordNet applications

WordNet has been identified as an important resource in the human language technology and knowledge processing communities. Its applicability has been cited in many papers and systems have been implemented using WordNet. Almost every NLP application nowadays requires a certain level of semantic analysis. The most important part of this process is semantic tagging: the annotation of each content word with a semantic category. WordNet gives a solution to the above problem and has been used in various applications including Information Retrieval, Word Sense Disambiguation, Machine Translation, Conceptual Indexing, Text and Document Classification and many others.

3. Need for domain-specific resources

A problematic issue is that general semantic resources like WordNet do not cover many terms and concepts specific to certain domains, and also include many unnecessary (general) concepts and relations. Therefore these resources need to be tuned to a specific domain at hand. This involves selecting those senses that are most appropriate for the domain, as well as extending the sense inventory with novel terms and novel senses that are specific to the domain (Buitealar, 2001; Turcato et al.,2000). Another problem is that in a specific domain only a subset of the semantic relations defined in the general semantic resource hold. Also many technical words or word meanings cannot be found in general resources. Partial overlaps can be found, but the domain specific description is likely to be more precisely defined and reliable.

As a result of these difficulties with existing generic resources, NLP system builders have tended to handcraft resources for each application domain, or have looked at techniques for automatically or semi-automatically constructing lexicons of various sorts from texts in the domain.

The main problem is how can we develop domain-specific resources either from scratch or by using existing resources with minimum effort.

There are two main problems. The extension/expansion of existing general resources and the

adaptation of these resources to a specific domain and how can we acquire the above with minimum effort. In particular, the first problem regarding extending already existing lexicographic resources with domain-specific terminology requires a lot of manual work since additional information needs to be attached to the contents of the resources emerging from the underlying domain of interest. This would imply that large corpora from various terminological domains should be used in order to perform a semantic annotation of the terms they comprise of. In the second case, adapting existing resources toi particular applications would require not only enriching those resources with specialized terminology but it would also need partial restructuring of the resource so that the new content is sufficiently represented in a meaningful way.

In the case of wordnets the solution that implies the development of domain specific semantic networks seems as the best way of solving many problems imposed by the lack of such resources from various NLP applications. In the following sections we briefly report on the work conducted so far in this area and we continue with a description of our approach towards the necessity of domain specific terminological resources.

4. Building Domain-Aware WordNets so far

It is obvious from the above how important is the need for domain-specific NLP resources in general, and particularly for domain-specific wordnets. Several methods for the creation of domain-specific resources have been applied ranging from:

- Creation from scratch, to
- Data Extension of generic WordNets for a specific domain and
- Structure Extension of generic WordNets for a specific domain.

More specifically, the methodology adopted for each of the abovementioned techniques is described as follows:

4.1. Creation from Scratch

One solution, and apparently the most costly, is to handcraft domain-specific wordnets from scratch for any specific-domain. Building wordnets by hand requires significant amount of time and effort even for restricted domains. Furthermore this effort is repeated when a system is ported to another domain. The above leads us to automatic or semi-automatic approaches for building wordnets and other NLP resources using already available existing generic resources.

4.2. Data Extension of generic WordNets for a specific domain

The adaptation of existing resources to a specific domains includes selecting those terms and meanings that are relevant for the domain, adding new terms and meanings that are missing from the existing resource, removing relations that are irrelevant or incorrect in the specific domain, keeping relevant relations and adding missing ones (Buitelaar & Sacaleanu, 2001,2002;Turcato et al.,2000).

4.3. Structure Extension of generic WordNets for a specific domain

Another solution to the problem is to extend existing generic wordnet structure incorporating in it semantic distinctions from external resources such as ontologies, semantic taxonomies, domain-specific corpora etc. One approach is to add an ontology layer, which refers to specific domain attributes and characteristics and thus relates the domain with the linked concepts (Vossen, 1998; O'Sullivan et al., 1995). Another way is to link concepts with relevant document collections or corpora and find a way to compute the weights of their topic signatures (Agirre et al., 2001). Automatically build an hierarchy of terms using terms extracted from documents of a specific domain, combine it with existing hierarchies in wordnet and by fusing and clustering we can derive a condensed tree that has maximum coverage due to the extension, but only contains distinctions and classifications that are relevant and desired (Vossen, 2001).

There have also been attempts to integrate the information of generic lexical databases with existing ones (Magnini & Speranza, 2001).

5. What is missing from WordNet?

The success of WordNet has determined the emergence of several projects that aim the construction of WordNets for other languages than English or to develop multilingual or specialized WordNets or to extend existing WordNets for specific domains or to incorporate WordNet in various NLP applications. Through these attempts many WordNet's advantages have been discovered and some weaknesses have appeared. According to *(Harabagiu et Al. 1999)* the main weaknesses of WordNet cited in the literature are:

1. The lack of connections between noun and verb hierarchies.
2. Limited number of connections between topically related words.
3. The lack of morphological relations.
4. The absence of thematic relations/ selectional restrictions.
5. Some concepts (word senses) are missing.
6. Since glosses were written manually, sometimes there is a lack of uniformity and consistency in the definitions.

Until now there has been a lot of research for methods and techniques for WordNet development, customization, multilinguality, alignment with existing resources etc. However all the attempts concentrated on everything that was related to the content of WordNet and WordNet's lexical and semantic coverage, leaving behind everything that is related to the data model of Wordnet and WordNet's structure (the way that WordNet's data are stored and manipulated).

From a WordNet's developer perspective the main disadvantage of WordNet is that WordNet is almost a black box. The WordNet community is increasing year by year, but till now there are no standards about WordNet structure. With a standard WordNet structure and all the methods and techniques that are already available for WordNet construction, extension, alignment with other NLP resources and link with other language WordNets will road the map for a new perspective towards wordnets and their usage in every day NLP applications.

6. Requirements for Domain-Specific WordNets

In this section we describe the requirements that a domain-specific WordNet must satisfy. Many of these requirements are also addressed to generic WordNets.

One key point is the integration of domain-specific wordnets with generic ones. On the one hand the domain-specific wordnet is a specialized resource, whose content is supposed to be more accurate and precise for the domain that it was designed; on the other hand, the generic wordnet guarantees a more uniform coverage as far as high level senses are concerned. There must be a flexible and modular integration procedure, which will give the ability many domain-specific wordnets to co-exist with one generic one. This procedure shall manage inconsistencies and overlaps between the different resources. Co-existence of lexical resources that are targeted towards various domains has many advantages.

First and foremost, it enables the comparison of concepts used in both genetic and domain specific vocabulary. ,It can also contribute towards the ease identification of the domain in which a concepts belongs to. However, the most important feature of such resources is the potential of using a domain specific semantic resource for various types of applications. The latter results in a global lexicographic resource of great usefulness in many tasks and applications.

A problematic issue in the field of NLP is that it does not often suffice to depend on any single resource , either because it does not contain all required information or the information is not organized in a way suitable for the purpose. So merging of different resources is necessary. Many different NLP resources are available to the NLP community e.g. corpora, morphological lexicons, semantic lexicons, ontologies. Many applications will benefit from the integration of such resources with WordNet (Kwong, 1998). So there shall be a flexible structure that will provide fully-automatic or semi-automatic mechanisms for the incorporation of such resources in WordNet.

WordNet has been criticized for its lack of relations between topically related concepts. The enrichment of WordNet's concepts with topic signatures and the application of topic relations open the avenue for interesting ontology enhancemenrs, as they provide concepts with rich topical information (Agirre et al., 2001). For instance, similarity between topic signatures could be used to cluster topically related word meanings. Word sense disambiguation methods could profit from these richer ontologies, and improve word sense disambiguation performance.

WordNet's concepts shall be enriched with additional semantic and non-semantic attributes. Some of these attributes may be word usage examples, words that accompany a concept in a specific meaning, morphology information, domain-specific information about the concept. For example if we meet the word 'world' with the meaning of 'people' we cannot find this word in plural. The above attributes may also be links to corpora or other incorporated resources. By the same way attributes shall be applied to relations, too. For instance some relations may exist under certain constraints in a domain-specific context, and there must be a way of identifying domain-specific relations that do not exist in generic or in other domain contexts, or generic relations that are also applied in domain-specific context. One such examples concerns the application of wordnet during language teaching tasks in which phonetic information could be added.

The WordNet structure shall be organized in a way that will allow the insertion of additional relations between concepts, additional attributes both for concepts and relations and constraints both for attributes and relations without affecting existing data and with a way that will be as easy and effective as possible.

Another feature that shall be made available to WordNet is the definition of the behavior of relations regarding the domain that the wordnet is designed for and the application usage of the WordNet. Following this approach different applications in a specific domain have the ability to share common data. This means that if somebody developed a domain-specific WordNet for domain A in order to use it in his document classification application and another one plans to develop a query expansion system for an information retrieval application he can use the already developed WordNet in the same way only by changing for instance the behavior of the synonymy relation which will now be used for searching in documents with the synonyms of a given word. In other applications for instance the hyperonym relation may be used for getting more general word meanings than the given one and in other applications may define an upper-level category of classification of documents.

An Ontology Layer should be present on the upper level of the semantic features of a language for the transfer of domain specific semantic characteristics and distinctions relative to the domain to the underlying concepts. However, it might be more effective if the concepts belonging to the upper level had as additional features the abovementioned distinctions and thus all terms related to these inherit these distinctions and features. . The above resembles much the wordnet-type of information storage and representation and would result in a more flexible semi-automatic extraction and development of domain-specific ontologies based on wordnet information.

All the above leads us to the conclusion that there is an imperative need for a flexible and unifying WordNet structure. The whole WordNet community shall concentrate in the standardization process of WordNet structure. The structure must be able of defining concepts, relations, attributes for both of them, flexible linking with existing NLP resources and components. It also must be easily customizable and extendable, allow the co-existence of generic and specialized wordnets providing mechanisms for domain resolution and identification. Such an approach will make easier the process of multilingual wordnet linking and will also provide an unifying approach to any NLP problem that wordnet is called to solve. Since the research concerning wordnet itself and its applications has grown extremely in the past years a standard structure will just provide wordnet an easy and effective way in everything concerning wordnet from wordnet development to wordnet usage in NLP applications.

With the need of the standardization of structure comes the need for a wordnet protocol, which will describe all the operations, methods, functions that wordnet offers. The existence of a wordnet protocol means that everyone is free to develop wordnet in the way

they prefer even if it is relational databases, xml files, polaris format files, indexed text files etc, as long as they follow the pre-specified protocol.

The need for a unified structure is requested to solve problems related to wordnet extension as long as other problems emerging from wordnet applications ands need to be solved via a unified and common way. The main idea behind this assumption is the conversion of wordnets into a linguistic resource that would apply to as much as possible to ll members of the NLP community.

The need for a common protocol needs to be solved through a unification of the applications of the already existing wordnets. A common protocol applications envisaged for one monolingual wordnet (e.g. the English Wordnets) could be used in other monolingual wordnets without any previous change required in the structure or content of the latter. Of course this implies that in case one application performs sufficiently for a particular domain then its usage in another domain needs solely the existence of a wordnet for another domain and no extra effort towards structural or content modifocations.

7. Conclusion

We identified the need for domain-specific WordNets and presented some requirements that shall be met both by generic and specialized WordNets. WordNets success in the field of NLP can be even greater but to achieve this there must be standardization concerning both the structure and the protocol, which will be used by applications that use WordNet. This is a long way, and it must be walked with the right steps.

8. References

Agirre E., Ansa O., Martinez D., Hovy E.(2001). Enriching WordNet concepts with topic signatures. In *Proceedings of the NAACL worshop on WordNet and Other Lexical Resources: Applications, Extensions and Customizations*. Pittsburg, USA

Buitelaar P., Sacaleanu B.(2002). Extending Synsets with Medical Terms In : *Proceedings of the First International WordNet Conference*, Mysore, India.

Buitelaar P., Sacaleanu B. (2001). Ranking and Selecting Synsets by Domain Relevance In *Proceedings of WordNet and Other Lexical Resources: Applications, Extensions and Customizations, NAACL 2001 Workshop*, Carnegie Mellon University, Pittsburgh.

Farreres, G. Rigau, and H. Rodriguez (1998). Using WordNet for Building WordNets. In *Proceedings of COLING-ACL Workshop on Usage of WordNet in Natural Language Processing Systems*, Montr'eal, Canada.

FellBaum Christiane (1998). WordNet: An Electronic Lexical Database. MIT Press Books.

Habert B., Nazarenko A., Zweigenbaum P., and Bouaud J.. (1998). Extending an Existing Specialized Semantic Lexicon. In *Proceedings of first International Conference on Language Resources and Evaluation*, pages 663--668, Granada.

Harabagiu S.M.,Miller A. G. and Moldovan (1999). WordNet 2 – a Morphologically and Semantically Enhanced Resource. In *Proceedings of SIGLEX-99* (pp. 1--8). University of Maryland.

Kwong, Oi Yee (1998). "Aligning WordNet with Additional Lexical Resources". In *Proceedings of the COLING/ACL Workshop on Usage of WordNet in Natural Language Processing Systems*. Montreal, Canada, August.

Magnini, Bernardo and Manuela Speranza (2001). Integrating Generic and Specialized Wordnets. In *Proceedings of the Conference on Recent Advances in Natural Language Processing, RANLP 2001*, Tzigov Chark, Bulgaria.

O'Sullivan D., A. McElligott, R. Sutcliffe (1995). Augmenting the Princeton WordNet with a Domain Specific Ontology, in *Proc. Workshop on Basic Ontological Issues in Knowledge Sharing, International Joint Conference on Artificial Intelligence (IJCAI-95)*. Montreal, Canada.

Turcato D., Popowich F., Toole J., Fass D., Nicholson D. and Tisher G. (2000). Adapting a synonym database to specific domains. In *Proceedings of the ACL'2000 Workshop on Recent Advances in Natural Language Processing and Information Retrieval*. Hong Kong.

Vossen P. (2001) Extending, Trimming and Fusing WordNet for Technical Documents. In: *Proceedings of WordNet and Other Lexical Resources: Applications, Extensions and Customizations, NAACL 2001 Workshop*, Carnegie Mellon University, Pittsburgh.

Vossen, P (ed.) (1998) EuroWordNet General Document. EuroWordNet (LE2-4003, LE4-8328), Part A, Final Document

Notes about Labelling Semantic Relations in Estonian WordNet

Kadri Vider

University of Tartu,
Department of General Linguistics
Tiigi 78-204, 50410 Tartu, Estonia
kvider@psych.ut.ee

Abstract

Estonian language is rich in derivation. Most of derivational suffixes have their regular meaning(s) and is very obvious, that source and target words in derivation have regular lexical-semantic relations between them. The problem of what regular derivational suffixes in Estonian lexica cover what kind of semantic relations in Estonian WordNet is discussed in this paper.
Another problem of labelling connected with semantic relations is related to proper nouns. In purpose to use referential character of proper nouns in word sense disambiguation, we need to connect proper nouns with objects carrying the names e.g. 'John ISA man', but not 'John ISA first name'.

1. Introduction

Compilation of Estonian WordNet (EstWN) started in 1997 and the work is still in progress. The work was funded partly by the Estonian Science Foundation and partly in the framework of the Estonian National Programme of Language Technology. Like other wordnets, EstWN is a lexical-semantic database, the basic unit of which is concept. Concepts are represented as synonym sets (synsets) that are linked to each other by semantic relations. In 1998-1999 EstWN was created as a part of EuroWordNet (EWN) and since then we have used semantic relations from EWN, which are more flexible and richer than in the original (Princeton) WordNet. Still, our experience has shown that there are at least some language-specific semantic relations needed. Up to now, the usage of semantic relations was limited by the set provided by Polaris, the EWN editing tool.

Which new words or concepts should be concentrated on to upgrade the EstWN? It is essential that words actually used in text will be added. Results of word sense disambiguation (WSD) task of corpus texts turned out to be a good way of adding missing and new synsets and senses into our wordnet. (Kahusk and Vider, 2002)

Estonian is usually considered to be an agglutinative language, thus belonging to same group as Finnish, Hungarian and Turkish. It is flective language with free word order.

In order to reach the lemma in the text, Estonian needs morphological analysis. The program ESTMORF, in use at present, renders it possible to analyse the productive derivatives and tag suffixes.

2. Semantic relations in EstWN

The existing Estonian WordNet contains nouns, verbs, some adjectives and proper nouns, more than 10,000 synsets all together. The more detailed description of EstWN is given in the final document of EuroWordNet, Estonian part (Vider et al., 1999)

Semantic relation	**No. of links**
has_hyperonym/has_hyponym	19002
belongs_to_class/has_instance	948
near_synonym	354
xpos_near_synonym	246
has_holonym/has_meronym	234
antonym	209
be_in_state/state_of	186
near_antonym	138
involved/role	134
causes/is_caused_by	128
has_subevent/is_subevent_of	36
has_xpos_hyperonym/has_xpos_hyponym	12
xpos_near_antonym	4
xpos_fuzzynym	2

Table 1: Semantic relations expressed in EstWN in the order of frequency.

3. System of Estonian derivation

Wordnet is based on word meaning and from this point of view such lexical feature as derivation should not play a significant role. But a lot of Estonian derivational suffixes have concrete meanings and this fact can be applied in connecting the derivational base and the derivation with a definite semantic relation, dependent on the meaning of the derivational affix.

In Estonian, derivation is mainly a process of appending derivational suffixes, more than 60 altogether, to both declinable and conjugable words. Suffixes can be appended sequentially; up to four suffixes in a row can be appended in some cases. About 8% of the word forms in a running Estonian text are derived words; in journalism and scientific texts the figure is even higher (Kaalep, 1997).

Derivation, a frequent and productive way in Estonian for forming new words, is a process where adding an affix produces a new lexical item having its own inflectional

paradigm. Derivational morphology in Estonian is always connected with changing the meaning of lexeme. The lexical meaning of the derived word is different from the word used as the derivational base, in some productive cases the derived words belong to a different part of speech

Thus it may be concluded that affixes in Estonian belong to the category of semantics, not grammar. Morphologically derivation can be defined as the formation of a new stem by adding an affix to the last morpheme of the stem.

In Estonian, compounding is even more frequently used for word formation than derivation. Compound words comprise more than 12% of the running words in an average Estonian text. The formation of Estonian compounds is quite free and derived words may also constitute a compound. In this paper we consider only such kind of compounds.

We proceed from the assumption that in a lexicon compiled on the semantic basis the semantic association between the words derived from the same stem should be fixed. It should be possible to automate the relation on the basis of meaningful affixes. The question is which relation should be attributed to which affix.

Derived/derived_from/has_derived relations exist in EWN structure (Vossen 1999), but they are clearly too general and ambiguous for such a language abundant in regular and ample derivation as Estonian.

4. Suffixes actual in EstWN data

This chapter deals, first and foremost with the productive derivation types (formation patterns) that have an independent meaning, e.g.

VERB+**mine** – PROCESS[NOUN],

VERB+**ja** – ACTOR,

PLACE[NOUN]+**lane** – INHABITANT

Lexicalised derivation also has quite a clear relation with the derivational base. Only the idiomised derivations have lost the distinct relation with the derivational base (Kasik 1996).

4.1. Verb suffixes

4.1.1. Verb -> Verb derivation

Most frequent verb suffix in Estonian is **-ta**, which has a causative meaning in verb-to-verb derivation, e.g. kulu/ta/ma (spend, expend) *causes* kuluma (go, be spent); levi/ta/ma (distribute, cause to spread) *causes* levima (spread, be disseminated); liigu/ta/ma (cause to move) *causes* liikuma (move); kao/ta/ma (lose, fail to keep) *causes* kaduma (disappear, vanish, get lost); meenu/ta/ma (remember, retrieve, recall, remind) *causes* meenuma (be reminded); kuiva/ta/ma (dry, make dry) *causes* kuivama (become dry); sünni/ta/ma (birth, give birth) *causes* sündima (be born); nõrges/ta/ma (weaken, make weak) *causes* nõrge/ne/ma (weaken, get weak); aren/da/ma (develop, evolve) *causes* are/ne/ma (evolve, undergo an evolution); puru/sta/ma (break, cause to break) *causes* puru/ne/ma (break, separate, be smashed); rahu/sta/ma (calm, make calm) *causes* rahu/ne/ma (calm, be pacified, become stable); unu/sta/ma (forget, fail to remember) *causes* unu/ne/ma (pass out of mind, be forgotten).

Productive verb suffix **–u** constructs intransitive verbs with reflexive meaning, eg. aeglus/ta/ma (retard) - aeglust/u/ma (slow, become retarded); asen/da/ma (substitute, replace) - asend/u/ma (be replaced); eral/da/ma (separate, divide) - erald/u/ma (separate from); eru/ta/ma (stimulate, shake, excite) - erut/u/ma (become excited about); eten/da/ma (perform, give a performance) - etend/u/ma (play, be performed); kahjus/ta/ma (damage, do harm) - kahjust/u/ma (be damaged); katma (cover) – katt/u/ma (be covered); kuhjama (heap, pile, stack) - kuhj/u/ma (be heaped, be piled); moodus/ta/ma (form, constitute) – moodust/u/ma (be formed, be constituted).

The most important derivation in this group is muutma (*change, alter, make different*) - muut/u/ma (*undergo a change, become different*). The source verb of derivation needs an active agent, but it does not render passive or *is_caused_by* meaning to the verbs with reflexive u-suffix. Lexical expression of passivity is not characteristic of the Estonian language. As to Estonian (perhaps French and German as well) reflexivity is one of the missing semantic relations in the EWN verb structure.

4.1.2. Noun -> Verb derivation

The most common semantic categories in derivations of this type are CAUSE, CHANGE, USE, ADD. Verb arguments behave in this case as derivatives, e.g. RESULT, ACTOR, INSTRUMENT. They all hold subtypes of *involved/role* relation. Often such arguments can be met in synonymous phrases or idioms, e.g. kirju/ta/ma, kirja panema (*write, write down, directly "put into letter"*).

(1) Productive suffix **-ta** and its variant **-sta** have factitive meaning, i.e. one of the arguments of the derived verb is the source of derivation as well. The semantic relation between the verb and its derivational base belongs, in this case, to the subtype of *involved/role* relation, e.g. huvi/ta/ma (interest, cause to be interested) *involved* huvi (interest); avar/da/ma (enlarge, expand, extend) *involved* avar (spacious); elav/da/ma (enliven, liven) *involved* elav (living, alive); nalja/ta/ma (joke, jest) *involved* nali (wit, humour, joke, jest); ahel/da/ma (chain) *involved* ahel (chains, chains); halven/da/ma (make worse, worsen) *involved* halb (bad)

(2) Suffixes **–u** and **–ne** have translative meaning. They present autonomic CHANGE (of state or situation); e.g. korts (wrinkle, fold, crease) – korts/u/ma (wrinkle, ruckle, crease, crinkle, scrunch) kõva (hard, firm, solid, stiff) – kõvast/u/ma (harden, indurate, solidify); kõver (crooked, bent, curved) - kõverd/u/ma (curve, crook, bend); külm (cold) – külm/u/ma (freeze, change to ice); lahus (solution) – lahust/u/ma (dissolv, resolve); niiske (damp, moist) – niisk/u/ma (moisten, dampen); puit (wood) – puit/u/ma (turn into wood, lignify); raev (rage, fury)- raev/u/ma (become furious, see red); rasv (fat, lardy) - rasv/u/ma (fatten, batten, grow fat); rohi (grass) - roht/u/ma (overgrow with grass); suund (direction) – suund/u/ma (head, travel in a direction); kitsas (narrow) – kitse/ne/ma (narrow, contract); halb (bad) – halve/ne/ma (worsen, decline); harv (sparse, thin) – harve/ne/ma (thin out); kauge (far) – kauge/ne/ma (recede, move away)

Existential verbs, where derivation changes only the part of speech should be brought out as a separate group.

4.1.3. Modifying derivation

Derivations formed with the help of affixes modifying the verb have a hyperonym/hyponym relation with the derivational base, for the affixes mentioned above only modify the way of action. The best label for describing such a relationship is troponymy.

Frequentatives (expressing repetition of an action, e.g. hüppama (jump)- hüp/le/ma (hop, skip, jump lightly); mulks (gurgle) - mulks/u/ma (bubble up); tukse (throbbing) - tuks/u/ma (pulsate, throb, pulse); momentanes (express the singleness or suddenness of an action, e.g. tuks/u/ma (pulsate, throb, pulse) - tuks/ata/ma (give a throb)) and continuatives (show the continuity and permanence of an action, e.g. mängima (play) - mängi/tse/ma (dally, trifle, play)) can be differentiated by the affixes.

4.2. Noun suffixes

In case of argument-nominalization the derivative is expressed in the function of one argument of the derivational verb. The more widely-spread arguments include ACTOR, RESULT, INSTRUMENT, OBJECT, PLACE.

4.2.1. Action derivatives

The suffix of absolute productivity **-mine** changes only the part of speech of the derivational base. With the help of this suffix every verb can be changed into a noun, which has cross-part of speech synonym relations, e.g. alustama (begin, start, commence) *xpos_near_synonym* alusta/mine (beginning, start, commencement); harjutama (drill, exercise, practice) *xpos_near_synonym* harjuta/mine (practice session, exercise).

Abstract and metaphorical meanings of the verb should not be bound to the suffix **-mine** but only the ones expressing a definite action.

Due to absolute productivity we have included only such mine-derivatives in the EstWN that were founded in corpus texts.

4.2.2. Personal derivatives

Actor's suffix **–ja** is also a very productive suffix, the application of which is universal for all kind of action, e.g. ehitama (build, construct, make) *involved_agent* ehita/ja (builder, constructor); esindama (represent, be a delegate for) *involved_agent* esinda/ja (representative); juhatama (head, lead) *involved_agent* juhata/ja (leader); kasvatama *involved_agent* kasvata/ja, koristama *involved_agent* korista/ja, kütma *involved_agent* küt/ja, laulma *involved_agent* laul/ja. Some of the ja-derivatives can besides the live agent also express appliances, e.g. ajamõõt/ja (timekeeper); voolumõõt/ja (ammeter); raadiosaat/ja (radio transmitter).

The most productive affix in forming generic names from proper names is **-lane**. The biggest group of lane-derivatives refers to persons by their origin, e.g eest/lane (Estonian); ameerik/lane (American); hiin/lane (Chinese); indiaan/lane (American Indian).

Terms of biological taxonomy form another big group, which could be formed with the help of suffixes **-lane** e.g. kass (cat) - kas/lane (feline, felid); koer (dog) - koer/lane (canine, canid); and **-line**, e.g. kabja/line (perissodactyl mammal); kiletiiva/line (hymenopterous insect); kõrre/line (graminaceous plant).

A productive affix in forming business titles is **-ur**, e.g. kaevama (dig) - kaev/ur (digger, miner); kala (fish) - kal/ur (fisher, fisherman); juus (hair) - juuks/ur (hairdresser); valvama (protect) - valv/ur (defender, guardian, protector).

Feminine suffixes **-nna, -tar** are also productive, e.g. luuleta/ja (poet) - luuleta/ja/nna (poetess); sõber (friend) - sõbra/nna, sõbra/tar (girlfriend). Estonian morphology lacks feminine markers, feminine suffixes exist only in noun derivation. The problem is not new, as in his first papers about EWN-1 Vossen declared that the semantic category WOMAN got lost in converting the Vlis (Dutch) database relations into EWN ones.

4.2.3. Place and set derivatives

All **-la** derivatives refer to a place and indicate a specific place (building, room), e.g. haige (sick person, sufferer, patient) *involved_location* haig/la (hospital); levima (spread, be disseminated) *involved_location* levi/la, parkima (park) *involved_location* park/la (parking lot, car park); suvitama (summer) *involved_location* suvi/la (summer house); sööma (eat, take in) *involved_location* söök/la (lunchroom, eating house).

-kond is a productive suffix expressing collectivism, e.g elanik (inhabitant) *has_holo_member* elanik/kond (population); inimene (human, man) *has_holo_member* inim/kond (humankind, mankind); võistleja (contestant) *has_holo_member* võist/kond (team, squad).

Apart from the kond-suffix, suffix **-stik** refers to the group or set of things or fenomena, e.g. kõrge (high) *has_holo_member* kõrgu/stik (highland, upland); leht (leaf) *has_holo_member* lehe/stik (leafage); mägi (mountain, hill) *has_holo_member* mäe/stik (mountain range); nimi (name) *has_holo_member* nime/stik (list, listing); rahvas (people) *has_holo_member* rahva/stik (population); seade (mechanism) *has_holo_member* seadme/stik (machinery, equipment); taim (plant, plant life) *has_holo_member* taime/stik (vegetation, flora).

4.2.4. Property derivatives

Productive suffix **-us** makes it possible to form property names from most of the adjectives, changing only the part of speech, e.g. intensiiv/ne (intense) – intensiivs/us (intensity, intensiveness); musikaal/ne (musical) – musikaals/us (musicality, musicalness); soola/ne (salty, salt) – soolas/us (saltiness, salt); keeru/line (baffling, knotty, problematic) – keerulis/us (complexity, complexness); lopsakas (buxom, chubby, plump) – lopsak/us (fleshiness, obesity); vürtsikas (hot, spicy) – vürtsik/us (spicery, spiciness)

Suffix **-ndus** forms abstract names of substances or fields of action from concrete nouns, e.g. kauba/ndus (commerce); kirja/ndus (literature); koka/ndus (cookery, cooking, cuisine); maja/ndus (economy); metsa/ndus (forestry); teeni/ndus (service); veondus (transportation, shipping).

4.3. Adjective suffixes

It is difficult to group adjective suffixes by meaning because most of the suffixes can express several

meanings. Very often it is dependent on the derivative base.

The adjectives formed from the nouns often convey a comparative or possessive meaning, e.g analoogia (analogy) – analoogi/line (analogous); kriitika (criticism, critique) – kriiti/line (critical); värv (colour) – värvi/line (coloured); kasu (use, good) – kasu/lik (useful); noorus (youth) – noorus/lik (youthful).

The EWN derivational relations *derived/has_derived/derived_from* and *pertains_to/is_pertained_to* are namely prescribed for adjective suffixes.

5. Semantic relations of proper nouns

The main inspiration for our WSD system *semyhe* is Agirre and Rigau (1996) similar system that disambiguates the English noun senses based on WordNet hyponym/hypernym hierarchy, taking into consideration the distances between the nodes corresponding to the word senses in the WordNet tree as well as the density of the tree (Vider and Kaljurand, 2001).

In order to improve the operation of the program, the density of the words, that will be disambiguated should be increased. Up to now proper nouns were left out of disambiguation and they comprised 30% of the 0-analysed nouns. As our WSD system uses EstWN, it is essential that proper names encountered in the texts be added to it. Fortunately the EWN database structure includes a type of entry meant for proper names — *word_instance*.

Hyponymy is a relation between classes of entities. Individual entities, presented in texts as proper nouns and in EWN structure as *word_instance* entries, can also be said to belong to some class. To distinguish this relation from hyponymy it is labelled *has_instance/belongs_to_class* in EWN (Vossen, 1999). It is good because it makes it also possible for the WSD system to find out referee among *word_meaning* entries. Thus WSD system can make more precise decisions about the right word meaning, because meaningful context is more dense. Therefore we added all proper nouns existent in the WSD training corpus to EstWN and linked *belongs_to_class/ has_instance* relation to *word_meaning* entries (see Table 1).

Now the question is which proper noun links to which *word_meaning* entry. It seems only natural to link e.g. capital *has_instance* Tallinn, river *has_instance* Volga. It is also possible to link e.g. male, male person *has_instance* John. But is it right to link family *has_instance* Smith, for family refers to a social group, not a person?

Most proper nouns listed in the EstWN refer to a person. The next group as to the frequency is toponyms that refer to a location or place (city, state, land, region) or natural objects (river, mountain, lake, island etc).

6. Conclusions

In the Estonian language derivation is not a feature of morphology. As to the richness of meaning of the Estonian derivation system, the semantic relations existent in the EWN and labeled as *derived/has_derived/derived_from* clearly too scarce. Making use of the recognizability of the suffixes, it is possible to link the derived words with the derivational base words (semi)automatically, specifying the semantic relation on the basis of the meaning of the derivational suffix.

Specifying the semantic relation of proper nouns is of vital importance to increase the conceptual density in solving the wordnet-based WSD task. One should only be careful and persistent in achieving the target concept.

References

Agirre, E. and Rigau, G. 1996. Word Sense Disambiguation using Conceptual Density. In *COLING-96*

Kaalep, H.-J. 1997. An Estonian Morphological Analyser ant the Impact of a Corpus on Its Development. *Computers and the Humanities*, 31:115-133.

Kahusk, N. and Vider, K. 2002. Estonian Wordnet benefits from word sense disambiguation. In *Proceedings of the First International Global Wordnet Conference* (pp. 26-31). Central Institute of Indian Languages, Mysore, India.

Kasik, R. 1996. Eesti keele sõnatuletus. Tartu Ülikooli Kirjastus.

Vider, K., Paldre, L., Orav, H. and Õim, H. 1999. The Estonian Wordnet. In C. Kunze, editor, *Final Wordnets for German, French, Estonian and Czech*. EuroWordNet (LE-8328), Deliverable 2D014.

Vider, K. and Kaljurand, K. 2001. Automatic WSD: Does it Make Sense of Estonian? In *Proceedings of SENSEVAL-2: Second International Workshop on Evaluating Word Sense Disambiguating Systems* (pp. 159-162).

Vossen, P. (ed). 1999. EuroWordNet General Document. EuroWordNet (LE2-4003, LE4-8328), Part A, Final Document. http://www.hum.uva.nl/~ewn/docs.htm

RussNet: Building a Lexical Database for the Russian Language

Irina Azarova, Olga Mitrofanova, Anna Sinopalnikova, Maria Yavorskaya, Ilya Oparin

Applied Linguistics Department, Philological Faculty, Saint-Petersburg University
Universitetskaya nab. 11, Saint-Petersburg, Russia
azic@bsr.spb.ru , asinopalnikova@yahoo.com, yav_mas@hotmail.com

Absract

The paper describes the on-going work on creating the WordNet-type lexicon for Russian, so called RussNet. The project started 3 years ago, preliminary results will be available at www.phil.pu.ru. The existing database contains verbs, nouns, and adjectives, the number of senses amounting to 2500.

The Top Ontology of RussNet is under construction, it will be co-ordinated with that of EuroWN. RussNet has inherited EuroWN language-internal relations. Several types of derivational links are added to describe Cross-Part-Of-Speech relations as well as Inner-Part-Of-Speech ones. Adjective-to-noun and verb-to-noun relations of words in collocations are described in details.

An overview of methods used for construction of the Russian WordNet is presented, the procedure of sense definition generation is also discussed.

1. RussNet Structure

1.1.Vocabulary

For the RussNet structure we accepted the general approach, presenting only **Standard Russian** lexis, as opposed to various terminological subsets. The position doesn't prevent us from including those terms that were incorporated into the common language.

On the one hand this approach follows Russian lexicography tradition and on the other hand allows us to provide first and foremost **frequently-used current** vocabulary, that will be exploited by the majority of users. The main sources for such words are newspaper and magazine articles.

1.2. Inherited Features in RussNet

- RussNet is structured along the same lines as Princeton WN, EWN (Vossen, 1998, Miller et al, 1993) and other wordnets: words are grouped into synonym sets (**synsets**), each representing one underlying concept.
- Synsets in their turn are linked by means of various **Language Internal Relations** (LIR), such as hyponymy/hyperonymy, antonymy, meronymy/holonymy, entailment, causation, etc., hyponymy/hyperonymy being the most important one.
- RussNet consists of **4 interrelated files** for basic POS: nouns, verbs, adjectives and adverbs. So far we dealt only with 3 of them, but later we are going to add adverbs as well.
- Each of the 4 files contains a number of hyperonymy trees, with concepts of top levels constituting so called **Top Ontology**.
- Now, we are elaborating mainly internal structure of Russian wordnet and are not dealing with **Inter-Lingual-Index** (ILI).

2. Synset Formation

There are two different ways to define synonymy:

- in terms of substitution
- in terms of semantic similarity.

Although in EWN the weaker notion of synonymy is adopted: «two words are synonyms if there is a statement (class of statements) in which they can be interchanged without affecting truth value», we have to combine substitution method with that of semantic similarity. The reason for such a decision is as follows: in Russian there are many words which are not interchangeable in a context because of the syntactic, stylistic, expressive differences, but they are considered by native speakers as having similar meanings, denoting the same objects, entities, etc., e.g. aspect opposition for verbs.

There are two types of synonymy dictionaries for Russian:

- New Explanatory Dictionary of Russian Synonyms (Apresjan et al.) is following the substitution strategy. The first issue of this dictionary was published in 1999, but so far it includes 132 entries only.
- Dictionary of Russian Synonyms (Evgenjeva,1970) & Explanatory Dictionary of Russian Verbs (Babenko, 1999) are based on semantic similarity.

Unfortunately, conventional Russian lexical resources may be used only partially because they don't cover all the lexis, the words definitions provided are made according to inconsistent patterns, and they may even obscure real semantic relations between words. That's why we can't simply import the data from those resources into RussNet without correcting it by means of our own lexical research procedures.

We begin with the collection of word senses for particular semantic groups of Russian words such as emotional verbs, nouns denoting the social relations and so on. The words realising the hyperlexeme sense were picked out from the sample of fiction or newspaper texts. A mean sample size ranges from 200 to 400 thousand word occurrences, from which about 150 core words and 70 peripheral words with appropriate senses were usually chosen. Having examined the synonymic relation in such groups we saw that words with the most abstract sense were encountered with relatively higher frequency and they would have synonymic equivalents. The hyponyms of the group were rare and may have derivational synonyms, but quite a few synonyms with different roots. So the collected words may be considered to be dominant representatives for respective synsets. Afterwards, extending the sample size or using synonymic information given in a conventional dictionary, we may expand synonymic sets with extra members.

3. Problems and discussion

3.1. Derivation

The Russian vocabulary, in particular verbs and nouns, is characterised by the high degree of derivation motivation. For example, dealing with verbs of thinking in Russian and English, we can see that there is about dozen of verbs with different roots in English (*to think, to contemplate, to consider, to regard, to reflect, to muse, to ponder, to cogitate, to meditate, , to conceive, to imagine, to picture etc*), and only 3 such items in Russian (*думать, мыслить, мозговать*), with a number of affixed derivatives amounting to 30 resultant verbs. Thus the total number of lexemes in Russian may be twice as much as that in English, while the situation with roots may be quite the opposite (Mitrofanova, 1999). From the point of view of frequency this causes specific distribution of lexical items in texts: it is rather flat in English in comparison with Russian sharp peak of frequencies for a hyperonym of this group *думать (think)* see Table 1, Table 2.

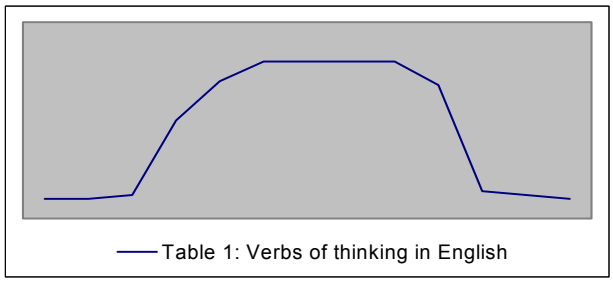


Table 1: Verbs of thinking in English

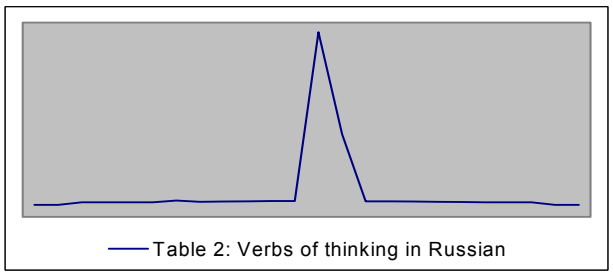


Table 2: Verbs of thinking in Russian

In many cases semantic relations between stem word and its derivatives couldn't be treated in terms of EWN Language Internal Relations (Vossen, 1998). They are more complicated: the main difficulty is that they are relations **between lexical items**, not synsets. Other reasons why we have to introduce new links are as follows:

- There are many almost **unlimited** derivational chains: verb denoting process => noun denoting the process => attribute denoting the relevance to the process => adverb denoting the changing quality and so on, e.g. *удивлять (to astonish, to surprise) - удивление (astonishment) - удивленный (surprised) - удивленно (surprisingly).*
- The important traits of these chains are, that derivatives may be used freely in **paraphrases**: the motivating item may substitute the motivated ones in syntactic transformations. For example, a Russian noun *проверка* (*a check*) is paraphrased as a denotation of the process expressed by Russian verbs *проверить, проверять, проверяться, проверяться (to check, to be checked)*. These links may be useful for syntactic analysis.
- **Lexical meaning** of derivatives is determined by that of the stem word.
- We would like to stress that verbal nouns inherit also the **syntactical** features of the motivating words. So if we describe the complex system of verb valences, they would be reproduced with little (and well known) changes by nouns denoting the same action or quality, on the one hand, and participants of action, on the other hand.

In those cases when it is possible we regard derivational relations in terms of LIR:

- **SYNONYMY -** relations between words which have the same root and different sets of affixes. They are not expressive and their senses differ so slightly that not every native speaker (researcher) is able to explain the distinction between them. Those words are also rarely interchangeable in the same context. *Семья – семейство (family), зло (malice) – злоба (malice, anger) – злость (malicious anger), бунтарь – бунтовщик (rebel, insurgent, mutineer, rioter), беда (misfortune, calamity) – бедствие (calamity, disaster).*
- **NEAR_SYNONYMY -** relations between
  - verb and abstract nouns, denoting processes of the same nature, e.g. *двигаться => движение (move => movement)*,
  - adjectives and abstract nouns, denoting characteristics and qualities, e.g. *красный => краснота (red => redness)*,
  - adjectives and nouns, e.g. гриб => грибной (fungus => relative to fungi)
  - verbs and adjectives, e.g. гнить => гнилой (rot => rotten).

In other cases we have to introduce a set of Derivational analogues of LIR, such as:

- **DERIVATIONAL_SYNONYMY** – relation between neutral words and their expressive derivatives. As those words differ from their stem word in style, they are not interchangeable in context, e. g. *старик (old man) => старикан, старикашка* (*impolite appeal to an old man), дом (house) – домик (house to which the speaker has positive emotions)*. Here we follow the idea, offered in Czech WordNet, of special attributes introduction. Thus *домик* will have X_EXPRESSES_ POSITIVE_EMOTION, while *старикашка* – X_EXPRESSES_IMPOLITE .
- **DERIVATIONAL_HYPONYMY** – verb-to-verb, noun-to-noun, adjective-to-adjective relations of following types. For verbs we may use
  - specific attributes X_HAS_INCHOATIVE or X_HAS_SPECIFIED_DURATION for actions restricted in time duration (inchoatives), e.g. *петь => запеть (to sing => to begin to sing), сидеть => посидеть (to sit => to sit for a while), сидеть => просидеть (to sit => to sit for a long time)*;
  - an attribute X_HAS_SPECIFIED_RECURRENCY for actions repeated only once or several times, e.g. *кричать => крикнуть, покрикивать (to shout => to shout out once, to shout not aloud many times);*
  - an attribute X_HAS_SPECIFIED_NUMBER for actions, having many objects involved, e.g. *думать - раздумывать (to think - to ponder about many things for a long time), резать - вырезать (to cut - to cut out some part from many things)*, and so on.

These special verbal derivatives interacting in a complex manner with an aspect category of verbs and having semi-grammatical nature. We still don't know in which manner to treat them, on the one hand, aspect pairs look like very

close synonyms, though on the other hand, they realise a very important semantic opposition, such as activity ⇔ action. We may introduce specific attributes, as follows: X_HAS_IMPERFECT, X_HAS_ PERFECT.

- For nouns and adjectives we may add attributes X_IS_SMALL and X_IS_BIG, and possibly several others, when the clear sense component is added by some affixes to the stem word meaning, and the resultant word couldn't be regarded as purely expressive variants; this why we should treat such pairs as *стол => столик (table => small table), дом => домишко (house => small house), пожар => пожарище (fire => big fire), громадный => громаднейший (huge => very huge)* as derivational hyperonym - hyponym.

We should note that the majority of these derivational variants doesn't belong to the core of Russian lexis because of their infrequency in texts. However, the highly inflected nature of Russian may turn any potential derivative into common and frequently used one, that's why all derivational regular models should be taken into account.
Moreover, we may find several cases when an expressive shade may disappear, then a word would change expressive synonym status for a synonym position. Another example of extending the sphere of usage for diminutives may be seen in the Russian spoken language (usually by women), when these words function as oral equivalents for their neutral motivating counterparts, so we may expect that in future they have a chance to become colourless synonyms.
Expressive synonyms and hyponyms may exist beyond the derivational scope, but in these cases they are rather few, irregular, and disputable, that's why it would be adequate to include them into the synset with a proper attribute.

- **DERIVATIONAL_ROLE_RELATIONS** are established to link a verb to its derivatives, designating action participants, such as ROLE_DERIVED_AGENT, ROLE_DERIVED_ OBJECT, ROLE_DERIVED_INSTRUMENT, ROLE_DERIVED_ LOCATION and so on, e.g. *сеять => сеятель, сеянец, сеялка (to sow => sower, seedling, seeding-machine*). The link in the opposite direction is a realisation of the semantic link INVOLVED_IN_ACTION. We are inclined to treat such cases as a specific derivational relation because the semantic link usually has wider scope, e.g. *принимать => приемник (receive => radio set = receiver)*, the object is involved in the first place into the situation *слушать (listen)*. This is usual for complex activity nomination, which as a rule is designated with regard to one action varying from one language to another, e.g. *шить => швея (to sew => seamstress)*. Above we have mentioned the inheritance of syntactic features, moreover, the collocation restrictions of stem verbs may be inherited by their derivatives.

3.2 Adjectives in RussNet

As there is no common solution for treatment of adjectives in EWN, we offer the following one.
We comply with the idea of GermaNet to make use of hyponymy relations wherever it is possible, but our German colleges determine hierarchical structure of adjectives according to semantic fields, while we regard adjectival hyperonymy in terms of their collocations with nouns. We received preliminary results which prove that on the level of adjectives grouping and nouns tree hyperlexeme, it is the **adjective** in Russian that **predicts** certain type of **nouns to collocate with** it, and not vise versa. For example, meaning of *долговязый (lanky)* involves the pointer to a human being, i.e. it can collocate with such nouns as *мальчик (a boy), человек (a man), папа (a father).*
We are prone to the opinion that **adjectival hyponymy trees** can be built according to their collocation with nouns from different levels of hyponymic tree. For example, lets take two adjectives, which express the similar semantic quality – denotation of *height*. In case when one adjective – *высокий (tall)*– may collocate with all nouns denoting "entity": objects, animals, humans and so on, while the other – *рослый (well-grown, srapping)* – collocates only with a certain part of the tree – human beings, the first one may be thought as hyponym for the second one. So checking the co-occurrence of adjectives with nouns, we are to produce hyponymic structure for groups of adjectives denoting the similar quality.

3.3. Verb Valencies

It is generally accepted that syntactic features of words, especially verbs, are determined by their semantic properties, that the meaning of a verb outlines the form and semantic features of words accompanying it.
The semantic and syntactic structure of verb arguments is called the **valencies frame**. Valencies may be thought in terms of morphological noun forms, which are obligatory or optional. This characteristic is vital for Russian syntax, as well as for that of other Slavonic languages (Pala, Sevecek, 1999).
Verbs have different valencies frames associated with dfferent meanings, cf.

- *Бить (посуду) [to crash]*
- *Бить (в барабан) [to bit into]*
- *Бить (врага) [to fight against]*

The minimal form of valency description implies the noun case specification, often it needs the indication of a preposition (or number of prepositions).
We may fix the **semantic** features of nouns as well, which a verb can take as arguments in a sentence. It means we want to use top-level concepts, to deal with **classes** of words, including verb-to-class relations in the synsets. In the example above, the argument of a verb in the first frame is a fragile object, in the second – musical instrument, more precisely – percussion, in the third – human beings, military units and so on. This references to the hyponymic tree structure of nouns would be very helpful for syntactic description as well, though sometimes this relation may be very comlicated.
The situation with valencies frames is not clear due to versatility of syntactic preferences of verbs included into a synset, while sometimes they behave uniformly. We'll use **a list of valencies frames** for a synset specifying which one fits the member of a synset. The set of frames is better than separate verb description, because in this case the paradigm influencing the native speaker is presented.
Moreover, it would be very useful to represent the inheritance of syntactic frames of a hyperonym by its hyponyms, e. g. *двигаться (to move) ==> идти (to walk):* hyperonym *двигаться* has valencies frames: (a) "starting

point – location", (b) "destination point – location", which are inherited by its hyponym *идти*.

4. Definition Generation

4.1. Subset Sense Definition

We still don't speak about definition generation procedure, but it's vital to have in mind guidelines for definition formulation because dictionary ones for a long time have been a target for an extensive criticism. In this respect we propose several key notes.

4.1.1. Hyponymic Definition

The definition of a synset incorporated into the hyponymic (or troponymic) tree should be constructed on the following pattern "the dominant **lexeme** of the **hyper** level **plus** a **distinguishing part** showing difference between co-hyponyms", e.g. *плыть (to swim)* has hyperlexeme: «to move in certain direction» + differentiation: «on the surface or in depth of water using special organs», *лететь (to fly)* has hyperlexeme: «to move in certain direction» + differentiation: «in the air using wings». In this case there is no Russian hyperlexeme denoting *moving in some direction*, though it's important to oppose this way of moving to the other one in various direction, with repetitions, to and fro.

It's clear that in case of a large number of co-hyponyms the problem may become practically insolvable because of a great number of necessary differential features, then it would be better to use other types of defining or artificial names (used in GermaNet) uniting several lexemes into a cluster.

4.1.2. Meronymic Definition

The definition of a synset incorporated into the meronymic relations may be based on either holonym, or meronym.

In the first case, a holonym is the referential part of the definition (similar to hyperlexeme), but a simple indication that something is a part of the holonym is not sufficient, so it is usually supplied with a special function (for artefacts) or construction peculiarties. For example, structure «part + construction characteristic + holonym + function» may be used: *крыша (roof)* = «the upper part of the building, covering it from precipitation».

In the second case, a limited number of meronyms may be used for generation of list-type definition, e.g. *фигура (chessman)*: «king, queen, castle, knight, bishop in chess opposed to pawns».

4.1.3. Derivational Definition

In those cases when a synset is associated with a purely derivational link we use a definition describing the additional sense of the derivational affixes, e.g. *столик* «a small table», *генеральша* «general's wife».

4.1.4. Semantic Pointer Definition

The simplest way of defining the quality is to show the synonyms expressing it, which are united in the synset, so in this case we have a rudimentary definition equal to an ordinary synset. This type of definition is frequent for adjectives and adverbs.

Antonymic definition is adequate in those cases when one member of the antonymic pair is marked showing the positive content while the other shows its absence, e.g. *глупый (foolish)* «not clever» <=> *умный (clever)* «having the intellect».

Causative definition is alike the derivational one so as it makes implicit the causative copula and the final state of transition, in Russian there is a specific affix with anti-causative meaning, e.g. *поднять (raise)*: *каузировать подняться (cause to rise)*. Usually in such a definition the artificial causative is used, which is the transliteration of English *cause*, because a Russian equivalent *заставить* means 'to enforce', that is not neutral at all.

Moreover, using semantic attributes, such as X_HAS_IMPERFECT, X_HAS_PERFECT, X_IS_SMALL, X_IS_BIG etc., incorporated into the WordNet structure, we may later elaborate a procedure for automatic definition generation.

4. Conclusions

To sum up we may say that RussNet presently covers the core of the Russian lexis (the resulting number of synsets is more than 2500). So it can be regarded as a reliable starting point for further extending and elaboration of the system, which will be carried out by addition of peripheral groups of words, emotionally coloured lexis and derivatives, in particular. This should enrich the content of the database. The introduction of new relations allows us to perform more adequate semantic analysis of the Russian language.

5. References

Apresjan, U. (ed.) (1997). Новый объяснительный словарь синонимов русского языка (=New Explanatory Dictionary of Russian Synonyms). Moscow.

Babenko, L. (ed.) (1999).Обяснительный словарь русских глаголов (=Explanatory Dictionary of Russian Verbs). Moscow.

Evgenjeva, A. (ed.) (1970). Словарь синонимов русского языка. (=Dictionary of Russian Synonyms) (vol 1-2). Leningrad.

Miller, G. et al (1993). Five Papers on WordNet. Technical Report, Cognitive Science Laboratory, Princeton University. ftp://ftp.cogsci.priceton.edu/pub/wordnet/5papers.ps

Mitrofanova, O. (1999). Структурный анализ сигнификативного значения: на материале глаголов процесса мышления английского и русского языков (Structural Analysis of Sense: Verbs of Knowing in English and Russian). PhD thesis. St-Petersburg State University, Philological Faculty, Department of Applied, Structural and Mathematical Linguistics.

Naumann, K. (2000). Adjectives in GermaNet. http://www.sfs.nphil.uni-tuebingen.de/Adj.html

Ozhegov, S. (1984). Словарь русского языка (=Dictionary of Russian). Moscow.

Ozhegov, S., Shvedova, N.(1992). Толковый словарь

русского языка (=Explanatory Dictionary of Russian). Moscow.

Pala, K., Sevecek, P. (1999). The Czech WordNet, EuroWordNet (LE-8928). Deliverable 2D014. http://www.hum.uva.nl./~ewn/docs.html

Vossen, P. (1998). EuroWordNet: A Multilingual Database with Lexical Semantic Network. Dodrecht: Kluwer.

Словарь современного литературного русского языка.(1991). (=Dictionary of Modern Literary Russian) (vol. 1-17). Moscow-Leningrad.

Development and Use of Thesaurus of Russian Language RuThes

Natalia V. Loukachevitch* and Boris V. Dobrov*

Research Computing Center of Moscow State University
339, Research Computing Center of Moscow State University,
Vorobyevy Gory, Moscow, 119899, Russia
{louk, dobroff}@mail.cir.ru

Abstract

In the paper we describe the main principles of developing Thesaurus of Russian Language RuThes, which is constructed specially as a tool for automatic text processing. The thesaurus contains more than 95 thousands words and multiword expressions. It has a specific system of conceptual relations, describing existential properties of concepts. Means of description and disambiguation of lexical ambiguity are discussed. The technology of development the bilingual resource based on RuThes is described. We also consider current stage of the thesaurus and describe the use of the Thesaurus in various applications of automatic text processing.

1. Introduction

Large volumes of electronic text collections require mighty tools for their processing. Texts in these collections include thousands of various words and syntactic constructions, they can have various sizes and styles. All these factors pose an important question what linguistic resources facilitating processing large collections of electronic documents could be.

The paper is devoted to description of main principles of development of the Thesaurus of Russian Language RuThes, which belongs to the same type of such linguistic resources as WordNet (Miller et al., 1990) and EuroWordNet (Climent et al., 1996).

This work arises from our experience in creation of domain-specific Thesaurus on Sociopolitical Life, which was constructed as a tool for automatic conceptual indexing in the large domain of social life (Loukachevitch et al., 1999). Development of Sociopolitical Thesaurus for automatic text processing of large text collections required use of two different traditions: the tradition of development of information-retrieval thesauri for manual indexing, which pay specific attention to terminology and representation of domain-specific relations (LIV, 1994; UNBIS, 1976; EUROVOC, 1995), and the tradition of development of linguistic resources with their attention to description of single words, lexical ambiguity, semantic relations.

The Sociopolitical thesaurus was used in such applications of automatic text processing as term disambiguation, automatic conceptual indexing, knowledge-based text categorization, automatic text summarization (Loukachevitch et al., 1999). The Sociopolitical thesaurus is an information retrieval tool in University Information System RUSSIA (Russian inter-University Social Sciences Information and Analytical Consortium; www.cir.ru/eng/).

The technique of text processing using Sociopolitical thesaurus is based on lexical cohesion property of coherent texts, that is, the thesaurus relations were used to find semantically related sets of terms in texts (Loukachevitch & Dobrov, 2000). For several years the results of the text processing were tested through manual analysis. We tried to understand how thesaurus relations work in the thematic structure of coherent texts. This activity led us to development of RuThes, a linguistic resource for automatic text processing of large Russian text collections.

Now thesaurus RuThes includes 95 thousand Russian words (nouns, verbs, adjectives), expressions and terms, 105 thousand senses, 42 thousand concepts (synsets). In contrast to European wordnets we began to describe Russian-English relations of RuThes after considerable part of RuThes had been already created.

2. General Structure of RuThes

RuThes is a hierarchical net of concepts. Every concept has a set of its textual expressions, a synonymic row (a synset in terminology of WordNet) and a set of relations with other concepts of the thesaurus. So its general structure is the same as the structure of WordNet and EuroWordNet.

RuThes consists of two main parts: general lexicon and Thesaurus on Sociopolitical life (Figure 1).

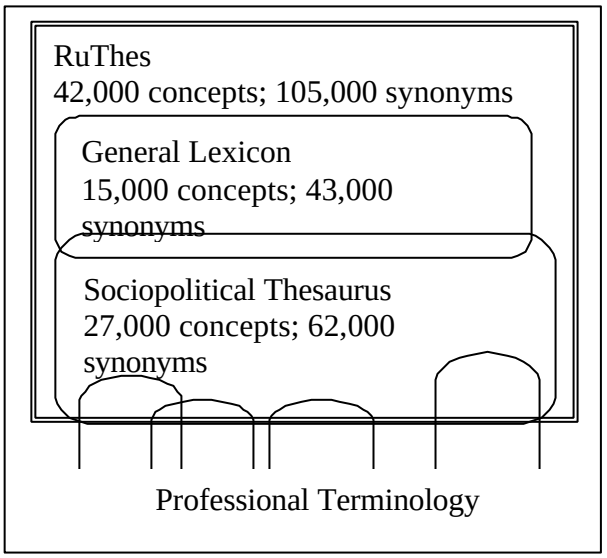


Figure 1. General Structure of RuThes

Thesaurus on sociopolitical life includes concepts which correspond to a domain of social life, these words and terms are usually thematically significant: *car, river, town, economy, computer, sports, serviceman* and many others. The domain of Sociopolitical Thesaurus is not domain of social research, but comprises situations and

problems of social life, which are discussed in official documents and newspapers. Sociopolitical thesaurus encompasses as words and expressions usually included in general explanation dictionaries and terminology of such domains as economy, law, defense and others. Besides Sociopolitical thesaurus includes the geographical subdomain describing 7000 geographical names.

General lexicon contains concepts, which can be met in texts of any domains, for example, *part, create, new*. In texts these words and expressions are less significant, they usually express relations and features of main entities discussed in texts. Besides general lexical contains concepts expressing human emotions, feelings, personal human relationships. General lexicon contains 15 thousand concepts (from 42 thousand in RuThes), 33 thousand words and language expressions (from 95 in RuThes).

The main goal of the division was as follows: this borderline separates very lexically ambiguous area from much less ambiguous, very relational area from much more thematically significant. The result of this division is that Sociopolitical thesaurus is used in various applications of automatic text processing for several years. General lexicon is now under development. Its numerous multiple senses are added and corrected. But Sociopolitical thesaurus and General lexicon are parts of the same system. Therefore if necessary, knowledge from General lexicon is used in computer applications together with Sociopolitical thesaurus.

3. Synsets in RuThes

3.1. Description of different parts of speech in RuThes

Elements of synsets in RuThes (below Thesaurus) are: single words (nouns, verbs, adjectives), noun groups, verb groups, adjective groups. We describe semantically equivalent words belonging to different parts of speech as elements of the same synset - our tradition from 1994.

Here and below we present examples in English to show how our decisions would look in English. So a synset looks like: *partake, participate, participation, participative, participatory, take part.*

Every synonym has its part of speech tag, and this information can be used in automatic text processing.

Incorporation of parts of speech makes description of relations more consistent. If we create three different concepts for different parts of speech and know about their semantic equivalence, we have to repeat similar relations for them. This leads to inconsistency in description of relations. For example, in WordNet 1.6. we can see that word *engagement* is in the same synset as *participation*, but there is no relation between *engage* and *participate*. However in Webster (1999) we can find the following example: *to engage in business or politics*, which means that the relation between verbs has to be described.

This incorporation also means that the hierarchy, the set of relations is the same for all parts of speech.

3.2. Multiword expressions in RuThes

We pay special attention to description of multiword expressions and terms as sources for lexical disambiguation and representation of situational and encyclopedic knowledge. A number of multiword expressions in the Thesaurus is 42 thousand (of 95 thousand).

We began development of Sociopolitical thesaurus using semi-automatic methods to find multiword terms in text collections of official documents and newspaper articles. Our procedure of terms acquisition consisted of two stages. At the first stage term-like expressions were automatically identified in the texts of the corpus. Rules defining term-like expressions included syntactical and lexical conditions. At the second stage our specialists had to look through the revealed expressions, choose terms from them and add new terms to the Thesaurus (Lukashevich, 1995). The procedure was working during four years, processed more than 200 Mb of texts and collected more than 200 thousand term-like expressions. It was stopped because it became difficult to find new useful terms, terminology coverage became very high.

From that experience we understood how important to add unambiguous multiword expressions containing ambiguous words to conceptual synonymic rows. They diminish percentage of ambiguous words in a text and help disambiguate neighbor expressions. Since we specially seek unambiguous multiword expressions in any sources we have: in glosses and examples of dictionaries, in text collections. For example, the following expressions could be added to synonymic rows of WordNet and are very useful in automatic text processing:

*Petition to god (*sense2 of noun *petition);*
transfer to private ownership (verb *privatize),*
*conductor of an orchestra (*sense1 of noun *conductor)*

As an example how a full synonymic row of multiword expressions could looks, let us see the synonymic row of Russian concept *ZDRAVOOHRANENIE* (*PUBLIC HEALTH*), which is similar to the following English list:

p*ublic health, community health, health care, health care sector, health care system, health field, health of population, health promotion, provision of health, public health field.*

So one can see how this list diminishes necessity to disambiguate such “difficult” words as *care, sector, field, system, public.*

A multiword expression can also initiate a new concept. There are several factors that can make possible creation of a new concept based on a multiword expression:

- A multiword expression presents an important and frequent enough subtype of a concept already described in the Thesaurus;
- A multiword expression is unambiguous and contains very ambiguous words;
- A multiword expression has conceptual relations that do not follow from its constituent parts;
- A multiword expression has relations with concepts of lower levels, based on single words, so a new concept additionally structures the thesaurus knowledge, can join separate conceptual substructures of the thesaurus net.

3.3. Name of concept

Every concept of the thesaurus has a unique name, which has to be clear and unambiguous for native speakers. Name of a concept can be

- one of unambiguous synonyms;
- a multiword term which is unambiguous and possible as one of textual expression corresponding to a concept;
- a pair of synonyms;
- a synonym with a fragment of the definition of a concept.

This name presents the whole synonymic row in different representations of results of text processing, for example, in structural summary of a text which is very convenient in cross-language information retrieval (Loukachevitch & Dobrov, 2000) or as explanation means for knowledge-based text categorization systems.

A concept usually does not have a full gloss but formulation of its name has to be enough to find a corresponding sense in explanation dictionaries if necessary.

4. Description of lexical ambiguity

In linguistic resources intended for automatic text processing there is a serious problem how detailed division of senses must be. The sources of the problem are as follows:

- it is difficult to disambiguate close meanings during automatic text processing in large domains;
- it is impossible to refine query with help of a user because a user must not understand and distinguish subtle linguistic distinctions;
- at last close meanings (even if we have divided and can disambiguate them) are often both relevant or not relevant to a query.

Therefore we have to understand, what types of ambiguous terms it is necessary to distinguish and represent as different concepts of the Thesaurus.

In a linguistic resource represented as a conceptual net the desire to reduce number of senses is in contradiction with other problem: if two senses have different sets of conceptual relations (especially different sets of links to lower levels of the conceptual net), then their clustering can lead to loss of descriptive clarity and new problems in efficiency of automatic text processing.

Therefore in RuThes we do not cluster senses that have different hyponyms and/or parts. If the difference between sets of conceptual relations consists only of hypernyms, sense clustering is possible.

For example, it is impossible to cluster concepts corresponding to the senses of word *building* as process and result as proposed in (Pustejovsky, 1995), because in the Thesaurus difference in conceptual relations between the concepts is significant. Compare fragments of lower levels corresponding to these concepts:

CONSTRUCTION OF BUILDINGS (build, building, building construction, construct, construction….)
RESIDENTIAL CONSTRUCTION (home construction, homebuilding, home building…)
...
COSTRUCTION EQUIPMENT(building equipment)
TOWER CRANE
BULLDOZER
EXCAVATION EQUIPMENT
...

BUILDING…
PUBLIC BUILDING
ADMINISTATIVE BUILDING
MUSEUM…
SCHOOL BUILDING…
...
RESIDENTIAL BUILDING
APARTMENT HOUSE
VILLA
...

The problem of description of close senses became less serious if it is possible to describe relations between corresponding concepts. The relations of RuThes allow us to connect various types of polysemic senses, and in automatic processing if it was not possible to distinguish a correct meaning, the most broad concept among all related senses is chosen in default way. In general, it is possible to have a special indicator, showing which concept can be chosen in default.

For example, we can introduce two concepts *SCHOOL (EDUCATIONAL ORGANIZATION)* and *SCHOOL BUILDING*, connect them with relation WHOLE-PART and include word *school* in synsets of both concepts. In automatic text processing if it is not proven that a school building is discussed, concept *SCHOOL (EDUCATIONAL ORGANIZATION)* is chosen. It means that there is no real difference between description of these polysemic senses as two concepts or a single concept. In RuThes choice of separated or clustered description of close polysemic senses depends on if sense concepts are central in the thesaurus net and require their own sets of lower relations or they are peripheral.

5. Relations in RuThes

Linguistic resources intended for automatic text processing usually include descriptions of semantic relations between their entries such as ‘part’, ‘agent’, ‘material’, ‘time’, ‘cause’, ‘result’, and others. At the same time when huge conceptual-based resources are developed, it is supposed that these resources have to be used in automatic text processing of large and heterogeneous text collections. However, at present text processing systems can not provide deep linguistic analysis of such texts. It means that a computer system can not check if described relations are valid in a current text. Therefore other, not semantic, characteristics of any relation become especially important, if a relation can change or disappear in a specific situation described in a text. These characteristics can be considered as existential characteristic of a relation.

Therefore if we describe that a tree is a part of a forest, but in fact a tree can grow in a lot of other places, the system can not rely on this description because in a specific text the relation can be not valid. It can lead to problems in efficiency of automatic text processing.

To test changeability of a relation between concepts *C1* and *C2* it is necessary to answer the following questions:

1) if every example of a concept *C1* has the relation with an example of a concept *C2* (and vice versa);
2) if an example of concept *C1* has the relation with *C2* (or its example) during all time of its existence, for example, concept *GARMENT* can be considered as *CONSUMER GOODS* (as described in WordNet 1.6), but when a specific person wears garment, it ceases to be goods;
3) if all properties of a concept *C1* are properties of concept *C2*, for example, concept *SHIPWRECK* loses very important properties of concept *SHIP*.
4) if existence of a concept *C1* is impossible without existence of concept *C2* or existence of an example of a concept *C1* is impossible without an example of concept *C2* (dependency relations (Guarino, 1998)), for example, existence of concept *BOILING* is impossible without existence of concept *LIQUID*.

At present description of relations in RuThes do not present semantic nature of relations distinct from hyponymy-hyperonymy relations and part-whole relations, but its existential properties. At the same time it gives additional very powerful possibility not to decide what a semantic name of a relation can be. It is very important for complex relations such as *CREDITOR – BANCRUPCY*: if the name of the relation is 'agent' or 'source' or both.

Current names of conceptual relations in RuThes were introduced in earlier version of Sociopolitical thesaurus and arise from names of relations in conventional information retrieval thesauri. There are three basic relations:

1) BT-NT relations (broader-narrower terms) is now used as equivalent to hyponym-hypernym relations;
2) WHOLE-PART relations for descriptions of conventional parts, properties and participants of situations;
3) RT (related term) relations for description of all other relations, which can be symmetrical and nonsymmetrical. Nonsymmetrical RT relation is denoted as RT1 – RT2 and serves for description of dependency relations.

Let us see fragments of description of concept *RIVER* to see usage of PART and RT relations

RIVER

PART	*RAPIDS OF A RIVER* (Russian 'bistrina')
PART	*WATERFALL* (Russian 'vodopad')
PART	*MOUTH OF A RIVER* (Russian 'ust'e')
RT1	*FRESHWATER* ('presnaya voda') /* concept *RIVER* does not exist without existence of concept *FRESHWATER* therefore there is a dependency relation denoted as RT1. At the same time a lot of concepts depend on existence of concept RIVER. So below reverse relation RT2 is used */
RT2	*CATCHMENT BASIN* (Russian 'bassein reki')
RT2	*HYDROELECTRIC PLANT* ('gidroelectrostancia')
RT2	*EMBANKMENT* ('nabereznaya')
RT2	*BOTTOMLAND* ('poima')
RT2	*RIVER TRANSPORT* ('rechnoi transport')
RT2	*SLUICE GATE* ('shljuz')

If for a BT or WHOLE relation there is an answer 'OFTEN' to one of questions 1-3, then a relation is marked with special modifiers.

If a relation can be considered as a default relation or there are only two main alternatives, we mark the relation with modifier V (variability)

If a relation exists during most time of existence of an example of a concept, we mark a relation with modifier A (aspect, point of view). The same modifier is used if a relation does not preserve all properties of an upper concept. For example:

PENSIONER

BT_V	*OLDER PERSON*
BT_V	*DISADVANTAGED PERSON*
$WHOLE_A$	*PENSION SYSTEM*

So we described that a pensioner is often an older person and a disadvantaged person. A pensioner is a role in pension system, which does not characterize it fully because of two first relations. In fact, a pensioner is also a social status. Therefore if a text mentions pensioners, it does not always mean that the text discusses some problems of pension system.

Every type of conceptual relations has its own set of properties such as transitivity and inheritance. Modifiers restrict transitivity of relations (Loukachevitch & Dobrov, 2002).

6. Lexical coverage of RuThes

Now thesaurus RuThes includes 95 thousand Russian words (nouns, verbs, adjectives), expressions and terms, 105 thousand senses, 42 thousand concepts (synsets). At present we have finished comparison of lexical units in RuThes and in a text collection of more than 600 thousand documents (Russian official documents and newspaper articles). Analysis of 100,000 most frequent lemmas of the collection (frequency > 25) showed that about 7 thousand lemmas are necessary to describe in RuThes. We plan to continue study of the text collection and to add new lexical units in RuThes for next 100 thousand lemmas (frequency > 10). We suppose that this stage will give us other 5-7 thousand words to include in RuThes.

The lexical analysis of the collection allows us to describe new words, not included in contemporary Russian dictionaries, and see new usage of words that are considered in the dictionaries as obsolete.

Other important stage of our current work is verification of sense representation for polysemic and homonymic words in RuThes. Beginning from very

frequent words we analyze senses of every lexeme described in various dictionaries of Russian language (Ozhegov & Shvedova, 1999; BTS, 1998) and decide if

a) all senses of a lexeme have to be represented;
b) there are obsolete senses;
c) different senses can be represented as a single concept;
d) a sense is only used within multiword expression.

So current stage of development of RuThes can be characterized as verification and correction.

7. RuThes and English linguistic resources

Development of cross lingual linguistic resources is a very important task. For Russians bilingual text processing of Russian to English and English to Russian is especially significant. We began development of RuThes from Sociopolitical thesaurus, which is an important searching tool in our information system. To provide bilingual retrieval in our information system we began to develop Russian-English Sociopolitical Thesaurus. It means that we could not connect RuThes and WordNet because of absence of significant in our technology concepts of the sociopolitical domain in WordNet. Besides we considered collection of multiword terms as very important for any language. The following list presents English terms included to English part of Sociopolitical Thesaurus recently and not included to WordNet: *wheelchair user, construction area, airline ticket, travel field, home building, civil rights activist, top manager, produce market, cargo shipper, stress disorder* and others (terms are extracted from newspapers).

Development of bilingual Sociopolitical thesaurus has the following main stages.

At first Russian terms were translated into English using traditional bilingual dictionaries (Apresyan & Mednikova, 2000; Multilex, 1996). We received 30 thousand terms in the English part of our Thesaurus. However these translation could not provide rich synonymic rows we needed and could not provide terms describing phenomena that are absent in Russia but are significant for other countries.

Therefore at the second stage we took well-known American and British dictionaries and thesauri: Webster dictionary (1999), Longman dictionary (1995), Collins (1990), WordNet (Miller et al., 1990), Thesaurus Roget's (1991), information retrieval thesauri Legislative Indexing Vocabulary (LIV, 1994), EUROVOC (1995), UNBIS (1976)). Our specialists analyzed these resources and manually extracted terms contained in these resources as vocabulary entries, parts of explanations, examples.

Therefore an English expression can have a mark, indicating its origin. For example, a concept *EQUALITY BETWEEN MEN AND WOMEN* has the following synonymic expressions:

equal rights for women (WordNet's gloss)
equal rights of men and women (EUROVOC)
*equality between sexes (*Multilex*)*
*equality between women and men (*texts - documents of Council of Europe)
gender equality (texts)
sex equality (texts*)*.

Text variants of related concept *SEX DESCRIMINATION* are as follows*:*

*Discriminations on the ground of sex (*texts)
*Gender descrimination (*LIV*)*
*Sex discrimination (*LIV*)*
*Sexism (*Webster, WordNet*)*

This stage is planned to take two years and be finished before 2003. Now the English part of Sociopolitical thesaurus comprises 48 thousand English terms.

Now we began the third stage of the development – revision and correction of collected material.

And the fourth stage is use of the bilingual resource in various applications of automatic text processing, which will lead to further improvement and enrichment of our linguistic resource.

It is important to stress that during analysis of dictionaries our specialists were approved to make Russian-English connections for any Russian words in RuThes (not only from Sociopolitical Thesaurus). Full volume of included English words and expressions is more 62 thousand entries, 67 thousand senses. So this work can be considered as a significant basis for connection to other English structural resources.

8. Use of RuThes in text processing applications

8.1. Use of Sociopolitical thesaurus

Thesaurus on sociopolitical life is used in automatic processing applications since 1996. The Thesaurus is a searching tool in University Information System RUSSIA (UIS RUSSIA, www.cir.ru/eng/), containing more than 600 thousand documents. The text collection of this information system includes such various types of documents as official documents of Russian Federation, legislative acts, international treaties, newspaper articles and statistical reports.

The Sociopolitical thesaurus is used as a linguistic resource in such information retrieval applications as automatic conceptual indexing, knowledge-based text categorization, automatic text summarization (Loukachevitch et. al., 1999). In these applications a thesaurus-based technique of construction of thematic representation of texts is used (Loukachevitch & Dobrov, 2000).

In (Loukachevitch & Dobrov, 2002) we describe an experiment which showed that use of this part of RuThes in information retrieval was much more efficient than retrieval based on vector model. Average precision of document retrieval with the Sociopolitical thesaurus (using its synonyms and hierarchy) was 1.4 times more than average precision of vector retrieval.

8.2. Use of RuThes in text categorization systems

RuThes is currently used as a linguistic resource for knowledge-based text categorization systems.

There are a lot of applications where machine-learning approaches (Joachims, 1998) to text categorization are impossible to use. There can be no sufficient training collection, or a system of categories can include hundreds of hierarchical categories. In these cases a knowledge-

based technique using RuThes can be appropriate (Loukachevitch, 1997). Knowledge described in RuThes substitutes information received from training examples in machine learning approaches.

In our text categorization technique the categories are manually described using Boolean expressions of a relatively small number of 'supporting' concepts. Boolean expressions including all necessary concepts of RuThes are generated on the basis of properties of the Thesaurus relations. The resulted Boolean expressions usually include much more disjunctive and conjunctive components, sometimes in hundreds times more. It became possible owing to detailed presentation of various aspects of described concepts and careful testing of the Thesaurus relations.

One of our last text categorization systems categorizes Russian legislative documents using the system of 1168 categories (3-4 levels of hierarchy), other text categorization system categorizes public opinion polls (almost 400 categories).

Description of categories in large hierarchical systems of categories usually requires large range of lexical knowledge from very specific terminology to very general words. For example, one of categories for categorization of public opinion polls was “Image of woman” and required detailed descriptions of human traits, the list of which was stored in RuThes.

9. Conclusion

In the paper we described main principles of developing Thesaurus of Russian Language RuThes, which is constructed specially as a tool for automatic text processing. The thesaurus contains a lot of multiword expressions, has a specific system of conceptual relations, describing existential properties of concepts, has specific means for lexical disambiguation. We describe current stage of the Thesaurus developing in comparison to 100,000 the most frequent lemmas of the text collection of University Information System RUSSIA, including more than 600 thousand documents. Now thesaurus RuThes is a basis for development the bilingual Russian-English resource for cross lingual text processing. Also we consider the use of the Thesaurus in various applications of automatic text processing.

10. Acknowledgements

Partial support for this research is provided by the Russian Foundation for Humanities through grant # 00-04-00272.

11. References

Apresyan, Yu.D. and Mednikova E.M., 2000. *Noviy Bolshoi anglo-russkiy slovar.* Yu.D. Apresyan and E.M. Mednikova (eds.), Moscow: Russkiy Yazyk. 5th edition. (*in Russian*).

BTS, 1998. *Bolshoi Tolkoviy Slovar Russkogo Yazyka*. S.A. Kuznetsov (ed.), Sankt Peterburg: Norint (*in Russian*).

Climent, S., Rodriguez, H. and Gonzalo, J., 1996. Definitions of the links and subsets for nouns of the EuroWordNet project. - Deliverable D005, WP3.1, EuroWordNet, LE2-4003.

Collins, 1991. *Collins English Dictionary*. HarperCollins. 3rd edition.

EUROVOC, 1995. *Thesaurus EUROVOC*. Vol.1-3, European Communities: Luxemburg: Office for Official Publications of the European Communuties. 3rd edition. English version.

Guarino, N., 1998. Some Ontological Principles for Designing Upper Level Lexical Resources. In *Proceedings of First International Conference on Language Resources and Evaluation*.

Joachims, T., 1998. Text categorization with support vector machine: Learning with many relevant features. In *European Conference on Machine Learning (ECML-98)*, Springer Verlag, 137-142.

LIV, 1994. *Legislative Indexing Vocabulary*. Congressional Research Service. The Library of Congress. Twenty-first Edition.

Longman, 1995. *Longman dictionary of contemporary English*. Harlow (Essex): Longman.

Loukachevitch, N., 1997. Knowledge Representation for Multilingual Text Categorization. In. *AAAI Symposium on Cross-Language Text and Speech Retrieval*, AAAI Technical Report, 133-142.

Loukachevitch, N., and Dobrov, B., 2000. Thesaurus-Based Structural Thematic Summary in Multilingual Information Systems. Machine Translation Review 11: 10-20.

Loukachevitch, N. and Dobrov, B., 2002. Evaluation of Thesaurus on Sociopolitical Life as Information-Retrieval Tool. In *LREC2002 Proceedings.* Las Palmas.

Loukachevitch, N.V., Salii, A.D. and Dobrov, B.V., 1999. Thesaurus for Automatic Indexing: Structure, Developement, Use. In P. Sandrini (ed.), *Proceedings Fifth International Congress on Terminology and Knowledge Engineering*. Vienna: TermNet. 343-355.

Lukashevich, N., 1995. Automated Formation of an Information-Retrieval Thesaurus on the Contemporary Sociopolitical Life of Russia. *Automatic documentation and mathematical linguistics*. 29(2): 29-35.

Miller, G., Beckwith, R., Fellbaum, C., Gross, D., and Miller, K., 1990. Five papers on WordNet, *CSL Report, 43*, Cognitive Science Laboratory, Princeton University.

Multilex, 1996. *Multilex 1.0a. Anglo-russkiy elektronniy slovar.* Medialingua Ltd.

Ozhegov S.I. and Shvedova N.Yu., (1999). *Tolkoviy Slovar Russkogo Yazyka*. Russian Academy of Sciences. Institute of Russian Language. Ì oscow: Azbukovnik. 4th edition. (*in Russian*).

Oxford, 2000. *The New Oxford thesaurus of English*. P. Hanks, (ed.), Oxford. Oxford Univ. Press.

Pustejovsky, J., 1995. *The Generative Lexicon*. Cambridge, Massachusetts, London, England: The MIT Press.

Roget, 1987. *Roget's thesaurus of English words and phrases*. B. Kirkpatrick (ed.), Harlow (Essex): Longman.

UNBIS, 1976. *UNBIS Thesaurus.* English Edition, Dag Hammarskjold Library of United Nations, New York.

Webster, 1999. *Random House Webster's Unabridged Dictionary*. Version 3.0. Random House, Inc.

The Workshop Programme

9:00 - 9:45 Hiroshi Uchida. The UNL: A language for computers
- How to develop a language for computers -. *Invited talk*

9:45 – 10:00 **BREAK**

10:00 – 10:30 Issues in Generating from Interlingua Representations.
Stephan Busemann

10:30 – 11:00 The UNL distinctive features: evidences through a NL-UNL encoding task.
Ronaldo Teixeira Martins, Lúcia Helena Machado Rino

11:00 – 11-15 **COFFEE BREAK**

11:15 – 11:45 Structural and Lexical Transfer: From an UNL graph to an Equivalent NL Dependency Tree. Etienne Blanc, Gilles Sérasset, WangJu Tsai

11:45 – 12:15 Some Lexical Issues of UNL. Igor Boguslavsky.

12:15 – 12:45 A rationale for using UNL as an Interlingua and more in various domains.
Christian Boitet.

12:45 – 13:15 UNL, Challenges and misunderstanding. Some answers.
Jesús Cardeñosa, Edmundo Tovar

13:15 – 13:30 **DEBATE**

Workshop Organisers

Edmundo Tovar. UNL-Spanish Language Centre; Validation and Business Applications Group, Universidad Politécnica de Madrid, Spain. E-mail: etovar@fi.upm.es

Carolina Gallardo. UNL-Spanish Language Centre; Validation and Business Applications Group, Universidad Politécnica de Madrid, Spain. E-mail: carolina@opera.dia.fi.upm.es

Workshop Programme Committee

Jesús Cardeñosa. UNL-Spanish Language Centre; Validation and Business Applications Group Universidad Politécnica de Madrid, Spain.
E-mail: carde@fi.upm.es

Igor Boguslavsky. Institute for Information Transmission Problems, Russian Academy of Sciences, Russia. E-mail: bogus@iitp.ru

Christian Boitet.. Universite' Joseph Fourier, GETA- CLIPS, Grenoble, France.
Email: Christian.Boitet@imag.fr

Irina Prodanof. Institute of Computational Linguistics Consorzio Pisa Riserche-Settore Linguistica, Italy. E-mail: irina@ilc.pi.cnr.it

Table of Contents

Author Index

Issues in Generating Text from Interlingua Representations

Stephan Busemann

DFKI GmbH
Stuhlsatzenhausweg 3
D-66123 Saarbrücken
busemann@dfki.de

Abstract

Multi-lingual generation starts from non-linguistic content representations for generating texts in different languages that are equivalent in meaning. In contrast, cross-lingual generation is based on a language-neutral content representation which is the result of a linguistic analysis process. Non-linguistic representations do not reflect the structure of the text. Quite differently, language-neutral representations express functor-argument relationships and other semantic properties found by the underlying analysis process. These differences imply diverse generation tasks. In this contribution, we relate multi-lingual to cross-lingual generation and discuss emergent problems for the definition of an interlingua.

1. Introduction

In this contribution, we relate multi-lingual to cross-lingual generation and discuss emerging problems for the definition of an interlingua. Multi-lingual generation starts from non-linguistic content representations for generating texts in different languages that are equivalent in meaning. The generation of weather forecasts or environmental reports are typical examples. In contrast, cross-lingual generation is based on a language-neutral content representation which is the result of a linguistic analysis process. Generation for machine translation is a most prominent example.

Non-linguistic representations do not specify linguistic semantics nor do they reflect the structure of the text to be generated. In contrast, language-neutral representations express functor-argument relationships and other semantic properties found by the underlying analysis process. These differences imply diverse generation tasks.

However, there are also commonalities. In both cases, generation is the mapping of some semantic representation onto linguistic strings. We may assume a single generation process that uses different separately defined language-specific knowledge sources. In both cases, we may view the underlying representation as an interlingua, since it attempts to cross the language barrier by providing content descriptions independently of the target language.

An instance of each type of tasks has been implemented using the generation system TG/2 (Busemann, 1996), quickly overviewed in Section 2.. The usage of the same framework allows us to relate the tasks to each other (Section 3.) and to gain insights relevant to a coherent definition of interlinguas, generation tasks, and generation knowledge (Section 4.).

2. TG/2 in a Nutshell

TG/2 is a flexible production system that provides a generic interpreter to a set of user-defined condition-action rules representing the generation grammar. The generic task is to map an input structure onto a chain of terminal elements as prescribed by the rule set. The rules have a context-free categorial backbone used for standard top-down derivation, which is guided by the input representation. The rules specify conditions on input ("tests") determining their applicability and allow navigation within the input structure ("access functions").

The right-hand side of a rule can consist of any mixture of terminal elements (canned text) or other categories associated with an access function. The presence of canned text is useful if the input does not express explicitly everything that should be generated. With very detailed input, the terminal elements of the grammar will usually be words.

Given a category C and some (piece of) input structure I, production rules are applied through the standard three-step processing cycle:

1. Identify the applicable rules;
2. Select a rule on the basis of some (freely programmable) conflict resolution mechanism; and
3. Apply that rule.

A rule is applicable if its left-hand side category is C and its tests hold on I. A rule is applied by processing its right-hand side elements from left to right. Canned text is output right away, and non-terminal elements induce a new cycle with the new category and the return value of the access function. Processing terminates when all right-hand side elements have been realized successfully. In the case of a failure, processing backtracks to step 2. If no more rules are applicable, a global failure occurs. For details see (Busemann, 1996).

3. Relating Two Distinct Generation Tasks

TG/2 has been used in a variety of NLG tasks. We look at multi-lingual report generation and cross-lingual summarization. We then locate the tasks on a scale ranging from shallow to in-depth generation, and discuss advantages and drawbacks of these locations.

3.1. Task 1: Generating air quality reports from measurement data

Reports about air quality in a German-French border region (Busemann and Horacek, 1998) are currently

```
[(COOP THRESHOLD-PASSING)
 (LANGUAGE ENGLISH)
 (TIME [(PRED SEASON)
        (NAME [(SEASON WINTER)
               (YEAR 2001)])])
 (POLLUTANT SO2)
 (SITE "Saarbruecken-City")
 (SOURCE [(THRESHOLD-TYPE MIK-WERT)])
 (EXCEEDS [(STATUS YES) (TIMES 1)])]
```

Figure 1: A Non-Linguistic Input Expression for Report Generation: "In Winter 2001 at the measuring station at Saarbrücken-City, the MIK value for sulfur dioxide was exceeded once."

```
(defproduction site "S01"
  (:PRECOND
     (:CAT SITE-E
      :TEST ((always-true)))
   :ACTIONS
     (:TEMPLATE
        "at the measuring station at "
        (:RULE SITE-NAME-E (self)))))
```

Figure 2: Making Implicit Meaning Explicit: A TG/2 grammar rule. The rule is "unconditioned" and uses the current piece of input structure to access the site name.

produced in six languages (a web demo is available at `http://www.dfki.de/service/nlg-demo`). The reports are based on real measurement data taken from a database and on the user's parameters determining the type of the report (time series, average or maximum value description, threshold passing description). A report consists of up to six statements most of which are verbalized by TG/2. The initial text organization stage retrieves the relevant data, decides about the content of the statements and defines their order. For each statement to be verbalized by TG/2 it produces a domain-oriented non-linguistic intermediate feature structure serving as input to TG/2 (cf. Figure 1 for an example). Input expressions for TG/2 may specify e.g. the pollutant, the actual measurements, and their date and location. Moreover, further information is specified according to the user's choice of parameters. It should be noted that some input is just carried forward from the original system input (in Figure 1, this is `LANGUAGE, TIME, POLLUTANT, SITE, THRESHOLD-TYPE`), whereas other information originates from the DB query and text organization stage (`COOP` and `EXCEEDS` in Figure 1).

The text organization stage is entirely content-oriented, and the intermediate feature structures do not exhibit linguistic properties. The 'language' feature causes the selection of the rule set for the language requested. The determination of linguistic structure for each input expression is achieved by the TG/2 grammar rules. Since implicit information is associated with some parts of input expressions, canned text is used to make it explicit at the surface. An example in Figure 1 is the added notion of "at the measuring station at" in the case of `(SITE "Saarbrücken-City")`, which is verbalized through the rule in Figure 2.

The grammars comprise about 100-120 rules for each language and are specifically designed for this application. The development of a grammar for another language takes between one and three weeks depending on skills.

3.2. Task 2: Generating medical scientific text for summaries

This generation task occurred in the context of the cross-lingual text summarization system MUSI (Lenci et al., 2002). MUSI involves a combination of analysis and generation similar to machine translation. An interlingua approach was chosen to represent selected English and Italian medical scientific sentences in a language-neutral way. The sentences can be complex and quite long (50 words are no exception). Interlingua expressions were fed to sentence generation components producing the elements of a French or German summary.

The generation of German sentences (Busemann, 2002) starts from so-called IRep4 interlingua expressions. A sample IRep4 expression is shown in Figure 3. IRep4 expressions are hierarchical predicate-argument structures complemented by a rich variety of features and modifiers. The basic elements are atomic and predicative concepts, forming an ontology shared across the MUSI system. In particular, predicative frames are based on the SIMPLE formal specifications (Lenci et al., 2000). IRep4 expressions are composed of `PROP` and `ITEM` elements used to represent propositions and terms, respectively. Although IRep4 is in principle a semantic representation language, its expressions also keep track of some syntactic properties of the source language elements. For instance, number and determiner information is specified for NPs as well as categorial information for propositions (`CAT`). This information can be very useful in guiding text generators.

IRep4 is suitable for representing the semantics of very complex sentences, but at the same time, it leaves room for various degrees of specification. In fact, co-reference resolution, attachment ambiguities and the incorrect identification of arguments and modifiers are common sentence analysis problems that may lead to incomplete output. To cope with these problems, IRep4 has been designed to integrate possibly underspecified or fragmentary representations. This feature greatly enhances the robustness of the system and can guarantee a better interface with the text analysis component.

A direct interpretation of IRep4 by TG/2 would require choosing the lexemes and the syntactic realizations. This could have been achieved within the TG/2 grammar through complicated tests. These choices partly depend on each other, which would have caused massive backtracking. Moreover, testing the presence of a concept in IRep4 would have been triggered by rules expanding the syntactic category of the lexemes (part of speech), e.g. the rule `Noun` $\rightarrow$ `"acetylcholin"` would have been associated with a test whether the current concept was `C_acetylcholine`. As there would have been hundreds of these, concerns of processing efficiency were in order. Finally, a pre-existing grammar should be reused that was not previously adapted

```
PROP{ Value = P_ARG1_cause_ARG2;
      Time_Rep = [PRESENT, PRES_USUAL];
      Cat = V_SEN;
      Arg1 = PROP{ Value = P_antagonism_with_ARG1;
                   Cat = NP; Det = INDEF;
                   Arg1 = ITEM{ Value = C_acetylcholine;
                                Mod1 = [LOC, ITEM{
                                        Value = C_level;
                                        Det = DEF;
                                        Mod1 = [RESTR, ITEM{
                                                Value = C_sight;
                                                Number = PLUR; Det = DEF;
                                                Mod1 = [RESTR, C_muscarinic];
                                                Mod2 = [RESTR, ITEM{
                                                        Value = C_substance;
                                                        Number = PLUR;
                                                        Det = DEMONST1;}]; }]; }]; };
                   Mod1 = [RESTR, C_competitive]; };
      Arg2 = ITEM{ Value = C_effect;
                   Det = DEF; Number = PLUR; }; }
```

Figure 3: IRep4 Expression for "Die Wirkungen werden durch einen kompetitiven Antagonismus zu Acetylcholin auf dem Niveau der muskarinischen Bindungsstellen dieser Substanzen verursacht." [The effects are caused by a competitive antagonism with acetylcholine on the level of the muscarinic sights of these substances.].

to IRep4.

For these reasons it appeared more convenient to introduce an initial sentence planning stage. The resulting representation – see Figure 4 for an example corresponding to Figure 3 – forms the input to TG/2. It can be viewed as a syntactically enriched, language-specific paraphrase of the underlying IRep4 expression. It represents explicitly the linguistic structure of the sentence. The TG/2 grammar is responsible for word order and inflection. Very much like in a classical sentence realization system, no canned text parts are used. If a phrase like "at the measuring station at" had to be generated here, an underlying interlingual semantic expression would be mandatory.

A pre-existing TG/2 grammar for German syntax was reused and adapted to the needs of MUSI (Busemann, 2002; Lenci et al., 2002). Its final version comprises over 950 rules.

3.3. Shallow and in-depth generation

The notion of shallow generation, as opposed to in-depth generation, has been coined by (Busemann and Horacek, 1998) to describe a distinction corresponding to that of shallow and deep analysis. In language understanding deep analysis attempts to "understand" every part of the input, while shallow analysis tries to identify only parts of interest for a particular application, omitting others. In-depth generation is inherently knowledge-based and theoretically motivated, whereas shallow generation quite opportunistically models only the parts of interest for the application in hand. Often such models will turn out to be extremely shallow and simple, but in other cases much more detail is required. Thus, techniques such as those developed within TG/2 for varying modeling granularity according to the requirements posed by the application are a prerequisite for reusing NLG systems.

Obviously a shallow NLG system is, in general, based on representations that carry implicit meaning. We call this shallow input. Additional text has to be "invented" by the generator (in TG/2, this is usually achieved using canned text in the grammar).[1] This leads to domain-dependent, shallow grammars that cannot be reused easily for another task. The in-depth models assume a very fine-grained grammar describing all the linguistic distinctions covered by the interlingua. Such a grammar corresponds closely to familiar generic linguistic resources.

The report generation task described was solved by a typical shallow approach, whereas the MUSI generation task required an in-depth model.

The tension between shallow and in-depth generation has been discussed further in the literature. According to Reiter and Mellish, shallow techniques (which they call "intermediate") are appropriate as long as corresponding in-depth approaches are poorly understood, less efficient, or more costly to develop (Reiter and Mellish, 1993). Bateman and Henschel describe ways of compiling specialized grammars out of general resources (Bateman and Henschel, 1999). A platform for generating, storing and reusing representations is described in (Calder et al., 1999), showing that such reuse can be seen as a shallow methodology to text generation. A major conclusion seems that there is no dichotomy between both approaches, but that shallow systems can indeed be based on theoretically sound in-depth models.

In practice though, NLG tasks turn out to be highly diverse, and no NLG system could be reused for a new application off the shelf. The necessary effort for adaptation and extension of large existing in-depth resources such as KPML (Bateman, 1997) or FUF/Surge (Elhadad and Robin, 1996) is often considered high. In fact, the de-

[1] Of course, these texts are defined by the application, viz. the customer, as all other output.

```
[(SENTENCE DECL)
 (VC [(VOICE PASSIV)
      (MOOD IND)
      (TENSE PRAESENS)
      (SBP S2)
      (STEM "verursach")])
 (DEEP-SUBJ [(TOP Y)
             (TY GENERIC-NP)
             (NUMBER SG)
             (DET INDEF)
             (NR V2)
             (GENDER MAS)
             (STEM "antagonismus")
             (PP-ATR [(LOCATIVE ...)
                      (GENDER NTR)
                      (STEM "Acetylcholin")
                      (DET WITHOUT)
                      (NUMBER SG)
                      (TY GENERIC-NP)
                      (PREP MIT)])
              (ADJ [(STEM "kompetitiv")
                    (POS ADJECTIVE)
                    (DEG POS)])])
 (DEEP-AKK-OBJ [(TY GENERIC-NP)
                (NUMBER PLUR)
                (DET DEF)
                (STEM "wirkung")
                (GENDER FEM)])]
```

Figure 4: TG/2 Input Expression Partly Corresponding to Figure 3. The material for "on the level of the muscarinic sights of these substances" would appear under `DEEP-SUBJ.PP-ATR.LOCATIVE`, but has been omitted for reasons of space. The representation contains content word stems and names for syntactic structures (`SBP, NR` features). Determiners and prepositions are also provided.

velopment from scratch of a shallow grammar for a small NLG application on the basis of a simple framework like TG/2 can be more cost-effective.

Shallow and in-depth generation tasks can be related with help of TG/2. As the amount of domain-specific canned text in the TG/2 grammars correlates to the shallowness of the input, the generation tasks described can be located on a scale that ranges from shallow to in-depth domain and input models. There are trivial systems at one end that just produce canned text according to triggers (e.g. system error reports). A bit further on the scale we find template-style systems, like the air quality report generator, which use canned text to make knowledge implicit in the input explicit. In-depth realizers with sophisticated grammars that do not use domain-specific canned text at all are located at the other end of the scale, such as the MUSI generator.

Why are shallow and in-depth interlinguas both viable? One obvious reason lies in the origin of the interlingua representations. Shallow representations usually originate from non-linguistic processing, such as accessing a database or interpreting some user interaction, whereas in-depth representations generally have a linguistic origin, e.g. from an NL parsing component.

More interestingly, the type of domain and application determines the depth of modeling. Air quality reports form a small and closed domain. Implicit knowledge is easy to make explicit. A shallow model, being inherently simple, is perfectly adequate. A complex functor-argument representation would mean a dramatic overshot for this type of application. The same holds for many generation applications, such as reporting about stock exchange (Kukich, 1983) or weather forecasts (Boubeau et al., 1990). Medical scientific texts, on the other hand, form a very large domain, requiring broad-coverage linguistic knowledge. A shallow model would not even be able to capture the most frequent semantic relations. General means of expressing semantic relationships are mandatory.

What are the advantages and drawbacks of either approach? Shallow interlinguas allow for a straightforward multi-lingual generation. All linguistic processing can be concentrated in the module consuming the interlingua expression, e.g. TG/2. A drawback consists in domain-dependent grammars, which are hardly reusable for other applications. Still it is worthwhile, as the effort to create a grammar for another language is low.

With in-depth language-neutral representations, the issue of reusing existing linguistically motivated grammars arises, simply because of the tremendous effort for developing them from scratch. Technically an existing grammar may be reused if a well-defined interface is available. In TG/2, the interface to the input representations consists of the tests and access functions called from within the grammar rules. Depending on the different organization of information within input languages, this interface must be modified. If the same types of information required by the grammar can be produced by the new input language, the way is paved for a successful reuse. If the new input language offers different types of information, the adaptation problem described above arises.

4. On the Definition of Interlinguas

We now address issues on the semantics and pragmatics of interlinguas from a generation perspective by discussing three types of problems generators may encounter with in-depth interlinguas, using experiences with IRep4 as our source of examples.[2]

4.1. Extrinsic problems

In MUSI, a variety of problems with interlinguas known from machine translation were experienced, showing that this interlingua, as so many others, is not language-neutral in a strict sense. The problems were related to the fact that languages encode information differently and the interlingua cannot sufficiently abstract away from this. More precisely, although IRep4 does not contain elements specific to any of the four languages involved, the analysis results reflected some grouping and nesting of phrases and clauses of the source language.

[2] By critically reviewing IRep4, we necessarily omit mentioning many excellent features that made it very useful for the challenging task of representing scientific text.

For instance, Italian (and English) uses post-nominal adjectival clauses that correspond to a post-nominal relative clause or pre-nominal adjectival modifiers in German (cf. Figure 5a). German does not have the possibility to linearize or nest several adjectival or participial clauses after the head noun. Moreover, large phrases in pre-nominal position are difficult to understand since the head noun is uttered only afterwards.

In IRep4, these clauses are typically represented as restrictive modifiers (RESTR), accompanied, in the case of a predicative concept, by the source-language specification CAT = ADJP. The generator follows the heuristic strategy of assigning small adjectival phrases to the pre-nominal adjective position and large ones to the post-nominal relative clause position. In the latter case, the CAT specification will be ignored, as a full sentence with a copula must be generated. A further requirement consists of the need for one argument of the adjective to be realizable as the relative pronoun.

The result is not satisfactory, as it can lead to recursive center-embedding causing bad readability (cf. Figure 5b). The sentence in Figure 5c is stylistically much better; it has fewer closing brackets in a sequence, which means less deep embedding and improved readability. Linguistically, it shows two extrapositions, i.e. the innermost relative clause (not bracketed further) occupies the post-field[3] of the embedding one, which in turn occupies the post-field of the main clause. The stylistically preferred solution would be to realize the innermost clause as a prenominal AP, while extraposing the larger clause as a relative clause, as in Figure 5d.

Another striking example of language differences experienced with IRep4 is the use of determiners. English text does not use always definite articles when they are mandatory in German. For instance, "features of malnutrition" should be translated into "Merkmale der Mangelernährung" (definite article included), whereas "features of chronic malnutrition" corresponds to "Merkmale chronischer Mangelernährung" (no article).

IRep4 does, of course, not represent definite articles when there are no such determiners in the source-language text. The generator uses as a general rule that "naked" generalized possessives – i.e. the head of a RESTRictive modifier that corresponds to a noun and does not have a determiner or a modifier – are automatically accompanied by a definite article, covering the above examples.

English "Treatment consisted in..." should translate to "Die Behandlung bestand aus...", using a definite article. In these cases, a decision within the generator on whether or not to use a definite article would rely on lexical semantic information about both the source and target language lexemes.

The obvious solution to the extrinsic problems is to complement the level of interlingua with a set of transfer rules specific for every pair of source and target language. This complicates the situation, but would, in MUSI, have led to considerable stylistic improvements of the generated sentences.

For shallow models, this problem simply does not exist.

4.2. Intrinsic problems

IRep4 also has a few intrinsic properties that affected generation. Most prominently, it does not represent scope and thematic, or constituent, order information. The scope of negation would be important for the proper placement of the negation particle. Moreover, the scope of modifiers is not represented. With the current, inherently flat representation, i.e. multiple modifiers at the same level of embedding, generation cannot decide between e.g. "the following clinical case" and "the clinical following case". Modifiers should be nested to express this information.

Deciding about word order in generation is relevant to represent the argumentative structure in complex sentences and ensure coherence. The order of constituents in the source language text is not marked in IRep4, which may cause a deviating target-language order in German. This can lead to a lack of textual coherence, if e.g. a modifier that starts the sentence appears at the end. Consider "upon objective investigation, the woman's face was red and congested", which was translated into "das Gesicht der Frau war rot und geschwollen bei objektiver Untersuchung", generating the introductory PP at the end. A possible subsequent anaphoric reference would be less felicitous than in the original text. In the absence of a super-ordinated text planning stage, interlingua expressions should specify thematic order, or constituent order, in the source language text.

German generation assumes a standard word order for active voice, unless other information is given. The standard word order does not take into consideration the complexity, or the "weight", of a constituent. A heavy-weight subject preceding a short object in a transitive sentence is often considered bad style. Based on heuristics about a constituent's "weight", passive voice could have been chosen within the generator, causing the short constituent to precede the complex one, which generally leads to more fluent text (cf. the example in Figure 3). An interlingua should include hooks to provide this information. IRep4 might indirectly allow a good estimate by counting concepts, arguments and modifiers; further investigation is needed to identify a reliable formula.

For shallow interlinguas, intrinsic problems of this kind do not exist, as they are entirely dealt with in the grammar.

4.3. Pragmatic problems

In this section, we sketch some issues that can take a lot of effort to create a shared understanding among the researchers looking at interlingua expressions from different perspectives.

A grammatically correct input sentence is a legitimate input to a parser. Few systems can deal with incorrect sentences in an error-tolerant way. For generation, in-depth interlingua expressions should be correct in a similar sense. A formal specification of the interlingua is required to define its syntax and, very importantly, its semantics. Generation requirements should be formally specified as well and

[3] The post-field follows the infinite verb complex in a German declarative sentence. This position can be occupied by one constituent.

a) [[In the clinical case described,] [the symptoms] [were] [caused] [by ingestion [of anticolinergic substances [probably contained [in the leaves [of plants [consumed a few hours before]]]]]]].

b) [[In dem beschriebenen klinischen Fall] [wurden] [die Symptome] [durch [Verzehr [von anticholinergen Substanzen, [[die] [die Blätter [der Pflanze], [die vor ein paar Stunden genossen wurden,] möglicherweise enthielten,]]]]] [verursacht]].

In the described clinical case were the symptoms by ingestion of anticolinergic substances, that-were in-the leaves of-the plants, that-were a few hours before consumed, possibly contained.

c) [[In dem beschriebenen klinischen Fall] [wurden] [die Symptome] [durch Verzehr [von anticholinergen Substanzen]] [verursacht], [[die] [die Blätter [der Pflanze]] möglicherweise enthielten, [die vor ein paar Stunden genossen wurden]]].

d) [[In dem beschriebenen klinischen Fall] [wurden] [die Symptome] [durch Verzehr [von anticholinergen Substanzen]] [verursacht], [[die] [die [vor ein paar Stunden genossenen] Blätter [der Pflanze]] möglicherweise enthielten]].

Figure 5: Stylistic Variations in Translation. Brackets indicate some syntactic structure. a) English original sentence; b) Corresponding sentence in German with APs realized as relative clauses, with inter-linear translation; c) Extraposition of the relative clauses beyond the respective verbs; d) Realization of the innermost clause as a prenominal AP.

should be part of the "pragmatics" of the interlingua. For instance,

- the omission of information about tense, aspect, determination and number may mean that a default applies;
- a personal pronoun must either refer to an antecedent, or be accompanied by information about gender, person and number;
- an expression realized as a relative clause must contain exactly one constituent with a plain coreference specification; this constituent will become the relative pronoun;
- etc.

During the development of IRep4, this effort was not spent due to shortage of resources.[4] While from an analysis viewpoint, some decent output looks more or less satisfactory, it is the details that make generation feasible or cause its failure. Most importantly, the interpretation of interlingua expressions in NLG should be functional. Different surface representations corresponding to the same interlingua expression should be considered as equivalent in meaning. If this fundamental principle is not maintained, translation is not guaranteed to be meaning-preserving.

An interlingua can support this principle by making meaning representation explicit. IRep4 unfortunately has a fairly abstract representation for PP adjuncts and modifiers. The scheme is "`Mod = [<name>, <Irep4-expression>]`", where `<name>` is taken from a finite set of strings that more or less denote the semantics of the modifier. These names can be interpreted unambiguously by generation, but analysis may encounter difficulties in relating prepositions and head nouns to them, if only little lexical semantic knowledge is available. In Figure 3, the same name `RESTR` is realized differently, depending on the part of speech used for the embedded concept. If it is a noun, the semantics is that of a generalized possessive, which is realized in post-nominal position in German. If it is an adjective, a prenominal adjectival modifier is usually generated. Other uses of `RESTR` were mentioned above. If two or more meanings are connected to one name, it may appear psychologically difficult to refrain from using this name as a waste-basket.

Pragmatic problems exist for shallow models as well, as shallow input expressions are partly produced by external systems. In the air quality report generator, measuring values are received as input from a database. Time series are occasionally shortened by aggregating information ("from 9.00 to 11.00: 6,7 μg/m^3"). During the development, we have not been aware of the systematic omission of certain half hour values in the database, which occasionally leads to awkward results: "at 9.00: 6,7 μg/m^3; at 9.30: 0 μg/m^3; at 10.00: 6,7 μg/m^3; at 10.30: 0 μg/m^3; at 11.00: 6,7 μg/m^3". We easily could have implemented another aggregation rule that leads to output like "from 9.00 to 11.00: 6,7 μg/m^3, with every half hour value at 0".

5. Conclusion

In this contribution, we have related multi-lingual to cross-lingual generation and discussed emerging problems for the definition of an interlingua. This discussion was based on experience gained from implementing NLG components for a multi-lingual report generator and a cross-lingual summarization system within the same framework, TG/2. Shallow interlinguas originate from non-linguistic processing. They usually carry implicit meaning that must be made explicit in the generation process. For relatively small-coverage, closed domains, such as air quality reports, weather reports, or stock market reports, it is adequate to write specialized grammars using domain-specific canned text for this purpose. In-depth interlinguas usually originate from linguistic analysis, as in machine translation. The nature of the interlingua is closely tied to the sophistication of

[4] It is debatable though whether the resulting difficulties have been resolved with less effort.

the generation task in hand.

While well-modularized generation systems can be easily adapted to shallow interlinguas, an in-depth interlingua is much more complex to work with, as so many distinctions need to be addressed. In this paper we have identified some NLG requirements on in-depth interlinguas. From the experience with the MUSI application, we have learned that it is worthwhile to formally specify NLG requirements on the interlingua at the outset.

For a new application involving multi-lingual or cross-lingual generation, the interlingua should be chosen, adapted or designed according to the kind of linguistic processing involved and in view of the depth of modeling envisaged. On the shallow/in-depth scale, it should be as shallow as possible.

6. References

John Bateman and Renate Henschel. 1999. From full generation to 'near-templates' without loosing generality. In (Becker and Busemann, 1999), pages 13–18. Also available at `http://www.dfki.de/service/NLG/KI99.html`.

John Bateman. 1997. KPML delvelopment environment: multilingual linguistic resource development and sentence generation. Report, German National Center for Information Technology (GMD), Institute for integrated publication and information systems (IPSI), Darmstadt, Germany, January. Release 1.1.

Tilman Becker and Stephan Busemann, editors. 1999. *May I Speak Freely? Between Templates and Free Choice in Natural Language Generation. Workshop at the 23rd German Annual Conference for Artificial Intelligence (KI '99). Proceedings*, Document D-99-01. Also available at `http://www.dfki.de/service/NLG/KI99.html`.

L. Boubeau, D. Carcagno, E. Goldberg, Richard Kittredge, and A. Polguére. 1990. Bilingual generation of weather forecasts in an operations environment. In *Proceedings of the 13 th International Conference on Computational Linguistics (COLING-90), Volume 1*, pages 90–92, Helsinki.

Stephan Busemann and Helmut Horacek. 1998. A flexible shallow approach to text generation. In Eduard Hovy, editor, *Nineth International Natural Language Generation Workshop. Proceedings*, pages 238–247, Niagara-on-the-Lake, Canada. Also available at `http://xxx.lanl.gov/abs/cs.CL/9812018`.

Stephan Busemann. 1996. Best-first surface realization. In Donia Scott, editor, *Eighth International Natural Language Generation Workshop. Proceedings*, pages 101–110, Herstmonceux, Univ. of Brighton, England. Also available at the Computation and Language Archive at `http://xxx.lanl.gov/abs/cmp-lg/9605010`.

Stephan Busemann. 2002. Language generation for cross-lingual document summarisation. In Huanye Sheng, editor, *International Workshop on Innovative Language Technology and Chinese Information Processing (ILT&CIP-2001), April 6-7, 2001, Shanghai, China*, Beijing, China, May. Science Press, Chinese Academy of Sciences.

Jo Calder, Roger Evans, Chris Mellish, and Mike Reape. 1999. "free choice" and templates: how to geth both at the same time. In (Becker and Busemann, 1999), pages 19–24. Also available at `http://www.dfki.de/service/NLG/KI99.html`.

Michael Elhadad and Jacques Robin. 1996. An overview of SURGE: a reusable comprehensive syntactic realization component. In Donia Scott, editor, *Eighth International Natural Language Generation Workshop. Demonstrations and Posters*, pages 1–4, Herstmonceux, Univ. of Brighton, England.

Karen Kukich. 1983. Design and implementation of a knowledge-based report generator. In *Proceedings of the 21st Annual Meeting of the Association for Computational Linguistics*, pages 145–150, Cambridge, MA.

Alessandro Lenci, Nuria Bel, F. Busa, Nicoletta Calzolari, E. Gola, M. Monachini, Alexandre. Ogonowsky, I. Peters, W. Peters, N. Ruimy, M. Villegas, and Antonio Zampolli. 2000. SIMPLE: a general framework for the development of multilingual lexicons. *International Journal of Lexicography*, 13(4):249–263.

Alessandro Lenci, Ana Água, Roberto Bartolini, Stephan Busemann, Nicoletta Calzolari, Emmanuel Cartier, Karine Chevreau, and José Coch. 2002. Multilingual summarization by integrating linguistic resources in the MLIS-MUSI project. In *Procs. Third International Conference on Language Resources and Evaluation (LREC)*, Las Palmas, Canary Islands, Spain, May.

Ehud Reiter and Chris Mellish. 1993. Optimizing the costs and benefits of natural language generation. In *Proc. 13th International Joint Conference on Artificial Intelligence*, pages 1164–1169, Chambery, France.

The UNL Distinctive Features: Inferences from a NL-UNL Enconverting Task

Ronaldo Teixeira Martins[*], Lúcia Helena Machado Rino[**], Maria das Graças Volpe Nunes[***], Osvaldo Novais Oliveira Jr.[****]

[*]Núcleo Interinstitucional de Lingüística Computacional - NILC
Av. do Trabalhador São-Carlense, 400 - 13560-970 - São Carlos, SP, Brazil
ronaldo@nilc.icmc.sc.usp.br
[**]Departamento de Computação - Centro de Ciências Exatas e de Tecnologia - UFSCar
Rod. Washington Luiz, km 235 - Monjolinho - 13565-905 - São Carlos, SP, Brazil
lucia@dc.ufscar.br
[***]Instituto de Ciências Matemáticas e da Computação (ICMC) - Universidade de São Paulo
Av. do Trabalhador São-Carlense, 400 - 13560-970 - São Carlos, SP, Brazil
mdgvnune@icmc.sc.usp.br
[****]Instituto de Física de São Carlos (IFSC) - Universidade de São Paulo
Av. do Trabalhador São-Carlense, 400 - 13560-970 - São Carlos, SP, Brazil
chu@ifsc.sc.usp.br

Abstract

This paper reports on the distinctive features of the Universal Networking Language (UNL). We claim that although UNL expressions are supposed to be unambiguous, UNL itself is able to convey vagueness and indeterminacy, as it allows for flexibility in enconverting. The use of UNL as a pivot language in interlingua-based MT systems is also addressed.

1. Introduction

Machine Translation (MT) is one of the most controversial subjects in the field of natural language processing. Researchers and developers are often at odds on issues concerning MT systems approaches, methods, strategies, scope, and their potentialities. Dissent has not hindered, however, the establishment of tacit protocols and core beliefs in the area. It has often been claimed that[1]: 1) fully automatic high-quality translation of arbitrary texts is not a realistic goal for the near future; 2) the need of some human intervention in pre-edition of the input text or in post-edition of the output text is mandatory; 3) source language should be rather a sublanguage, and the input text should be domain- and genre-bounded, so that the MT system could cope with natural language ambiguity; 4) the transfer approach is more feasible than the interlingual one, since the latter, albeit more robust and economic, is committed to the somewhat insurmountable task of designing a perfect (universal) language, comprising any other one; 5) common sense and general knowledge on both the source and the target cultures are as important as linguistic information, like in Knowledge-Based Machine Translation Systems (Nirenburg et al., 1992); 6) existing human translations can be used as a prime source of information for the production of new ones, similarly to the Example-Based Machine Translation Systems (Furuse and Iida, 1992); 7) existing MT systems are not appropriate to monolingual users, although they can be used to facilitate, speed up or reduce the costs of human translation, or to produce quick and cheap rough translations that may help the users to get a very broad idea of the general subject of the text.

[1] Most of these assumptions can be extracted from the Survey on the State of the Art in Human Language Technology (Cole et al., 1995). Of special interest are the articles concerning multilinguality by Martin Kay (8.1, 8.2) and Christian Boitet (8.3, 8.4).

Many authors obviously do not endorse all the listed statements, specially the fourth one. Hozumi Tanaka (1993), for example, argues in favor of the interlingua-based approach, and so do the research and development groups involved in interlingua-based systems, such as ULTRA (Farwell and Wilks, 1993), KANT (Mitamura et al., 1993), or PIVOT (Okumura et al., 1993). These works, however, rather confirm the very general observation that commercially available MT systems (e.g., SYSTRAN, VERBMOBIL, DUET (Sharp), ATLAS I (Fujitsu), LMT (IBM), METAL (Siemens)) are primarily transfer-based.

The most serious arguments against the interlingua approach concerns its alleged universality and excessive abstractness (Hutchins and Somers 1992). In order to cope with multilinguality, the interlingua should put aside language-dependent structures (such as the phonological, morphological, syntactical and lexical ones) and work at the logical level, which is supposed to be shared by human beings. Even at such uppermost level, however, there seems to be cultural differences. Eco (1994) reports, for instance, the case for Aymara, a South-American Indian language which would have three truth values, instead of the two "normal" ones. Furthermore, it has been said that, even if one comes to find this kind of perfect language, it would be so abstract that it would not be cost-effective, since the tools for departing from natural language and arriving at the logical representation would be excessively complex.

In what follows, we present some extra evidence towards the feasibility of interlingua-based MT. The Universal Networking Language (hereafter, UNL), developed by Uchida et al. (1999), brings some distinctive features that may lead to overcome some of the bottlenecks frequently associated to the interlingua approach. Although UNL was not designed as an interlingua, and MT is only one of the possible uses for UNL, it has been claimed that multilingual MT systems can use UNL as a pivot language. In this paper, some of the distinctive features of UNL are analyzed. We build

upon the experience in developing the Brazilian Portuguese (hereafter, BP) UNL Server, a bilingual MT system for translating Portuguese into UNL and vice-versa.

This paper is organized as follows. Section 2 provides a brief introduction to the UNL approach and some of its premises. In Section 3 we describe an experiment in which human subjects were asked to enconvert sentences from Portuguese into UNL. Section 4 brings the general results of the experiment. One of them is specially addressed in Section 5. Some issues arising from the results are presented in Section 6. Conclusions are stated in Section 7. The reader is supposed to have previous information on the UNL Project and knowledge on UNL Specification (at http://wwww.unl.ias.unu.edu) is considered mandatory.

2. The Universal Networking Language

The Universal Networking Language (UNL) is "an electronic language for computers to express and exchange every kind of information" (Uchida et. al., 1999, p. 13). According to the UNL authors, information conveyed by each natural language (NL) sentence can be represented as a hyper-graph whose nodes represent concepts and whose arcs represent relations between concepts. These concepts (called Universal Words or simply UWs) can also be annotated by attributes to provide further information on the circumstances under which they are used.

In this context, UNL is not different from the other formal languages devised to represent NL sentence meaning. Its structure is said to suffice to express any of the many possible meanings conveyed by any sentence written in any NL. This does not mean, however, that it is able to represent, at the same time, all the possible meanings conveyed by the very same NL sentence. Instead, UNL is able to represent each of them independently, and it is by no means able to provide a single structure coping with all of them. In this sense, there will never be a single UNL expression that completely suffices the meaning correspondence to a NL sentence. Or else: no UNL expression will be ever completely equivalent to a NL sentence, since the latter, but not the former, will allow for ambiguity.

In the following section, we report on results of a BP-UNL enconverting task that has been carried out by BP native speakers. In this experiment, we observe evidences that BP sentences must be disambiguated in order to be represented as UNL expressions.

3. The Experiment

In August 2001, we carried out an experiment on BP-UNL enconverting that involved 31 BP native speakers, all of them graduate and postgraduate students. Most of them (over 95%) were Computer Sciences students, aging 21 to 42 years old (90% of them were under 30 years old).

The experiment was split into training (steps 1-4) and test sessions (step 5), as follows: 1) a very general description of the UNL structure; 2) a general presentation of the definitions provided for five relation labels by the UNL Specification (1999), namely, 'agt' (agent), 'cag' (co-agent), 'obj' (affected thing), 'cob' (affected co-thing), and 'ptn' (partner); 3) an individual exercise on the use of the presented relation labels, in which subjects were asked to identify 50 different relations appearing in different BP sentences, indicating the corresponding UNL relation labels; 4) a public discussion on the exercise results; and 5) a final individual test in which subjects were asked again to identify 30 different relations appearing in different BP sentences, through their correspondence with the very same set of UNL relation labels. In Step 3 and 5, the subjects had also the option of pinpointing the impossibility of identifying either a relationship or its corresponding relation label, by choosing a "catch all" alternative (see option (a) in Figure 1). This exercise aimed at providing the means for the subjects to understand and explore BP-UNL enconverting, concerning the relation labels identification. This was then reinforced in Step 4, which was supervised by a UNL specialist. As it can be observed, these steps aimed at Step 5, the actual BP-UNL assignment, focusing on specific relation labels. In this step, some of the BP sentences presented to the subjects in Step 3 have been replicated.

Altogether, this experiment has taken 1 hour and 40 minutes, considering a 20-minute interval between the training and test sessions. Steps 1 and 2 have last 20 minutes, and so has Step 3 alone. Step 4, the longest one, has taken 40 minutes. Step 5, the actual test, has taken another 20 minutes. The interval between training and test aimed at allowing for the subjects settling on UNL specification, since test has been totally unsupervised. This also justifies our replication of some of the BP sentences used in training.

An English version of the task proposed in Step 3 is presented in Figure 1 below.

Considering the information presented in the first part of this experiment, identify the following:
1) If the relation depicted between the words signaled in each of the sentences below belongs to the five-relation set discussed previously; and
2) If so, which relation label would most suitably describe the involved relationship.

Use, for reference, the following code:
a) if NO label describes the relationship between the signaled words;
b) if the label AGT (agent) is the most suitable one;
c) if the label CAG (co-agent) is the most suitable one;
d) if the label COB (affected co-thing) is the most suitable one;
e) if the label OBJ (affected thing) is the most suitable one;
f) if the label PTN (partner) is the most suitable one.

Figure 1. Instructions for identifying and classifying relations.

The 30-sentence set used in the test session, along with its corresponding English translation, is shown in Figure 2.

	SENTENCES
1.	A crise quebrou o empresário >> ???(quebrou, crise) *The crisis broke the business man.* >> ???(broke, crisis)
2.	A crise quebrou o empresário >> ???(quebrou, empresário) *The crisis broke the business man.* >> ???(broke, business man)
3.	A farsa acabou. >> ???(acabou, farsa) *The farce is over.* >> ???(is over, farce)
4.	A neve caía lentamente. >> ???(caiu, neve) *Snow felt slowly.* >> ???(felt, snow)

5.	Alugam-se casas. >> ???(alugar, casa) *Houses are rented* (also: *Someone rents houses*) >> ???(are rented, houses)
6.	Choveu canivete ontem. >> ???(choveu, canivete) *It rained knives yesterday >> ???(rained, knives) (Brazilian Idiom)*
7.	João jogou o vaso com Maria contra Pedro. >> ???(jogou, Maria) *John threw the bowl with Mary against Peter.* >> ???(threw, Mary)
8.	João jogou o vaso com Maria contra Pedro. >> ???(jogou, Pedro) *John threw the bowl with Mary against Peter.* >> ???(threw, Peter)
9.	João lutou com Maria para vencer a doença. >> ???(lutou,Maria) *John fought with Mary to win the disease.* >> ???(fought, Mary)
10.	João não teve filhos com Maria. >> ???(ter, João) *John did not have children with Mary.* >> ???(have, John)
11.	Maria esqueceu o dia do aniversário da filha. >> ???(esquecer, dia) *Mary forgot her daughter's birthday.* >> ???(forgot, birthday)
12.	Maria foi despedida. >> ???(despedir, Maria) *Mary was fired.* >> ???(fire, Mary)
13.	Maria lembrou Pedro do horário. >> ???(lembrou, horário) *Mary remembered Peter about the schedule.* >> ???(remembered, schedule)
14.	Maria morreu com a falta de oxigênio.. >> ???(morreu, falta) *Mary died with the lack of oxygen.* >> ???(died, lack)
15.	Maria namorou Pedro. >> ???(namorou, Maria) *Mary flirted (with) Peter.* >> ???(flirted, Mary)
16.	Maria não foi ao cinema com a vizinha. >> ???(foi, vizinha) *Mary did not go to the cinema with her neighbor.* >> ???(go, neighbor)
17.	Maria não quis matar Pedro! >> ???(matar, Maria) *Mary did not intend to kill Peter.* >> ???(kill, Mary)
18.	Maria não se sentiu bem. >> ???(sentir, Maria)
	Mary did not feel well. >> ???(feel, Mary)
19.	Maria nunca conquistou Pedro. >> ???(conquistou, Pedro) *Mary never conquered Peter.* >> ???(conquered, Peter)
20.	Maria parece cansada. >> ???(parece, Maria) *Mary looks tired.* >> ???(looks, Mary)
21.	Maria se esqueceu de João. >> ???(esquecer, João) *Mary forgot John.* >> ??(forgot, John)
22.	Maria se matou. >> ???(matou, Maria) *Mary killed herself.* >> ???(kill, Mary)
23.	O filme deu origem a muitas controvérsias. >> ???(deu, filme) *The movie raised many controversies* >> ???(raised, movie)
24.	O frio congelou o pássaro. >> ???(congelar, frio) *The cold froze the bird.* >> ???(froze, cold)
25.	O medo da morte provoca insônia. >> ???(provoca, medo) *Fear of death causes insomnia.* >> ???(causes, fear)
26.	O pai com os filhos matou a mãe. >> ???(matou, filhos) *The father with the children killed the mother.* >> ???(killed, children)
27.	O pássaro congelou com o frio. >> ???(congelar, frio) *The bird froze (i.e., was frozen) with the cold.* >> ???(froze, cold)
28.	Os carros se chocaram na estrada. >> ???(chocaram, carros) *The cars crashed each other on the road.* >> ???(crashed, cars)
29.	Pedro se parece com a mãe. >> ???(parece, mãe) *Peter looks like his mother.* >> ???(looks, mother)
30.	Precisa-se de funcionários. >> ???(precisar, funcionários) *Employees are needed.* (also: *Someone needs employees*) >> ???(need, employees)

* Students were presented only to the original Brazilian Portuguese sentence. In the translation from Portuguese into English we tried to preserve the Portuguese syntactic structure as often as possible, even when the resulting English sentence sounds agrammatical.

Figure 2. Test corpus.

4. Results

The results of the experiment were the following:

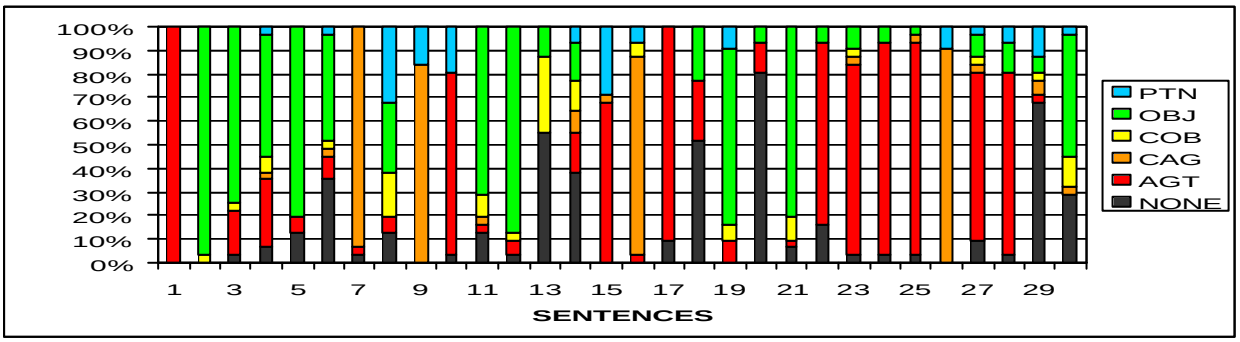


Figure 3. Distribution of BP-UNL enconvertings by subjects, with respect to the 5-relation labels set

Figure 4 below groups the results according to the agreement among enconverters.

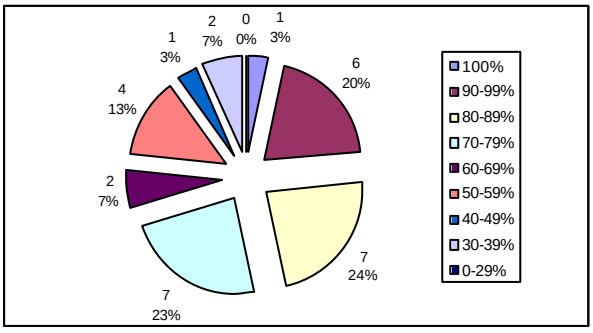


Figure 4. Agreement among enconverters.

A single relation (between "crise" (*crisis*) and "quebrou" (*to break*) in sentence 1: *"A crise quebrou o empresário" (= The crisis broke the business man*) led to an agreement of 100% among enconverters: they all used the 'agt' label in this case. There was an agreement between 90% to 99% on labeling relations in 6 sentences. Enconverters also agreed between 80% to 89% in assigning labels in 7 sentences. Other 7 sentences involved 70% to 79% agreement. In the remaining 9 sentences, agreement among enconverters was lower than 70%.

5. Case Study: Sentence 14

Sentence 14 ("Maria morreu com a falta de oxigênio." (literally: "Mary died with the lack of oxygen.") can be taken as a typical example of those involving considerable disagreement among enconverters. The relation between the verb "morreu" (to die) and the noun "falta" (lack) was encoded in varied ways, as follows: a) as an agent one (16%); b) as an object one (16%); c) as a co-object one (13%); d) as a co-agent one (10%); e) as a partner one (6%); and f) as none of the previous five relations (39%).

The unavoidable issue that follows from the above is why UNL labels were used in such apparently fuzzy way. Several reasons could be pinpointed here: a) the lack of expertise (or even of attention) of human enconverters', for they could not have had enough knowledge of language, or motivation, to carry on the experiment (although they are BP native speakers and seemed to be willingly helpful and interested in participating); b) the lack of clarity of the UNL Specification itself, even though there had been considerable discussion in the training session, for the problems posed by the enconverters to be tackled; c) the structure of the experiment itself, which was indeed too brief and too shallow to properly evaluate the human enconverters' performance; and, finally, d) the ambiguity of test sentences.

The analysis of the enconverters' choices certifies that disagreements are due to the latter point. Although it is unlikely for a BP speaker to say that 14 above, out of context, could have many different colliding meanings, the experiment has proved that apparently unambiguous sentences are unambiguous only apparently. Although eventually invisible, NL vagueness and indeterminacy would be pervasive in ordinary language,

Actually, none of the labels assigned to the relation between "morreu" (to die) and "falta" (lack) in sentence 14 could be considered wrong. The lack of oxygen could be understood in many distinct ways, such as:

a) an agent ("agt"), or the "initiator of the action" of "Mary dying" (or "killing Mary");

b) a co-agent ("cag"), or a "non-focused initiator of an implicit event that is done in parallel", in the sense it was not the lack of oxygen that killed Mary but either b.1) the situation (or the person) that has provoked the suppression of Mary's air supply or, in a more precise way, b.2) the reaction provoked (mainly in the brain) by the lack of oxygen;

c) an object ("obj") for the event described by "dying", since it is somehow "directly affected" by it, as the conclusion that the oxygen was lacking might be said to come directly from the fact that Mary died, otherwise no one would perceive that oxygen was lacking;

d) an affect co-thing ("cob"), or as being "directly affected by an implicit event done in parallel", if the observation that the oxygen was lacking were said not to come directly from the fact that Mary died, but from the fact that her lungs stopped working, which caused her to die;

e) a partner ("ptn"), for it could be somewhat "an indispensable non-focused initiator" of the action of "Mary dying", as if the main responsible for Mary's death was Mary herself (or someone else) that turned the oxygen suply off.

Besides such illustrations, many other relations can be said to hold between 'lack of oxygen' and 'die', namely, "met" (method), "man" (manner), "ins" (instrument), and "rsn" (reason), all easily applicable to such a case.

Such a variety proves that sentence 14 was indeed vague. The syntactic relation between the BP verb and its adjunct can convey many different semantic cases. Nevertheless, the UNL expression – whatever it may be – will have, in turn, a single interpretation, because relation labels are not supposed to overlap. The relations agt(die,lack), cag(die,lack), cob(die,lack), obj(die,lack), ptn(die,lack), although applicable to that very same NL sentence, are expected to label different (albeit related) phenomena. Indeed, to say agt(die,lack) is not the same as to say cag(die,lack) or ptn(die,lack). No intersection between these relations is envisaged in the UNL Specification, since they are meant to be exclusive[2].

This makes clear that the UNL specification forces filtering possible interpretations for NL sentences, in the sense a UNL expression must provide a completely unambiguous representation for the source sentence. As a matter of fact, although UNL is intended to be as expressive as any NL, UNL expressions cannot convey, at least at the relation level, NL vagueness and indeterminacy. Like any other formal language, UNL is committed to disambiguate NL sentences and, hence, to impoverish their semantic power.

Nevertheless, in no one of the above situations it is possible to say that a relation label is wrong, or that is completely inappropriate, although some of them may seem really unlikely to hold, depending on the context. The point is that the meaning of the sentence "Mary died with the lack of oxygen." is not encapsulated in the sentence itself but it is built out from the reading (and hence from the analysis) made by human enconverters. Since different enconverters have different underlying assumptions during their readings, the same BP phenomena can naturally imply different interpretations, which in turn lead to distinct UNL labeling. To conclude, it seems impossible to prevent subjectivity (or context-sensitiveness, or else, enconverter-sensitiveness) at that extent, no matter how univocal NL sentences seem to be.

6. Consequences

From the above it is possible to state that UNL should not seek for a straightforward correspondence between UNL expressions and NL sentences. It would be useless. As meaning is not encrypted in NL sentences but build through the analysis process, different enconverters will unavoidably propose different UNL expressions for the

[2] Accordingly, it is worthy to observe that the individuality of relations seems to be less strong when we consider other UNL relation labels set, e.g., that comprising "qua" (quantity), "nam" (name) and "pos" (possessor), which seems to be, to some extent and context, replaceable by "mod" (modification), implying that the latter can quite feasibly be at an uppermost level in a relation hierarchy. The same could be said of "met" (method) and "ins" (instrument), which seem to be under the scope of "man" (manner). Conversely, this does not mean that "mod" comprises any of "qua", "nam", or "pos", or that "man" embeds "met" and "ins". Instead, it does mean that both "mod" and "man" seem to share a comprehensive set of features with the relations that they replace. This is not the case of "agt", "cag", "cob", "obj", and "ptn", which seem to be in a more outstanding opposition.

very same NL sentence and many of these different expressions are legitimate.

Due to structure of UNL, UNL expressions cannot replicate NL sentence vagueness and indeterminacy. Enconverters are obliged therefore to choice a single interpretation among many different possible ones. This choice will be inevitably affected by the enconverters' context, which will be unreplicable itself by other enconverters. Once all these enconvertings will be valid, in the sense they are context-motivated, there will never be a one-to-one mapping between NL sentences and UNL expressions.

Accordingly, correctness, in UNL, instead of representing a (impossible) single possibility of enconverting, should rather be considered as fidelity to enconverters' intentions. UNL should clearly state that it would be up to the (human and machine) enconverter to decide what should the UNL representation be for a NL sentence. That is to say, the object of the UNL representation should be considered not exactly the meaning conveyed by the NL sentence but the *interpretation inferred by the enconverter from the use of that NL sentence in the enconverter's specific context*.

The fact that there could be more than a single (and adequate) UNL expression for the same NL sentence implies that UNL allows for flexibility in the enconverting process, although the UNL expression itself is not supposed to be flexible. It is up to the enconverter, and not the UNL specification itself, to decide which of the many possible interpretations is to be represented by a UNL expression. This is a significant UNL distinctive feature. Most formalisms do not allow for such variability and postulate that there should be a biunivocal relation between NL and its artificial representation. Otherwise, the formal representation would keep mirroring NL vagueness and indeterminacy, resulting useless.

The problem here is how to assure that enconverting flexibility will not prevent UNL from being a machine tractable language. As far as UNL expressions are dependent on the enconverter, there could be uncontrolled variations, which could blow out UNL into many different (and maybe mutually unintelligible) dialects.

This problem can be divided into two parts: 1) how to be sure that the UNL expression represents indeed what is intended by the enconverter; and 2) how to be able to generate, from such varied UNL expressions, NL grammatical sentences.

The first question is somewhat an educational problem. There are obviously misunderstandings and misuses of many relations. To say that it is up to the enconverter to decide which label should be used is not to say that the enconverter can do whatever he/she/it wants. The UNL Specification and other guidelines are to be followed. The relation "agt" must be applied to "a thing that initiates an action", and "ptn" should stand for "an indispensable non-focused initiator of an action". The relation "agt" cannot be used in a different sense: it would be wrong. Flexibility in encoding should not be mistaken for permissiveness. There are many correct UNL expressions for the same NL sentence, but there are also wrong UNL expressions.

The solution to such a problem cannot be, however, to state a rigid (a culture-, language-, context- and even enconverter-independent) relationship between a NL and UNL, otherwise UNL will not suffice to cope with inevitable varying enconvertings. The fact that meaning is build through the enconverting process and its main consequence, the fact that different enconverters will propose different expressions for the same NL sentence, should be both considered starting points, instead of something that one can or should avoid.

The best solution is, thus, to trust the enconverter (and maybe to certify enconverters), and to be conscious that, as in any other translation activity, there are good and bad translations, and bad translations do not prove that translating is not possible or that it does not work. Only time and enconverters' expertise can make UNL expressions better.

Nevertheless, to trust enconverters may imply making deconverting extremely difficult and costly. The more UNL allows flexibility in enconverting, the more costly will be UNL-NL deconverting, since the UNL expression may contain unexpected relations.

This is, however, a false problem. Deconverters are not committed to generate back the source sentence enconverted into UNL. Instead, they should be supposed to generate a NL sentence corresponding to the UNL expression. The original source sentence is definitely lost as it has been enconverted into UNL; only one of its possible interpretations (the one carried out by the enconverter) is preserved. Deconverters should take then UNL expression as the new source sentence, instead of using it just as an intermediate expression.

Furthermore, deconverting seems to be easier than enconverting, since much of the eventual meaning gaps may be inferred from the context by a human being (which is supposed to be the final user), instead of a machine. There is a very fragile break-even-point, from which generation results become excessively degraded, but the extent to which this happens will depend on the architecture of the UNL System.

7. Conclusion

The main conclusion to be extracted from the previous section seems to be a paradox: in multilingual MT Systems, in order to be a pivot language, UNL should not be treated as an interlingua, but as a source and a target language, at the same level as any other NL. Flexibility in enconverting brings UNL to be just like any other NL, in the sense it would allow UNL for coping with NL vagueness and indeterminacy, without sacrificing, however, the explicitness and clarity of UNL expressions, which would continue to be univocal and machine-tractable.

Acknowledgments

The authors acknowledge Mr. Tadao Takahashi, for his management of the Brazilian branch of the UNL Project, and to CNPq (Brazil) for the partial financial support. This work has also been partially supported by the UNU/IAS and the UNDL Foundation. The opinions expressed in the paper, however, do not necessarily represent the view of the UNL mentors.

References

Cole, R.A.; Mariani, J.; Uszkoreit, H.; Zaenen, A.; Zue, V. (Eds.) (1995). *Survey of the State of the Art in Human Language Technology*. NSF/CEC/CSLU. Oregon Graduate Institute. November.

(http://cslu.cse.ogi.edu/HLTsurvey/HLTsurvey.html)
Eco, U. (1994). *La recherche de la langue parfaite dans la culture européene*. Paris, France: Editions du Seuil.
Farwell, D. and Wilks, Y. (1993). ULTRA: A Multilingual Machine Translator. In S. Nirenburg (Ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.
Furuse, O. and Iida, H. (1992). Cooperation between transfer and analysis in example-based framework. In *Proceedings of the 14th International Conference on Computational Linguistics*. Nantes, France.
Hutchins, W. J. & Somers, H. L. (1992). *An Introduction to Machine Translation*. San Diego, CA: Academic Press.
Mitamura, T., Nyberg, E. and Carbonell, J. (1993). In S. Nirenburg (Ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.
Nirenburg, S., Carbonell, J, Tomita, M. and Goodman, K. (1992). *Machine Translation: A Knowledge-Based Approach*. San Mateo, CA: Morgan Kaufman.
Okumura, A., Muraki, K and Akamine, S. (1993). Multi-lingual Sentence Generation from PIVOT Interlingua. In S. Nirenburg (ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.
Tanaka, H. (1993). Multilingual Machine Translation Systems in the Future. In S. Nirenburg (ed.), *Progress in Machine Translation*. Washington, DC: IOS Press.
Uchida, H., Zhu, M. and Della Senta, T. (1999). *Universal Networking Language: A gift for a millennium.* Tokyo, Japan: The United Nations University.

Structural and lexical transfer from an UNL graph to an equivalent natural language dependency tree

Etienne Blanc, Gilles Sérasset, WangJu Tsai

GETA, CLIPS-IMAG
BP 53, F-38041 Grenoble cedex 09
{etienne.blanc, gilles.serasset, tsai}@imag.fr

Abstract

We describe the transfer of an UNL graph into a equivalent tree, allowing to build UNL deconverters using existing MT systems based on tree processing.

1. Introduction

In the Universal Networking Language, a text is represented as a graph where nodes, bearing "Universal Words" (UWs), are linked by directed arcs bearing semantic "Relations Labels". A particular node, the "entry node", is distinguished in the graph.

The structure of these UNL graphs makes them quite suited to be processed by various linguistic tools. In particular, the Deconversion (from a UNL graph into an equivalent Natural Language text) or the Enconversion (from a Natural Language text into a UNL graph) may be achieved not only using the specially devised Deco and Enco tools, but also using adapted existing classical MT systems. For instance, UNL to Russian, UNL to Chinese, UNL to French deconverters are being developed using transfer MT systems.

Most of the classical MT systems use tree representation and not graph representation. Therefore the first step in the deconversion based on such systems is a graph-to-tree transfer. The aim of this paper is to discuss such a transfer, and to present the method used in the UNL-to-French deconverter.

We will begin by an overall presentation of the UNL-to-French deconvertor based on the ARIANE-G5 generator of MT systems. We will then discuss in more detail the process of graph-to-tree transfer.

2. A UNL-to-French deconverter deriving from a classical transfer system

2.1. Ariane-G5, a generator of MT systems

ARIANE-G5 is a generator of MT systems, that is an integrated environment designed to facilitate the development of MT systems (Boitet, 1997). These MT systems are written by a linguist using specialized languages for linguistic programming. ARIANE is not devoted to a particular linguistic theory. The only strong constraint is that the structure representing the unit of translation (sentence or paragraph) must be a decorated tree.

Fig.1 shows an overview of a classical transfer MT system using the ARIANE environment. The processing is performed through the three classical steps : analysis, transfer and generation.

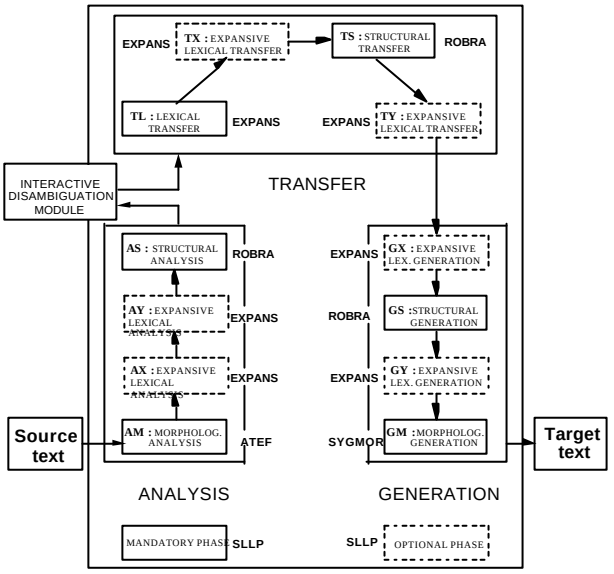


Figure.1 The Ariane-G5 environment as used for generating a transfer MT SYSTEM

2.2. Principle of the French Deconverter

Fig 2 shows an overview of the UNL-to-French deconverter using the ARIANE environment.

The first step is a graph-to-tree transfer, achieving both:

- the graph-to-tree structural transfer necessary for the ulterior Ariane processing
- a lexical "Universal Words" to French words lexical transfer.

The resulting tree is a classical "deep tree" ready for generation.

This first structural and lexical step will be discussed in detail below. The following classic generation step will not be discussed here.

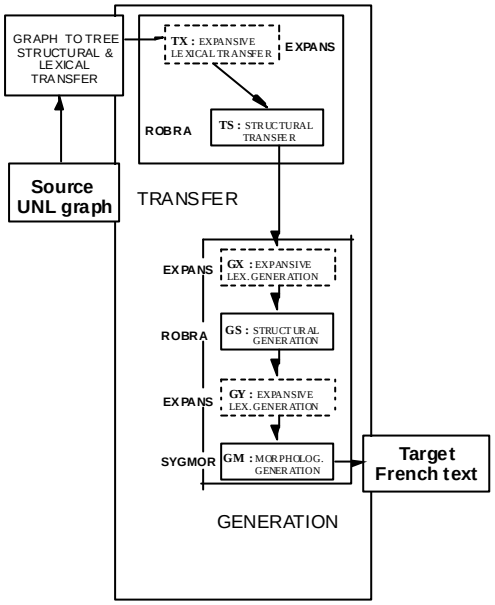


Figure 2 : The Ariane-G5 environment as used for generating a French deconverter.

3. UNL graph to NL tree structural transfer

The aim of the graph-to-tree structural transfer is to supply an output tree displaying all the structural information contained in the input UNL graph.

We will consider the following examples of tructural features encountered in a graph and needing some special coding in a tree are for instance:

- node having several mother nodes
- closed circuit
- hypergraph structure, that is graph containing nodes having themselves a graph structure (subgraphs, or "Compound Universal Words")

But before considering these examples, let's first illustrate the transfer on the simplest case, that is the transfer of a graph having in fact already a tree structure.

3.1. Graph with tree structure

In this simple case, the transfer is straightforward, as illustrated on figure 3.

This figure gives successively, from top to bottom:

- the meaning of the input graph as expressed in English
- the graph itself
- a sketch of its structure
- the structure of the equivalent tree as given by the structural transfer module (in this case the structure is the same as the structure of the graph)
- the decoration of the tree nodes.

The decoration of each node lists

- the Universal Word
- the semantic relation relative to its moither node (noted as a monovalued variable RSUNL)
- the attributes of the node (noted as a multivalued variable VARUNL)
- the id number (noted as the monovalued variable INST).

3.2. Graphs containing nodes with more than one mother node

In a tree, the root node has no mother node, and the other nodes have only one mother node. This is of course generally not the case for a graph, where all the nodes (including the entry one) may have several mother nodes.

Let's for instance consider the graph of fig. 4, where the entry node (« institute ») has a mother node (« establish ») the arc joining the first node to the second bearing the relation *obj*:

```
obj(establish(icl>found).@past,institute(ic
l>facilities).@present.@entry)
```

In order to get a tree, with a root node without mother node, the relation is inverted in the transfer module, and becomes

```
xxobj(institute(icl>facilities).@present.@e
ntry, establish(icl>found).@past)
```

where *xxobj* represents the inverse relation of the *obj* relation . The *obj* relation in the original graph expresses the fact that « institute » is the *obj* of establish, whereas the *xxobj* relation in the modofied graph expresses the fact that « establish » has « institute » as *obj*. Such an "inverted relation" is usally deconverted into French as a relative clause. The deconverted French text reads *"L'université des Nations Unie est un institut que l'Assemblée Générale des Nations Unies a fondé en 1975"."*

3.3. Graph containing a closed circuit

An equivalent tree structure of a graph containing a closed circuit may be obtained by opening the circuit, splitting one of its nodes as shown on fig.5 (the node "lecturer".splitted)

The new created node bears the same id number as the original one, indicating that it refers to the same object. In this example, this new node will be translated in French by the possessive "son", and the deconverter output reads *Le conférencier a lu son papier "*

3.4. Hypergraphs

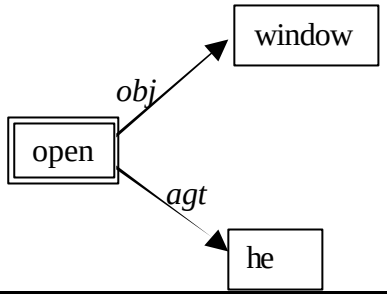
The processing of an hypergraph (graph containing subgraphs) is quite straightforward: the resulting tree is a tree containing subtrees.

English text: **He doesn't open the window.**

Graph :

```
agt(open(icl>do).@entry.@not,he)
obj(open(icl>do).@entry.@not,window.@def)
```

Graph structure:



Output tree:

```
                                          |-- 2:'WINDOW
                                          |
                                1:'OPEN' ---!-- 3:'HE'
```

Tree decoration:

```
1 'OPEN': VARUNL(ENTRY,NOT),INST(1)
    2 'WINDOW': VARUNL(DEF), RSUNL(OBJ),INST(1)
    3 'HE': RSUNL(AGT),INST(1)
```

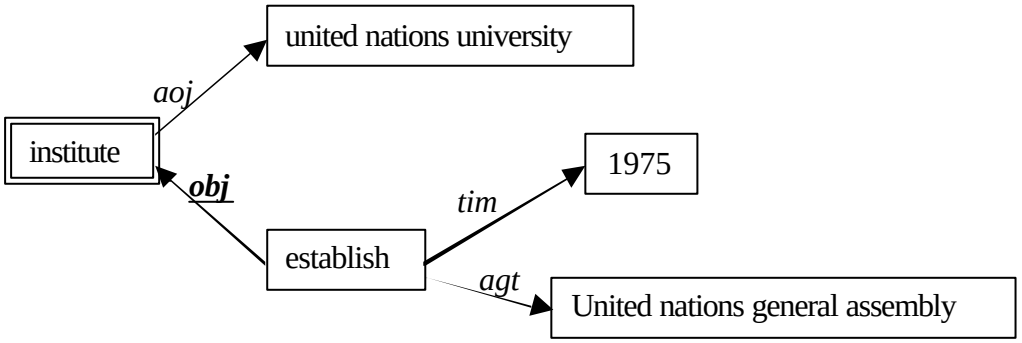
Fig 3. Structural transfer for a graph with tree structure..

English text: **The United Nations University is an institute which was founded by the United Nations General Assembly in 1975.**

Graph:

```
aoj(institute(icl>facilities).@present.@entry,united nations
university(icl>facilities))
obj(establish(icl>found).@past,institute(icl>facilities).@present.@entry)
agt(establish(icl>found).@past,united nations general
assembly(icl>organization))
tim(establish(icl>found).@past,1975)
```

Graph structure



Output tree

```
                        |-- 2:'UNITED NATION UNIVERSITY'
                        |                      |-- 4:'1975'
-- 1:'INSTITUTE' ------!-- 3:'ESTABLISH' -----!-- 5:'UNITED NATIONS GENERAL ASSEMBLY'
```

Tree decoration:

```
1 'INSTITUTE': VARUNL(PRESENT,ENTRY),INST(1)
    2 'UNITED NATIONS UNIVERSITY' RSUNL(AOJ),INST(1)
    3 'ESTABLISH': UL('<ESTABLISH>'), VARUNL(PAST), RSUNL(XXOBJ),INST(1)
      4 '1975':  RSUNL(TIM),CAT(CATCARD)
      5 UNITED NATIONS GENERAL ASSEMBLY': RSUNL(AGT),INST(1)
```

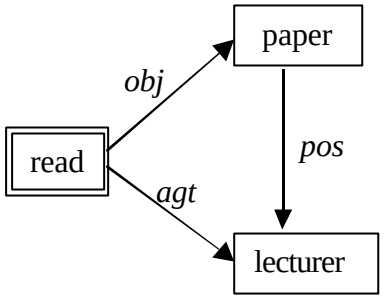
Figure 4 :Structural transfer of a graph whose entry node has a mother node

English text : **The lecturer read his paper.**

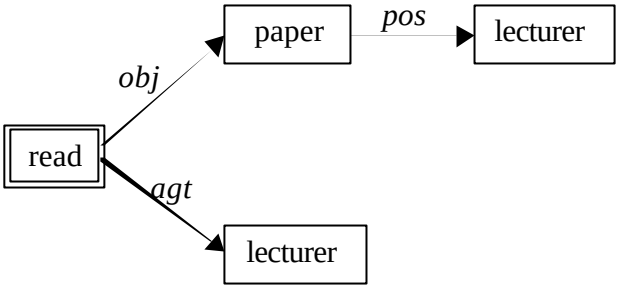
Graph :

```
agt(read(icl>do).@entry.@past,lecturer.@def)
obj(read(icl>do).@entry.@past,paper(icl>article))
pos(paper(icl>article),lecturer)
```

Graph structure:



Node splitting:



Output tree:

```
                                |-- 2:'PAPER' ----- 3:'LECTURER
                                |
                  1:'READ' -----!-- 4:'LECTURER
```

Tree decoration

```
1 'READ': VARUNL(ENTRY,PAST),INST(1)
    2 'PAPER': RSUNL(OBJ),INST(1)
      3 'LECTURER': RSUNL(POS),INST(1)
    4 'LECTURER': VARUNL(DEF), RSUNL(AGT),INST(1)
```

Figure 5. Structural transfer for a graph containing a closed circuit.

4. UNL graph to NL tree lexical transfer

The structure of the UNL universal words makes in principle the lexical transfer a straightforward process.

A Universal Word like *mouse(icl>animal)* comprises indeed an headword "*mouse*" and a restriction "*icl>animal*" whose aim is to disambiguate the UW : distinction between *mouse(icl>animal)* and *mouse(icl>device)*.

But in practice incompletness or inadequacies of the dictionaries leads either to use a treatment of the unknown word or an interactive lexical transfer.

4.1. Treatment of the unknown word

The treatment of the unknown words (that is of Uws whose NL language equivalents are not available in the dictionaries) may be based on the restriction of the UW and/or on the semantic relations the UW participates to.

4.1.1. Treatment of the unknown word based on the UW restriction

Using the restriction of the UW, we perform a partial treatment of the unknown word: the UW is not translated (the headword appears in the deconverted sentence), but the sentence is as far as possible correctly build.

This is shown on figure 6 where the graph contains two UWs supposed unknown. Testing the restrictions of the unknown UWs `rake(icl>do)` and `rake(icl>thing)` indicates that the first one is a verbal concept, the second one a thing concept, which allowed a correct construction of the sentence.

*English text :***He rakes the leaves with the big rake.**

Graph :

```
agt(rake(icl>do).@entry,he)
obj(rake(icl>do).@entry,leaf(fld>botany).@def.@pl)
ins(rake(icl>do).@entry,rake(icl>thing))
mod(rake(icl>thing),big(mod<thing))
```

French output text : **Il <<rake>> les feuilles avec le? grand? <<rake>>.**

Fig 6 Treatment of the unknown word based on the UW restrictions

4.1.2. Treatment of the unknown word based on the semantic relations

The semantic relations may also be used to determine the nature of the unknown word, allowing thus to obtain the correct sentence structure.

Figure 6 shows the deconversion result for a (unrealistic) graph where two unknown UWs without restrictions are present : *rake:01* and *rake:02* (the two different ids :01 and :02 indicate that these UWs are associated to two different nodes).

The different natures of both UWs were determined by using the semantic relations: the first instance of the UW rake, being the origin of an *agt* relation, was considered as a verbal concept, while the second one, being the target of an *ins* relation, was considered as a nominal concept.

English text **He rakes the leaves with the big rake.**

Graph : `agt(rake:01.@entry,he)`
`obj(rake:01.@entry,leaf(fld>botany)`
`.@def.@pl)`
`ins(rake:01.@entry,rake:02)`
`mod(rake:02,big(mod<thing))`

French output text: **Il <<rake>> les feuilles avec le? grand? <<rake>>.**

Fig 6 Treatment of the unknown word based on the semantic relations.

4.2. Interactive lexical transfer

Our local deconverter may work in an interactive lexical mode. In this mode, for each UW in the graph, the French equivalent(s) present in the dictionaries are displayed for choice (figure 7).

`Meeting(icl>event)` `Click on one item below`

`Entering a new equivalent`
 	
`meeting(icl>event)`
`réunion`
`CAT(CATN),GNR(FEM)`
 	
`meeting(icl>event)`
`rencontre`
`CAT(CATN),GNR(FEM)`

Figure 7 : Interactive lexical transfer

If no satisfactory equivalent is present in the dictionaries, the user may enter the correct equivalent, which is stored in an auxiliary dictionary, and becomes immediately available.

This interactive mode makes use of the PARAX-UNL hypertextual multilingual database (Blanc 1999)

5. Argument transfer

By argument transfer, we mean the relation between a UNL semantic relation and the corresponding syntactic function in the target natural language. It is not a one to one relation.

We will show here on an example how testing the restriction of a predicate may help finding the syntactic function associated to a semantic relation.

In the UNL language, one distinguishes the verbal concepts *do*, *occur*, *be*. For instance, the graph of fig. 8 contains the UW « open(icl>do) », whereas the graph of fig. 9 below contains the UW « open(icl>occur) ».

Both UWs are translated into French by the same verb, « ouvrir » (or in English by the same verb « to open »). But it is clear that in the case of « open(icl>do) », the subject syntactic relation for the French (or the English) verb corresponds to the *agt* relation (figure 8), but to the *obj* relation in the case of the « open(icl>occur) » UW.

That means that in such a case the restriction had to be tested in order to find the subject of the sentence.

He doesn't open the window.

`agt(open(icl>do).@entry.@not,he)`
`obj(open(icl>do).@entry.@not,window`
`.@def)`

Il n'ouvre pas la fenêtre.

Figure 8 The obj relation of this graph corresponds to the syntactic object relation in French or English

The window doesn't open.

`[S]`
`;<SUZHOU_4>`
`obj(open(icl>occur).@entry.@not,win`
`dow.@def)`
`[/S]`

La fenêtre n'ouvre pas.

Figure 9 The obj relation of this graph corresponds to the syntactic subjet relation in French or English

6. Conclusion

Such a UNL graph to Natural Language tree transfer proved to be quite feasible, and allowed us to reuse an existing French generator.

7. References

Boitet C. 1997 GETA's methodology and its current developments *PACLING'97, Meisei University, Ohme, Japan, sept 97, Proceedings 23-57.*

Blanc E (1999) PARAX-UNL, a large scale multilingual hypertextual database. *Proceedings of the 5th Natural Language Processing Pacific Rim Symposium 1999 (NLPRS 99), pp 507-510. Tsinghua University Press, Beijing 1999.*

Some Lexical Issues of UNL

Igor Boguslavsky

Institute for Information Transmission Problems, Russian Academy of Sciences
19, Bolshoj Karetnyj, 101447, Moscow, Russia
bogus@iitp.ru

Abstract.

The Universal Networking Language (UNL) developed by Dr. H. Uchida at the Institute for Advanced Studies of the United Nations University is a meaning representation language designed for multi-lingual communication in electronic networks, information retrieval, summarization and other applications. We discuss several features of this language relevant for correct meaning representation and multi-lingual generation and make some proposals aiming at increasing its efficiency.

1. UNL approach to the lexicon.

The Universal Networking Language (UNL) developed by Dr. H. Uchida at the Institute for Advanced Studies of the United Nations University is a meaning representation language designed for multi-lingual communication in electronic networks, information retrieval, summarization and other applications.

Formally, a UNL expression is an oriented hypergraph that corresponds to a natural language sentence in the amount of information conveyed. The arcs of the graph are interpreted as semantic relations of the types agent, object, time, reason, etc. The nodes of the graph can be simple or compound. Simple nodes are special units, the so-called Universal Words (UWs) which denote a concept or a set of concepts. A compound node (hypernode) consists of several simple or compound nodes connected by semantic relations.

In addition to propositional content ("who did what to whom"), UNL expressions are intended to capture pragmatic information such as focus, reference, speaker's attitudes and intentions, speech acts, and other types of information. This information is rendered by means of attributes attached to the nodes.

After 6 years of the UNL project development, it is possible to take stock of what has been achieved and what remains to be done. In this presentation, I am going to concentrate on one of the central problems with which any artificial language is faced if it is designed to represent meaning across different natural languages. It is a problem of the language vocabulary.

I would like to single out three distinctive features of the UNL dictionary organization.

1. **Flexibility.** There is no fixed set of semantic units. There is only a basic semantic vocabulary that serves as a building material for free construction of derivative lexical units with the help of semantic restrictions. This makes it possible to balance to some extent the non-isomorphism of lexical meanings in different languages.

2. **Bottom-up approach.** The UNL dictionary consisting of Universal Words is not constructed a priori, top-down. Since it should contain lexical meanings specific to different languages, it grows in an inductive way. It receives contributions from all working languages. Due to this, one can expect that linguistic and cultural specificity of different languages will be represented more fully and more adequately than it would be possible under the top-down approach.

3. **Knowledge base.** As the UNL dictionary comprises unique semantic complexes lexicalized in different natural languages, we are facing the task of bridging the gap between them. It is supposed to be done by means of the Knowledge Base – a network of UNL lexical units connected by different semantic relations. Special navigation routines will be developed that will help to find the closest analogue to a lexical meaning not represented in the given language.

There are, however, some circumstances that impede full realization of these features, at least at the moment. Inductive storing of UWs from different languages is a good idea, but this process should be well organized. If a specific UW that is not self-evident is introduced to the UNL dictionary, it should necessarily be supplied at least by an informal comment to make it understandable to other users. Lucidity and easy interpretability of UWs is a goal at which all the developers of the UNL dictionary should aim.

Below, I am going to discuss in more detail two problems that have not so far received sufficient attention in UNL: the argument frames and lexical collocations.

2. Argument frames.

The need to introduce the information on the arguments does not seem to require justification. Any meaning representation language should have an ability to draw a distinction between the argument and non-argument links of predicates. In the UNL expressions, semantic links between the UWs are represented by means of UNL semantic relations. UNL disposes of an inventory of relations which, according to the latest specification, contains 41 items. Here are some examples of the UNL relations:

agt – agent (*John runs*),
obj – object (*read a book, A tree grows*),
ben – beneficiary (*He did not do anything for her*),
cag – co-agent (*I live with him*),
cob – co-object (*He fell into the river with the car*),
aoj – a thing which is in a certain state or is ascribed a property (*I love Mary; my brother is a student*).
dur – duration (*He worked nine hours*),
fmt – a range between two things (*He worked from Monday till Sunday*),
gol – final state (*turn red*),

ins – instrument (*observe with the telescope*),
met – method or means (*separate by cutting*),
pos – possession (*John's mother*),
rsn – reason (*They quarrel because of money*).

It is well known that for correct generation it is essential to know the argument structure of the predicates and the way each argument is expressed in the sentence. The UNL dictionary does not contain explicit information on the argument structure. According to the UW manual, the restrictions which should be included in the UW definitions are not meant for this purpose. As the UNL relations roughly correspond to semantic roles, it is supposed that each argument can be reliably identified based on its semantic role. However, this is not the case. Numerous attempts to construct a set of semantic relations, made over the last decades, showed that only a part of the relations between the words can be unambiguously interpreted in terms of semantic roles. In many cases this interpretation is largely arbitrary. This could not be a problem for the purposes of generation, if it were possible to assign semantic roles in a consistent way. Unfortunately, in practice it is hardly possible, especially when it is done by different people trained in different frameworks and working in different countries. The UNL texts compiled by the UNL project participants from 14 countries over the last years abound in mismatches in the representation of the same or very similar phenomena. Not surprisingly, most of them concern the representation of argument relations. For example, the phrase *base on respect* was interpreted by one team by means of the locative relation (lpl) and by another team by means of the comparative relation (bas), *freedom for all* was described with the purpose relation (pur) and with the beneficiary relation (ben), *bottleneck for the flow of information* received two labels – purpose (pur) and object (obj). Very often, the interpretation of a phrase in the corpus was motivated by the surface form rather than by its meaning. A typical example is *relations among nations* which was described by means of the locative relation obviously under the influence of the literal meaning of *among*. However, nations are by no means the place where relations occur. Rather, nations are participants of the "relations" situation and therefore are more likely to be objects (obj).

Sometimes the motivation behind the use of certain relations may be difficult to understand (at least, this is the case for the author of this paper). For example, in one of the sentences of the corpus, the argument structure of the verb *prevent* was presented as follows:

(1) *Nothing* (obj) *prevents members* (ben) *from discussing* (gol) *this problem*.

In our opinion, these problems are rooted not so much in the erroneous use of relations as in the fundamental impossibility of a consistent interpretation of all argument relations in terms of a small number of semantic roles.

What could one do to avoid the mismatches?

First, one could renounce using semantic roles in cases in which they are not obvious and replace them by semantically uninterpreted relations (subject, first object, second object, etc.). In this case, sentence (1) will receive a more transparent representation:

(2) *Nothing* (subject) *prevents members* (1 object) *from discussing* (2 object) *this problem.*

Obviously, it will be in many cases easier for those who write UNL expressions to develop a common approach to deciding which argument is the first object and which is the second than a common approach to finding appropriate semantic roles for them.

Second, one could accept the proposal of the French team and assign special markers to the case relations when they attach arguments (for example, @A would correspond to the first argument, @B – to the second, etc.). In this case, sentence (1) would be represented as:

(3) *Nothing* (obj.@A) *prevents members* (ben.@B) *from discussing* (gol.@C) *this problem.*

This would certainly reduce the area of uncertainty, but not eliminate it completely. To be able to interpret representation (3), the deconverter should know in advance the argument frame of the UW *prevent*. Otherwise, the uniformity of interpretation will still not be ensured. The only way to eradicate any ground for discordance between different users of the UNL language is to LIST ALL THE ARGUMENT STRUCTURES IN THE UNL DICTIONARY.

To incorporate this proposal, one need not introduce to the dictionary format any new possibilities: the existing apparatus of restrictions is quite sufficient. The only – but very serious – problem is to acknowledge that the argument frame should be explicitly and systematically specified in the UWs. If this is done, then one could keep using semantic roles in all the cases. For example, the word *bottleneck* (in the meaning of an obstacle) can receive the information that its syntactic object (*for something*) has the semantic role "pur" (or any other role which seems appropriate to the lexicographer). If every predicate is supplied with this information in the UNL dictionary, the discordance of opinion between different UNL users will become their private concern and the uniform treatment of the UNL relations in the most controversial zone – that of the argument relations – will be fully assured.

It should be emphasized however that in a general case the marking of the argument frame in a UW is not sufficient either. In some cases the same relation can attach to a UW both an argument and a free adjunct. For example, emotional states (of the type *be afraid, be surprised, be angry,* etc.) have an argument denoting a cause of the state. In sentence (4)

(4) *She is afraid to go out alone at night*

going out alone at night is what makes her to be in the state of fear. Therefore, relation "rsn" between *afraid* and *go out alone at night* is appropriate. On the other hand, *afraid* can have a non-argument cause, as in (5):

(5) *She is afraid (to go out alone at night), because this area is not very safe.*

Even if UW "afraid" is assigned a cause as one of the arguments (afraid(rsn>*)), we should know whether or not a "rsn"-link in the UNL expression denotes this argument. A good solution would be to mark the argument relation by a special label, as proposed in (3). Then, (5) will be represented as (6):

(6) rsn.@A(afraid(rsn>*), go out)
rsn(afraid(rsn>*), safe)

3. Lexical collocations.

Lexical collocations pose a serious problem for any language designed for representing meaning. Here are some examples of collocations from English: *give a lecture, come to an agreement, make an impression, set a record, inflict a wound; reject an appeal, lift a blockade, break a code, override a veto; strong tea, weak tea, warm regards, crushing defeat; deeply absorbed, strictly accurate, closely acquainted, sound asleep; affect deeply, anchor firmly, appreciate sincerely.* For simplicity, I will only dwell below on verbal collocations.

One of the problems such collocations raise is as follows. Some of the members of these collocations do not have a full-fledged meaning of their own. For example, the verb *give* in the collocation *give a lecture* does not denote any particular action. Its meaning, or rather its function, is the same as that of *take* in the collocation *take action*, or that of *make* in *make an impression*. The verbs *give, take* and *make* in these collocations are practically completely devoid of any meaning. Still, they have a very definite function – that of a support verb. This function is exactly the same in all the three cases, and nevertheless the verbs are by no means interchangeable. One cannot say **take an impression, *give action* or **make a lecture.* Moreover, this function is not only performed by different verbs with respect to different nouns. Very often, similar nouns in different languages require different verbs. For example, in Russian a lecture is not given but read, an action is not taken but accomplished, an impression is not made but executed.

How should these phenomena be treated in UNL? In particular, what UWs should be used for support verbs? The current practice suggests that UWs should be constructed on the basis of the source languages. Each language center should produce UWs for the words of its language, without any regard to other languages or any general considerations. A UNL expression and the UWs it consists of are considered adequate if they allow generating a satisfactory text in the same language they originated from. To what extent is this approach applicable to lexical collocations?

To answer this question, we will consider a concrete example. Suppose we have to convert to UNL Russian sentences with the meaning (7), (8), (9) or (10):

(7) *They began the war.*

(8) *We began the battle.*

(9) *The army suffered heavy losses.*

(10) *He took a shower.*

The problem is that in these contexts Russian uses quite different verbs than English. In Russian, correct sentences would be:

(7a) *They undid (razvjazali) the war.*

(8a) *We tied up (zavjazali) the battle.*

(9a) *The army carried (ponesla) heavy losses.*

(10a) *He received (prinjal) a shower.*

If UWs for support verbs in sentences (7a) – (10a) are constructed on the basis of Russian, they would look as follows: "undo(obj>war)", "tie up(obj>battle)", "carry(obj>loss)", and "receive(obj>shower)". These UWs will allow the Russian deconverter to produce perfect Russian sentences (7a) - (10a). In this case, the condition for adequacy mentioned above is met. Still, I would not consider UNL expressions based on these UWs adequate. They are produced without any regard for anything except the needs of Russian deconversion and are not fit for other purposes. In particular, these UWs are incomprehensible for anybody except Russians and it is doubtful that any other deconverter will be able to produce acceptable results from them. UWs originating from English will probably look like "take(obj>shower)", "begin(obj>thing)", "suffer(obj>loss)". To generate English sentences (7) - (10) from the UNL expressions constructed on the basis of (7a) – (10a), one would need to somehow ensure the equivalence of UWs "carry(obj>loss)" and "suffer(obj>loss)" in the Knowledge Base. This does not seem to be a natural and easy thing to do. Therefore, UWs for support verbs should not be constructed based on the lexical items of the source language.

Another possibility would be to make use of the co-occurrence properties of English lexical items. UNL vocabulary employs English words as labels for UWs and their meanings – as building blocks for UNL concepts which can be to a certain extent modified by means of restrictions. If lexical labels and meanings of UWs have been borrowed from English, their combinatorial properties can also be determined by the properties of corresponding English words. In this case, UWs and UNL expressions for sentences (7a) – (10a) will be identical to those for (7) – (10).

The advantage of this solution is obvious: since knowledge of English is indispensable for all the developers of X-to-UNL dictionaries, they can be sure that UWs for support verbs they produce are understandable and predictable. This solution has also drawbacks.

First, the inventories of support verbs in different languages are different. Therefore, we will often be faced with gaps in the lexical system of English and find no equivalent for a verb we need. Second, support verbs are bad candidates for the status of UWs. They do not denote any concept. Different support verbs often do not differ in meaning but only in their co-occurrence properties. It seems unreasonable to have different UWs to represent *take* (in *take action*), *make* (in *make an impression*) and *give* (in *give a lecture*), since the difference between these words is not semantic but only combinatorial. This difference should not be preserved in a meaning representation language.

The best solution would be to abstract from asemantic lexical peculiarities of support verbs and adopt a language-independent representation of these phenomena. Theoretical semantics and lexicography have long ago suggested a principled approach to the whole area of lexical collocations. It is the well-known theory of lexical functions by I. Mel'čuk implemented in the Explanatory combinatorial dictionaries of Russian and French (Mel'čuk 1974; Mel'čuk & Zholkovsky 1984; Mel'čuk et al. 1984, 1988, 1992, 1999). Possible use of lexical functions in NLP is discussed in (Apresjan et al. (in print)). Briefly, the idea of lexical functions is as follows. For more details, the reader is referred to the works mentioned above.

A prototypical lexical function (LF) is a general semantic relation R obtaining between the argument lexeme X (the keyword) and some other lexeme Y which is the value of R with regard to X (by a lexeme in this context we mean a word in one of its lexical meanings or some other lexical unit, such as a set

expression). Sometimes Y is represented by a set of synonymous lexemes Y_1, Y_2, …, Y_n, all of them being the values of the given LF R with regard to X; e. g., MAGN (*desire*) = *strong / keen / intense / fervent / ardent / overwhelming*.

There are two types of LFs – paradigmatic (substitutes) and syntagmatic (collocates, or, in Mel'čuk's terms, parameters).

A substitute LF is a semantic relation R between X and Y such that Y may replace X in the given utterance without substantially changing its meaning, although some regular changes in the syntactic structure of the utterance may be required. Examples are such semantic relations as synonyms, antonyms, converse terms, various types of syntactic derivatives and the like.

A collocate LF is a semantic relation R between X and Y such that X and Y may form a syntactic collocation, with Y syntactically subordinating X or vice versa. R itself is a very general meaning which can be expressed by many different lexemes of the given language, the choice among them being determined not only by the nature of R, but also by the keyword with regard to which this general meaning is expressed. Typical examples of collocate LFs are such adjectival LFs as MAGN = 'a high degree of what is denoted by X', BON = 'good', VER = 'such as should be' and also support verbs of the OPER/FUNC family. Examples of the latter are OPER1 = 'to do, experience or have that which is denoted by keyword X (a support verb which takes the first argument of X as its grammatical subject and X itself as the principal complement)'; OPER2 = 'to undergo that which is denoted by keyword X (a support verb which takes the second argument of X as its grammatical subject and X itself as the principal complement)'; FUNC1 = 'to originate from (a support verb which takes X as its grammatical subject and the first argument of X as the principal complement)'; FUNC2 = 'to bear upon or concern (a support verb which takes X as its grammatical subject and the second argument of X as the principal complement)'.

If used in UNL, lexical functions will ensure a consistent, exhaustive and language-independent representation of support verbs and all other types of restricted lexical co-occurrence. For example, English and Russian support verbs we discussed above – *take* (*a decision, a shower*), *make* (*an impression*), *give* (*a lecture*), *suffer* (*losses*), *prinimat'* (*reshenie* 'decision', *dush* 'shower'), *proizvodit'* (*vpechatlenie* 'impression'), *chitat'* (*lekciju* 'lecture'), *nesti* (*poteri* 'losses') – are correlates of the same lexical function – OPER1.

Being abstract and completely language-independent, lexical functions are devoid of all the drawbacks discussed above and can serve as an optimal solution to the problem of representation of the lexical collocations in UNL.

4. Acknowledgements.

This work has been supported by the Russian Foundation of Fundamental Research (grants Nos. 00-15-98866 and 01-07-90405).

5. References.

Apresjan Ju., I. Boguslavsky, L. Iomdin, L. Tsinman (in print). *Lexical function collocations in NLP*.

Mel'čuk I. A., 1974. *Opyt teorii lingvisticheskix modelej "Smysl – Tekst"* [A Theory of Meaning – Text Linguistic Models"]. Moscow, Nauka, 314 p.

Mel'čuk I. A., Zholkovskij A.K., 1984. *Tolkovo-kombinatornyj slovar' sovremennogo russkogo jazyka.* [An Explanatory Combinatorial Dictionary of the Contemporary Russian Language] Wiener Slawistischer Almanach, Sonderband 14, 992 p.

Mel'čuk I., Nadia Arbatchewsky-Jumarie, Lidija Iordanskaja, Adèle Lessard, 1984. *Dictionnaire explicatif et combinatoire du français contemporain, Recherches lexico-sémantiques I.* Les Presses de l'Université de Montréal.

Mel'čuk I., Nadia Arbatchewsky-Jumarie, Louise Dagenais, Léo Elnitsky, Lidija Iordanskaja, Marie-Noëlle Lefebvre, Suzanne Mantha, 1988. *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques II.* Les Presses de l'Université de Montréal.

Mel'čuk I., Nadia Arbatchewsky-Jumarie, Lidija Iordanskaja, Suzanne Mantha, 1992. *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques III.* Les Presses de l'Université de Montréal.

Mel'čuk I., Nadia Arbatchewsky-Jumarie, Lidija Iordanskaja, Suzanne Mantha et Alain Polguère, 1999. *Dictionnaire explicatif et combinatoire du français contemporain. Recherches lexico-sémantiques IV*. Montréal.

A rationale for using UNL as an Interlingua and more in various domains

Christian BOITET
GETA, CLIPS, IMAG
385, av. de la bibliothèque, BP 53
F-38041 Grenoble cedex 9, France
Christian.Boitet@imag.fr

LREC-02 First International Workshop on UNL, other Interlinguas and their Applications, 1 June 2002

Abstract

The UNL *language* of semantic graphs may be called as a "semantico-linguistic" interlingua. As a successor of the technically and commercially successful ATLAS-II and PIVOT interlinguas, its potential to support various kinds of text MT is certain, even if some improvements would be welcome, as always. It is also a strong candidate to be used in spoken dialogue translation systems when the utterances to be handled are not only task-oriented and of limited variety, but become more free and truly spontaneous. Finally, although it is not a true representation language such as KRL and its frame-based and logic-based successors, and although its associated "knowledge base" is not a true ontology, but rather a kind of immense thesaurus of (interlingual) sets of word senses, it seems particularly weel suited to the processing of multilingual information in natural language (information retrieval, abstracting, gisting, etc.).

The UNL *format* of multilingual documents aligned at the level of utterances is currenly embedded in html (call it UNL-html), and used by various tools such as the UNL viewer. By using a simple transformation, one obtains the UNL-xml format, and profit from all tools currently developed around XML. In this context, UNL may find another application in the localization of multilingual textual resources of software packages (messages, menu items, help files, and examples of use in multilingual dictionaries.)

Keywords: UNL, multilingual communication, cross-lingual information retrieval, localization

Introduction

UNL is the name of a project, of a meaning representation language, and of a format for "perfectly aligned" multilingual documents. There is some hefty controversy about the use of the UNL language as an "interlingua", be it for translation or for other applications such as cross-lingual information retrieval. On the other hand, there is almost no discussion on the UNL format, in its current form, embedded in HTML, or some directly derivable form, embedded in XML.

We argue that the UNL language is indeed a good interlingua for automated translation, ranging from fully automatic MT to interactive MT of several kinds through, we believe, spoken translation of non task-oriented dialogues. It is also more than that, due to the associated "knowledge base", and has a great potential in textual information processing applications.

We will first give our view of what the UNL language is, and then develop a "rationale" for using the UNL language UNL along the previous lines. We will then describe some interesting potential uses of the UNL format in an "XML-ized" form.

1. The UNL language

The UNL representation is made of "semantic graphs" where a graph expresses the meaning of some natural language utterance. Nodes contain lexical units and attributes, arcs bear semantic relations. Connex subgraphs may be defined as "scopes", so that a UNL graph may be a hypergraph.

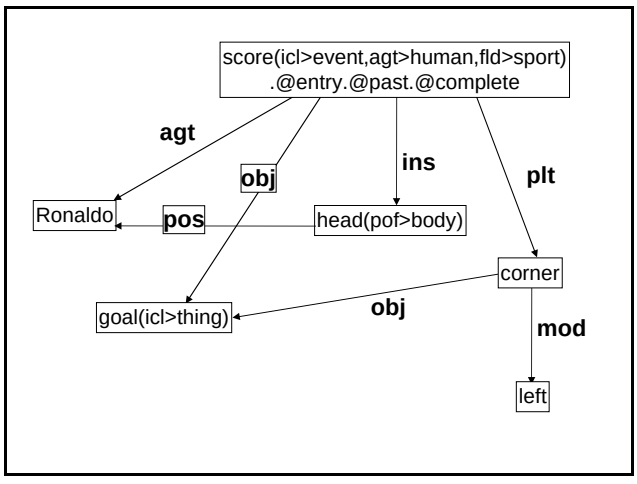


Fig. 1: a possible UNL graph for "Ronaldo has headed the ball into the left corner of the goal"

The lexical units, called Universal Words (in French, not "mot universel" but better "Unité de Vocabulaire Virtuel" or UVV or UW), represent word meanings, something less ambitious than concepts. Their denotations are built to be intuitively understood by developers knowing English, that is, by all developers in NLP. A UW is an English term or pseudo-term possibly completed by semantic restrictions.

A UW such as "process" represents all word meanings of that lemma, seen as citation form

(verb or noun here). The UW "process(icl>do, agt>person)" covers the verbal meanings of processing, working on, etc.

The attributes are the (semantic) number, genre, time, aspect, modality, etc.

The 40 or so semantic relations are traditional "deep cases" such as agent, (deep) object, location, goal, time, etc.

One way of looking at a UNL graph corresponding to an utterance U-L in language L is to say that it represents the abstract structure of an equivalent English utterance U-E as "seen from L", meaning that semantic attributes not necessarily expressed in L may be absent (e.g., aspect coming from French, determination or number coming from Japanese, etc.).

2. Some arguments for using the UNL language in various contexts

To show that using UNL is not only a workable but a good or perhaps the best idea at the moment, we can say that

- the "pivot" technique HAS BEEN not only experimented but deployed successfully (ATLAS, PIVOT, ULTRA, KANT).
- in particular, ATLAS-II (Fujitsu) is built on the basis of a pivot from which the UNL representation has evolved. The main designer of UNL, H. Uchida, was also the main designer of ATLAS-II.
- ATLAS-II has been recognized as the best EJ/JE MT system in Japan for over 10 years and has a very large coverage (586,000 words in English and Japanese).
- interlingual representations can not in principle be used (alone) to achieve the highest quality achievable by transfer systems, BUT they can give quite high quality as demonstrated by ATLAS-II.
- due to the precise nature of UNL, it is possible for human non-specialists to improve a UNL representation interactively, a posteriori, from any UNL-related language, and on demand (meaning partially — think of "lazy improvement").
- in many contexts other than translation, an interlingual, semantic-oriented representation like UNL is actually the best solution. For example, all applications related to information processing in multilingual contexts don't need a very precise representation of the FORM of the information, they need a precise ENOUGH representation of the INFORMATION CONTENT of the information.
- applications such as information retrieval and abstracting have already been prototyped successfully with UNL. It is far easier to generate SQL or SQL-like queries and answers from a UNL form than from text in many languages.

3. Applications of the UNL format

The UNL *format* of multilingual documents aligned at the level of utterances is currenly embedded in html (call it UNL-html). A sentence is represented between the [S] and [/S] tags. Its original text is contained between {org:el} (English, here) and {/org}, its UNL graph between {unl} and {/unl}, each French version between {fr} and {/fr}, and analogously for other languages. Atrtibutes such as version, date, location, author, etc. may appear in the tags. Here is a slightly simplified example of a file in UNL-html format.

```
<HTML><HEAD><TITLE>
Example 1  El/UNL
</TITLE></HEAD><BODY>
[D:dn=Mar Example 1, on= UNL French, mid=First.Author@here.com]
[P]
[S:1]
{org:el}I ran in the park yesterday.{/org}
{unl}
agt(run(icl>do).@entry.@past,i(icl>person))
plc(run(icl>do).@entry.@past,park(icl>place).@def)
tim(run(icl>do).@entry.@past,yesterday)
{/unl}
{cn dtime=20020130-2030, deco=man}
我昨天在公園裡跑步 {/cn}
{de dtime=20020130-2035, deco=man}
Ich lief gestern im Park. {/de}
{es dtime=20020130-2031, deco=UNL-SP}
Yo corri ayer en el parque.{/es}
{fr dtime=20020131-0805, deco=UNL-FR}
J'ai couru dans le parc hier. {/fr}[/S]
[S:2]
{org:el}My dog barked at me.{/org}
{unl}
agt(bark(icl>do).@entry.@past,dog(icl>animal))
gol(bark(icl>do).@entry.@past,i(icl>person))
pos(dog(icl>animal),i(icl>person))
{/unl}{de dtime=20020130-2036, deco=man}
Mein Hund bellte zu mir.{/de}
{fr dtime=20020131-0806, deco=UNL-FR}
Mon chien aboya pour moi. [/S] [/P][/D]
</BODY></HTML>
```

The French versions have been produced automatically while the German and Chinese versions have been translated manually.

The output of the UNL viewer for French is:

```
<HTML><HEAD><TITLE>
Example 1  El/UNL
</TITLE></HEAD><BODY>
J'ai couru dans le parc hier.
Mon chien aboya pour moi.
</BODY></HTML>
```

and will probably be displayed by a browser as:

Example 1 El/UNL

J'ai couru dans le parc hier. Mon chien aboya pour moi.

and similarly for all other languages.

The UNL viewer produces on demand as many html files as languages selected and sends them to any available browser.

The UNL-html format predates XML, hence the special tags like [S] and {unl}, but it is easy to derive from it an XML format and to transform the documents into an equivalent "UNL-xml" format. Then, using DOM and javaScript, it is possible to produce various views, including that of a classical viewer, a bilingual or multilingual editable presentation, and a revision interface where not only the text but the UNL graph and possibly other structures may be directly manipulated.

Let us take an example from an experiment performed for the "Forum Barcelona 2004" on documents in Spanish, Italian, Russian, French and Hindi. Hindi and Russian are not shown, but Japanese has been added by hand. The XML form is simplified.

Correct sentences are produced by the deconverters from correct and complete UNL graphs.

Suppose for the sake of illustration that some UNL graph has been produced from a Chinese version, and does not contain definiteness and aspectual information. All results may be wrong wrt articles, and some wrt aspect.

```
<unl:S num="1">
<unl:org lg="cn">在博覽會之後，城市 將獲得一片海岸域 </unl:org>
<unl:unl>
<unl:arc> agt(retrieve(icl>do).@entry.@future, city) </unl:arc>
<unl:arc> tim(retrieve(icl>do).@entry.@future, after) </unl:arc>
<unl:arc> obj(after, Forum) </unl:arc>
<unl:arc> obj(retrieve(icl>do).@entry.@future, zone(icl>place).@indef) </unl:arc>
<unl:arc> mod(zone(icl>place).@indef, coastal) </unl:arc> </unl:unl>
<unl:cn> 在博覽會之後，城市 將獲得一片海岸域 </unl:cn>
<unl:el> After a Forum, a city will retrieve a coastal zone.</unl:el>
<unl:es> Ciudad recobrará una zona de costal después Foro. </unl:es>
<unl:fr> Une cité retrouvera une zone côtière après un forum. </unl:fr>
<unl:it> Città ricuperarà une zona costiera dopo Forum. </unl:it>
<unl:jp> フォーラムの後で, 都市は沿岸水域を取り出す。 </unl:jp>
</unl:S>
```

The idea of "coedition" is applicable if there is a UNL graph associated with a segment one wants to modify. The goal is to share the revisions across languages, by reflecting them on the UNL graph, e.g.

- add ".@def" on the nodes containing "city", "Forum".
- replace "retrieve" by "recover" and add ".@complete" on the node containing it.

It is not possible in principle to deduce the modification on the graph from a modification on the text. For example, replacing "un" ("a") by "le" ("the") does not entail that the following noun is determined (.@def), because it can also be generic ("il aime la montagne" = "he likes mountains"). Hence, the technique envisaged is that:

- revision is not done by modifying directly the text, but by using a menu system,
- the menu items have a "language side" and a hidden "UNL side",
- when a menu item is chosen, only the graph is transformed, and the action to be done on the text is stored and shown next to its focus in the "To Do" zone,
- at any time, the new graph may be sent to the L0 deconverter and the result shown. If is is satisfactory, that shows that errors were due to the graph and not to the deconverter, and the graph may be sent to deconverters in other languages. Versions in some other languages known by the user may be displayed, so that improvement sharing is visible and encouraging.

New versions will be added with appropriate tags and attributes in the original multilingual document in UNL-xml format, or in a DBMS, so that nothing is ever lost, and cooperative working on a document is feasible. UNL may find another application in the localization of multilingual textual resources of software packages (messages, menu items, help files, and examples of use in multilingual dictionaries.)

Apart of the "coedition", there are many other portential applications of UNL, such as:

- crosslingual information retrieval, on which we are currently working,
- abstracting & gisting, which has been prototyped at NecTec and in India,
- localization of software packages: messages in multiple languages could be created from UNL graphs produced from a graphical interface or by enconversion, and then sent to appropriate deconverters.

For this last point, we have found how to represent messages including variables (such as integers, file names etc.), but not yet how to handle messages including morphological or even lexical variants (as "4 goda / 5 let" for "4 years / 5 years" in Russian).

Conclusion

The UNL language is an artificial interlingua, embeddable in html or xml formats for multilingual document representation and processing. Because of its both abstract and linguistic nature, the UNL language offers many more interesting potential applications than other types of interlingua such as task and/or domain specific interlingua.

The history of MT shows that UNL will also be usable in the context of high-quality MT,

quality being obtained through typology specialization and/or interactive improvement, a priori (interactive disambiguation after all-path robust analysis) and/or a posteriori by coedition of the text in any language and the corresponding UNL graph.

References

Blanc É. & Guillaume P. (1997) *Developing MT lingware through Internet : ARIANE and the CASH interface*. Proc. of Pacific Association for Computational Linguistics 1997 Conference (PACLING'97), Ohme, Japon, 2-5 September 1997, 1/1, pp. 15-22.

Blanchon H. (1994) *Perspectives of DBMT for monolingual authors on the basis of LIDIA-1, an implemented mockup*. Proc. of 15th International Conference on Computational Linguistics, COLING-94, 5-9 Aug. 1994, 1/2, pp. 115—119.

Boitet C., Guillaume P. & Quézel-Ambrunaz M. (1982) *ARIANE-78, an integrated environment for automated translation and human revision*. Proc. of COLING-82, Prague, July 1982, North-Holland, Ling. series 47, pp. 19—27.

Boitet C. (1994) *Dialogue-Based MT and self-explaining documents as an alternative to MAHT and MT of controlled languages*. Proc. of Machine Translation 10 Years On, 11-14 Nov. 1994, Cranfield University Press, pp. 22.21—29.

Boitet C. & Blanchon H. (1994) *Multilingual Dialogue-Based MT for Monolingual Authors: the LIDIA Project and a First Mockup*. Machine Translation, Vol. 9, N° 2, pp. 99—132.

Boitet C. (1997) *GETA's MT methodology and its current development towards personal networking communication and speech translation in the context of the UNL and C-STAR projects*. Proc. of PACLING-97, Ohme, 2-5 September 1997, Meisei University, pp. 23-57.

Boitet C., Réd. (1982) *"DSE-1"— Le point sur ARIANE-78 début 1982*. Contrat ADI/CAP-Sogeti/Champollion (3 vol.), GETA, Grenoble, janvier 1982, 616 p.

Brown R. D. (1989) *Augmentation*. (Machine Translation), Vol., N° 4, pp. 1299-1347.

Ducrot J.-M. (1982) *TITUS IV*. In *Information research in Europe. Proc. of the EURIM 5 conf. (Versailles)*, edited by Taylor P. J., London, ASLIB.

Kay M. (1973) *The MIND system*. In *Courant Computer Science Symposium 8: Natural Language Processing*, edited by Rustin R., New York, Algorithmics Press, Inc., pp. 155-188.

Lafourcade M. (2001) *Lexical sorting and lexical transfer by conceptual vectors*. Proc. of MMA'01, 29-31/1/01, SigMatics & NII, Tokyo, 10 p.

Lafourcade M. & Prince V. (2001) *Synonymies et vecteurs conceptuels*. Proc.,, 29-31/1/01, SigMatics & NII, Tokyo, 10 p.

Maruyama H., Watanabe H. & Ogino S. (1990) *An Interactive Japanese Parser for Machine Translation*. Proc. of COLING-90, 20-25 août 1990, ACL, 2/3, pp. 257-262.

Melby A. K., Smith M. R. & Peterson J. (1980) *ITS : An Interactive Translation System*. Proc. of COLING-80, Tokyo, 30/9-4/10/80, pp. 424—429.

Moneimne W. (1989) *TAO vers l'arabe. Spécification d'une génération standard de l'arabe. Réalisation d'un prototype anglais-arabe à partir d'un analyseur existant*. Nouvelle thèse, UJF.

Nirenburg S. & al. (1989) *KBMT-89 Project Report*., Center for Machine Translation, Carnegie Mellon University, Pittsburg, April 1989.

Nyberg E. H. & Mitamura T. (1992) *The KANT system: Fast, Accurate, High-Quality Translation in Practical Domains*. Proc. of COLING-92, 23-28 July 92, ACL, 3/4, pp. 1069—1073.

Sérasset G. & Boitet C. (2000) *On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter*. Proc. of COLING-2000, Saarbrücken, 31/7—3/8/2000, ACL, 7 p.

Slocum J. (1984) *METAL: the LRC Machine Translation system*. In *Machine Translation today: the state of the art (Proc. third Lugano Tutorial, 2–7 April 1984)*, edited by King M., Edinburgh University Press (1987).

Wehrli E. (1992) *The IPS System*. Proc. of COLING-92, 23-28 July 1992, 3/4, pp. 870-874.

A Platform for Experimenting UNL (Universal Networking Language)

Wang-Ju TSAI

GETA, CLIPS-IMAG
BP53, F-38041 Grenoble cedex 09 France
Wang-Ju.Tsai@imag.fr

Abstract

We introduce in this article an integrated environment, which provides the initiation, information, validation, experimentation, and research on UNL. This platform is based on a web site, which means any user can have access to it from anywhere. Also we propose an XML form of UNL document as the base of future implementation of UNL on the Internet.

1 Introduction

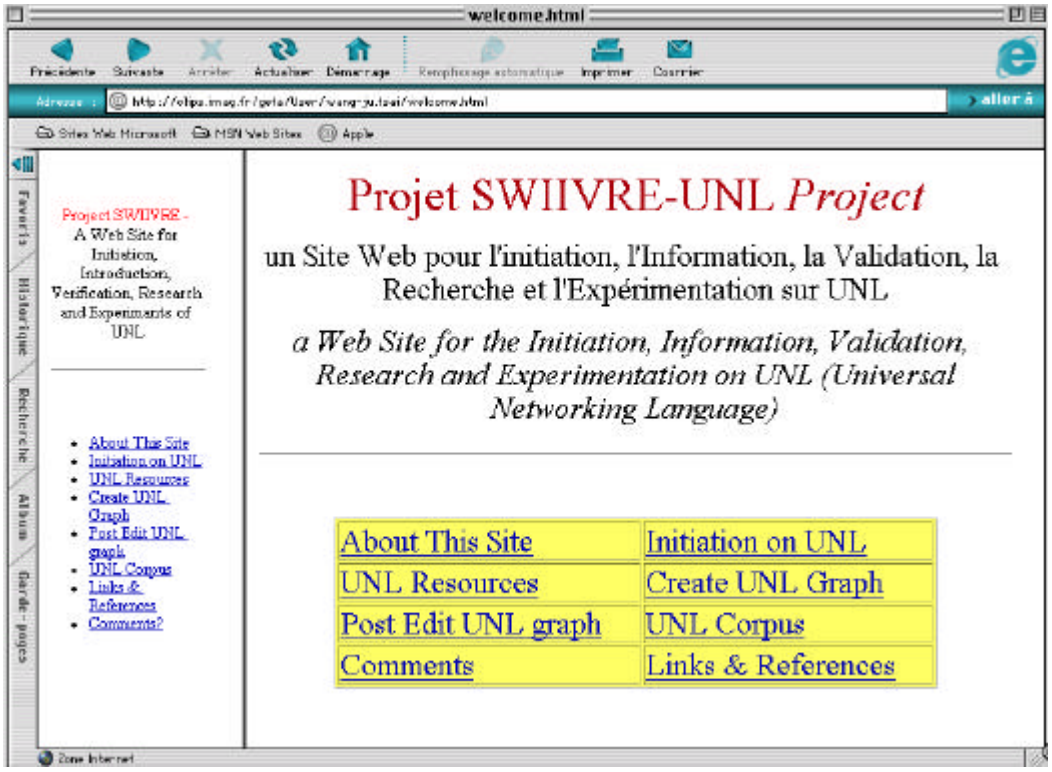
Since proposed 5 years ago, UNL project has attracted 16 international teams to join and is regarded as a very promising semantic Interlingua for knowledge representation on the Internet. The articles and applications of UNL have been found in many domain, such as: machine translation, information retrieval, multilingual document generation, ..etc. Now we can find on the Internet not only the web sites of UNL language centres but also some discussions. The applications to facilitate the usage of UNL have been produced as well. Now we see the need to create a platform to integrate these applications also to introduce UNL to new ordinary users. We create this platform on a web site SWIIVRE (http://www-clips.imag.fr/geta/User/wang-ju.tsai/welcome.html), which has several goals: for the initiation, information, verification, research, and experimentation of UNL. And since this platform is based on a web site, any user from anywhere can have access to it.

2 Introduction of the site SWIIVRE

In Appendix (I) we list all the resources accessible for UNL society members from internet. We can find out that most of the LC's connect vertically to UNL Centre but the horizontal connection among LC's is not enough, which means any user who wants to try the multilingualism of UNL will feel frustrated, since he will need to spend a lot of time try out every LC to know what service he can get.

The main purpose of this site is rather to integrate the current UNL applications and complete the services of Language Centres', when the function is available on a Language Centre, we simply provide the link to it, we also produce some applications to integrate or provide new functions, which all serve to facilitate the usage of UNL. Also we collect the useful information and publications on UNL, the web site is updated regularly. Lastly, by collecting the useful information and recording the related data, this site finally can serve as an evaluation of the performance of UNL community.

Here we show the welcome page of this site:



The following is the introduction to each link on the welcome page:

2.1 About This Site:

This page provides the introduction, why and how this site exists, the site log and current status of this site, also the new projects to come on this site, lastly all the recent activities of UNL community. When clicked, a news flesh will also show the most recent UNL activities and the new updates on this site. In the future, we think we will at least UNL-ise this page to demonstrate the multilingualism of UNL.

2.2 Initiation on UNL:

This page is to help users to take a first step in UNL, understand how UNL works. We first provide a copy of most recent UNL specifications, for the moment only Spanish Centre has prepared a "multilingual interactive page" can serve as the tutorial and give examples to each UNL relations, thus we put a link to this page. When UNL becomes more well known, there will be more and more tutorials for beginners in the future. Or we might finally create an graphical interface for user to manipulate and show the spirit of UNL. We would also like to introduce the XML-UNL document here. We put an example of XML-UNL document here and with the help of XSLT, we can create the same effect like UNL browser, then the users can choose to read the document in the language they wish. We will explain later in the article why we want to XML-ise a UNL document.

2.3 UNL Resources:

This page provides all the UNL<->NL deconverters / enconverters, dictionaries that are accessible on the Internet. Some deconverters accept the deconversion of one single UW (Universal Word), in this case they can serve as the UNL-NL dictionaries. We can simply add some scripts in our site to help users to access these deconverters as if they are accessing dictionaries. In the future, the status report of each server will be added; we hope we can provide "UNL daily bulletin" to report the updates and status of each server. Currently only French server report can be seen. To complete the services, we developed a "multilingual simultaneous deconverter" (Preedarat 2001), which can handle several deconversions at one time. Users can click on the language versions they want as output, the program will contact these servers at once, thus they don't need to do the deconversions one by one, and they can experience the automatic multilingual generation.

2.4 Create UNL Graph:

Since ordinary users are not able to write UNL graph without being trained, to help users create UNL graph will be an important function to develop. In this page we collect the links to accessible UNL editors, including editor for professional writers or for beginners. We have put a link to our "Basic UNL graph editor" (Preedarat 2001), which is implemented by

using a similar XML-UNL format and XSL transformation. The users can manipulate the UNL graph represented in tree-like structure, and save the result in XML format. We also put a link to the "interactive multilingual page" of Spanish Language Centre, here users can manipulate the UNL graph by the options provided, actually users can already generate many sentences based on these examples.

2.5 Post Edit UNL Graph:

This function is still under development. Our idea is to provide the users the possibility to correct the UNL document after it is deconverted. It provides ordinary users with the ability to correct the faults in the UNL graph and improve the quality of graph.

2.6 UNL corpus:

We collect all the UNL corpora here, and also we are currently working on designing a data base to store these corpora thus to facilitate the further exploitation or calculation. We can finally design an interface to allow users to upload the corpora in different forms, or produce the forms they desire. In appendix (II) is the first statistics we made on the corpus FB2004.

2.7 Comments:

To sends your comments to us.

2.8 Links & References:

We collect all the links to UNL Centre, Language Centres, articles, papers, discussion of UNL, and users can trigger the search engines here to find more information about UNL when they want.

3.XML-UNL document

The applications compatible to XML have been increasing a lot and XML can replace HTML as the next norm of a web-based document. And from an XML form, we can further produce other form, exchange or integrate the existing data easily. It would thus be reasonable to XML-ise the UNL document. We would like to propose here an XML form of UNL document as in Appendix (III). We created this DTD according to the UNL specification Version 3 Edition 1 (20/02/2002). Based on this DTD, we can create the UNL document in XML form, with an XSL Transformation we can produce the same effect as an UNL browser. Further more, we can easily expand this DTD to enable the XML-UNL document to register all the modifications and corrections on a UNL document, this can be very useful in our post-edition project.

Appendix (I)

4 Conclusion

We have made the first step in the integration of all the UNL components under a website. Next step is to streamline the procedures between current functions and to include more services.

5 References

Boitet Ch. (2001) Four technical and organizational keys for handling more languages and improving quality (on demand) in MT, " MT-SUMMIT VIII (2001) ", Proceedings of the Workshop (Towards a Road Map for MT), p.14-21. 18/09/2001

Coch & Chevreau (2001) Interactive Multilingual Generation. Proc. CICLing-2001 (Computational Linguistics and Intelligent Text Proceeding), Mexico, Springer, pp. 239-250.

Sérasset G. and Boitet Ch. (2000) "On UNL as the future "html of the linguistic content" & the reuse of existing NLP components in UNL-related applications with the example of a UNL-French deconverter", COLING 2000, Saarbruecken, Germany 31/07-04/08, p.768-774

Sérasset G. & BOITET Ch. (1999),"UNL-French deconversion as transfer & generation from an interlingua with possible quality enhancement through offline human interaction" MT Summit 99, 13-17 september 1999, Singapore, pp 220-228.

Boitet Ch. (1999) A research perspective on how to democratize machine translation and translation aids aiming at high quality final output, Machine Translation Summit VII (1999), Singapore, 13-17/9/99

Munpyo HONG & Olivier STREITER (1998) "Overcoming the Language Barriers in the Web: The UNL-Approach" , in 11.Jahrestagung der Gesellschaft für Linguistische Datenverarbeitung (GLDV'99), 1999, Frankfurt am Main.

Preedarat JITKUE (2001) Participation au projet SWIIVRE-UNL et première version d'un environnement web de déconversion multilingue et d'éditeur UNL de base, report de stage de Maîtrise Informatique Université Joseph Fourier – Grenoble 28/05-31/08

The resources accessible at each LC for UNL society members

	Enco	Deco	Dico	Introduction of UNL system	Linked by UNLC	remarks
Arabic	√	√	√	Arabic	√	
Chinese		√		English	√	
French		√				
Indonesian				Indonesian		
Italian		√		Italian	√	
Russian		√	√	English		
Spanish		√		English Spanish	√	Tutorials/Interactive Page/ Document Repository
Thai				Thai	√	
UNLC			√	English		UNL specs/ development modules

Appendix (II) Some Statistics about FB2004 Corpus

Corpus Name : FB2004
Original Language : English
Other available versions : French, Spanish, Italian, Russian, Hindi, UNL
No. of Sentences : 122
No. of Words : 2799
No. of Relations in UNL: 1519

Part I. The relation count

Relaion	**Outside scope**	**In scope**	**TOTAL**	**Relation**	**Outside Scope**	**In scope**	**TOTAL**
AGT	**66**	**10**	**76**	SEQ	0	0	0
AOJ	**64**	**37**	**101**	FMT	5	0	5
OBJ	**225**	**89**	**314**	FRM	6	3	9
AND	**63**	**120**	**183**	PLF	0	0	0
OR	26	3	29	SRC	2	0	2
BAS	2	2	4	GOL	17	7	24
CAG	0	0	0	PLT	1	0	1
CAO	0	0	0	TO	5	1	6
COB	1	1	2	INS	0	0	0
PTN	4	1	5	**MAN**	**49**	**17**	**66**
BEN	7	5	12	MET	10	3	13
PUR	28	1	29	PER	0	0	0
CNT	22	6	28	QUA	12	5	17
MOD	**263**	**186**	**439**	PLC	17	3	20
NAM	21	15	36	SCN	13	5	18
POF	5	2	7	TMF	2	0	2
POS	17	8	25	TMT	0	1	1
CON	2	0	2	VIA	1	0	1
RSN	1	0	1	DUR	5	4	9
COO	4	2	6	TIM	20	5	25

Total no. of relations							**1519**

Remarks:
a.)The 6 most frequently used relations are marked in bold type. The result is not surprising, since these relations have either an important or a broad usage. MAN and AGT's usage are frequent though straight forward. Besides its own static verb and copula usage, AOJ also shares part of adjective-noun relation, otherwise the frequency of MOD will be even higher.
b.)AND relation appears much more frequently within a scope, which is not surprising, since scope is used to represent the union of the similar things or ideas, and AND relation links these UW's in te scope.
c.)Some other relations' usage is not very braod, so they didn't appear.

Part II. Attribute count

(1)Time Attribute
.@past 40 / **.@present 114** / **.@future 187**
(2)Aspect Attribute
.@complete 20 / .@progress 13 / .@state 16 / else 0
(3)Reference Attribute
.@generic 9 / **.@def 659** / .@indef 79 / .@not 2 / .@ordinal 8
(4)Focus Attribute
.@entry 530 / .@topic 48 / .@title 21 / else 0
(5)Attitude Attribute
.@exclamation 1 / else 0
(6)Viewpoint Attribute
.@ability 7 / .@obligation 7 / .@possibility 8 / .@should 2 / .@unexpected-consequence 2 / else 0
(7)Convention Attribute
.@pl 558 / elso 0

Remarks:
a)The original text langue is English, so the frequency of .@pl, .@def, .@indef and time attributes are among the highest. If the original language is one of those isolated languages, such as Thai, Vietnamese, Chinese, which don't provide so much information about definitiveness or time, it might be difficult to use or to decide these attributes. It's not because that the graph authors or enconverters are bad, it's simply because they can't find these informations from the text when encoding.

Appendix (III)

```
<!DOCTYPE D [
<!ELEMENT D (P+) >
<!ELEMENT P (S+)>
<!ELEMENT S (org,unl,GS+)>
<!ELEMENT org (#CDATA)>
<!ELEMENT unl (#CDATA)>
<!ELEMENT GS (#CDATA)>

<!ATTLIST D dn CDATA #REQUIRED
            on CDATA #REQUIRED
            did CDATA #IMPLIED
            dt CDATA #IMPLIED
            mid CDTAT #IMPLIED>
<!ATTLIST P number CDATA #REQUIRED>
<!ATTLIST S number CDATA #REQUIRED>
```

```
<!ATTLIST org lang CDATA #REQUIRED
                        code CDATA #IMPLIED >
<!ATTLIST unl sn CDATA #IMPLIED
            pn CDATA #IMPLIED
            rel CDATA #IMPLIED
            dt CDATA #IMPLIED
            mid CDTAT #IMPLIED>
<!ATTLIST GS lang CDATA #REQUIRED
            code CDATA #IMPLIED
            sn CDATA #IMPLIED
            pn CDATA #IMPLIED
            rel CDATA #IMPLIED
            dt CDATA #IMPLIED
            mid CDTAT #IMPLIED>

]>

<!-- GS = generated sentence -->
<!-- dn = document name -->
<!-- on = owner name -->
<!-- did = document id -->
<!-- dt = date -->
<!-- mid = mail address -->
<!-- lang = lang tag -->
<!-- code = character code name -->
<!-- sn = system name -->
<!-- pn = post editor name -->
<!-- rel = reliability-->
]>
```

UCL – UNIVERSAL COMMUNICATION LANGUAGE

Carlos A. Estombelo Montesco
Dilvan de Abreu Moreira
Universidade de São Paulo, Instituto de Ciências Matemáticas e de Computação
Av. do Trabalhador São-Carlense, 400 - Centro - Cx. Postal 668 São Carlos - SP - Brazil CEP 13560-970
{cestombe, dilvan}@icmc.sc.usp.br

Abstract

For successful cooperation to occur between agents they have to be able to communicate among themselves. To enable this communication an Agent Communication Language (ACL) is required. Messages coded in an ACL should adequately express their meaning from a semantic point of view. The Universal Communication Language (UCL) can fulfill the role of an ACL and, at the same time, be convertible to and from a natural language. UCL design is concerned with the description of message structures, their underlining semantic context and the support for protocols for agent interaction. The key point about UCL is that the language can be used not only for communication among software agents but among humans too. This is possible because UCL is derived from the Universal Network Language (UNL), a language created to allow communication among people using different languages. UCL was defined using the Extended Markup Language (XML) to make it easier to integrate into the Internet. In addition, an enconverter-deconverter software prototype was written to serve as a tool for testing and experimenting with the language specifications.

INTRODUCTION

The technology of software agents can be an interesting tool for the creation of new models for complex software systems. In the project of software agents, many of the traditional techniques of artificial intelligence can be mixed with techniques from the field of distributed computer systems, theories about negotiation and theories about working teams (Dignum, 2000). Software agents are basically designed to cooperate (either with others or with humans) in a seemingly intelligent way. But for cooperation to occur a communication language is necessary.

What does it mean to be able to communicate with someone? Simplifying it, useful communication requires shared knowledge. While this includes knowledge of language, words and syntactic structures, meaningful communication is even more focused on knowledge about a problem to be solved. To interact with a florist you need some knowledge of flowers.

The widespread use of the Word Wide Web (WWW) and the growing Internet facilities has sparked enormous interest in improving the way people communicate using computers. To date, communication among software agents and humans has been done under limited conditions: communication is reduced to basic information exchange, ignoring the richness and flexibility implied by human language.

However to deal with any human language would be very difficult. To solve this problem, communication systems can use an Agent Communication Language (ACL) based in a simplified form of human language, which could be converted from and to a natural language.

OBJECTIVES

The main objective of this work is the specification of a new ACL, called UCL - Universal Communication Language, that focus on the specification of the semantic model and structure of the messages it represents. It also adds support for message transmission over the Internet and can be translated into or generated from natural language (English or other languages).

UCL is derived from the Universal Network Language (UNL) (Ushida et al., 1999) and implemented using the language XML (Extensible Markup Language) (Connolly, 2000). XML is a W3C (World Wide Web Consortium) standard language, like HTML, this means an easy integration with the Internet.

Another goal of this paper is to show a working UCL enconverter-deconverter prototype using the tool Thought Treasure and its associated ontology.

COMMUNICATION AMONG AGENTS

In the communication process among agents, it is indispensable an appropriate understanding of what will be communicated through the exchange of messages. A good representation of the knowledge domain, shared by the agents, can collaborate for a better understanding of the context where a message exchange takes place. As a consequence, it is important to explore concept classifications and their hierarchical structures for knowledge domain representation. The concepts in the knowledge domain have to be shared by the agents exchanging messages and be reusable in more than one context.

The specification of an ACL has to deal with the description of the message structure, his semantic model and the interaction protocols (Mamadou, 2000):

- The message format defines the communicative acts primitives and the parameters of the message (as sender, receiver, etc.). The message content describes facts, actions, or objects in a content language (KIF, Prolog, etc).

- The semantic model of an ACL should allow for messages with a concise meaning and no ambiguity.

- The interaction protocols are projected to facilitate the communication among agents. Protocols are optional, but, in case they are used, the communication among agents should be consistent with the chosen protocol.

ONTOLOGIES FOR COMMUNICATION

'Ontology' is a term used to refer to the common sense of some domain of interest. The ontology can be used as a uniform framework to solve communication problems.

An ontology necessarily links or includes some type of "general vision" regarding a certain domain. This "general vision" is frequently conceived as a group of concepts (for example: entities, attributes, processes), their definitions and their interrelations. That is called a conceptualization.

A conceptualization can be concretely implemented, for example, in a software component, or it can be abstract, being the implied concepts of a person. The use adopted in this work is that ontology is an explicit idea, or a representation (of some part) of a conceptualization.

An explicit ontology can take a variety of forms, but necessarily they will include a vocabulary of terms and some specification of their meanings (for example: definitions).

The level of formality for a vocabulary varies considerably. This variation can be shown in the following four points of view:

- Highly informal: expressed freely in natural language.
- Semi-informal: expressed in a restricted form and structure in natural language. Larger clarity for ambiguity reduction.
- Semi-formal: expressed in an artificial language defined formally.
- Strictly formal: defined meticulously with formal semantics, theorems and proofs.

A shared ontology is necessary for communication between two agents. Unfortunately UNL does not have a public available ontology. For this reason, the ontology embedded in the tool Thought Treasure was used to implement the enconverter-deconverter prototype.

THE TOOL *THOUGHT TREASURE* (TT)

This is a powerful tool for processing natural language, developed by Erik T. Mueller (1998). It is capable of interpreting natural language, as well as extending its ontology-based knowledge base. TT has a compiler for natural language that allows it to extract information of sentences.

TT has a database with 25,000 concepts organized in a hierarchical way. For example, Evian is a flat-water type, which is a drinking-water type, which is a food type and so on.

Each concept has one or more word translations what forms a total of 55,000 words and sentences of the English and French language. For instance, as it is observed in the Figure 1, the association with the concept *food* in the English language are the words *food* and *foodstuffs* and in French *aliment* and *nourriture* (among others).

In addition, *ThoughtTreasure* has approximately 50,000 assertions related to concepts such as: a *green-pea* is a *seed-vegetable*, a *green-pea* is *green*, the *grean-pea* is part of *pod-of-peas*, and *pod-of-peas* is found usually at a store of foodstuffs.

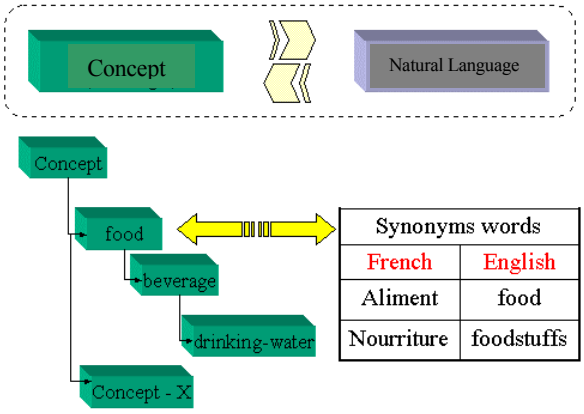


Figure 1: Association of the ontology with a natural language

UCL - UNIVERSAL COMMUNICATION LANGUAGE

The language UCL represents information in the same way UNL does, but using syntax based in XML. XML is a meta-language, a simplified form of SGML, which developers can use to create new languages based in tag elements. The new tags, created to represent the new language elements, can be described in a special file called DTD (Document Type Definition). UNL is a formal language for representing the meaning of natural language sentences and exchange information over a network. Information that is written in a native natural language is "enconverted" into UNL and stored in a server. This information can be "deconverted" into other languages to be read by each native reader. Thus, UNL can play the role of an interface between different human languages to exchange information.

UNL represents information expressed in sentences as a set of relations between meanings, expressed by words, and a

syntactic structure that makes up the sentence. The vocabulary of UNL consist of:

- Universal Words (UWs), to represent word meaning.
- Relation Labels, to represent relationships between UWs
- Attribute Labels, to express further definitions or additional information for the UWs that appear in a sentence.

In UNL, the information about a sentence includes its meaning, tense and aspect information (how the speaker grasp the event), intention of utterance, speaker's feeling or judgment upon contents, and sentence structure. In the language, the meaning of a sentence is represented by the description of the relationships between UWs and its structure is described by attaching attribute labels to these UWs.

UCL GOALS

The language UCL is to be used for high-level communication among agents through the exchange of messages. Some characteristics that guided the definition of the language were:

- To aid the communication involving agents giving importance to the semantics of the message;
- To be easy to use;
- To facilitate its integration into the Internet environment writing it in XML (*Extensible Markup Language*)

The language UCL represents the information in sentences (that can form messages) that involves a syntactic structure with a group of concepts, relationships and attributes similar to UNL:

- Universal Words (UW),
- Relationship labels,
- Attribute labels.

To define a language based in XML a specific DTD file is used. This DTD is essentially a grammar of free context, like the extended BNF form (*Backus Naur Form*) used to describe computer languages (Grosof & Labrou, 1999).

As in UNL, a Universal Word (UW) is the minimum unit that represents a concept, which denotes a specific meaning in a message. When a concept needs to be defined in more detail Relationship Labels and Attribute Labels are used. In addition, UCL uses a shared ontology, from the tool ThoughtTreasure, to add meaning to the UWs. All agents participating in a communication process should share this ontology.

In a UCL sentence, each defined UW has an identifier label (id) that is used to identify a particular concept inside a sentence. A sequence of alphanumeric characters forms this labels. The label `head` corresponds to the place where the name of the concept will be defined. The concepts used are always related to the ontology being used (ThoughtTreasure ontology). It is at this point that a sentence in UCL is connected with the ontology for a specify knowledge domain.

In UCL messages possess a certain meaning involving concepts. This composition of concepts is represented by groups of binary relationships, which allow different relationships involving the concepts. The relationship labels used come from UNL. Figure 2 shows an English sentence and its translation to UCL.

- *UNL is a common language that would be used for network communications.*

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE sentence SYSTEM "Sentence.dtd">
<sentence>
        <uw id="uw00" head="language">
            <icl direction="to">
                <uw head="abstract thing"/>
            </icl>
            <tense attribute="present"/>
            <focus attribute="entry"/>
        </uw>
        <uw id="uw01" head="UNL">
            <icl direction="to">
                <uw head="language"/>
            </icl>
            <focus attribute="topic"/>
        </uw>
        <uw id="uw02" head="common">
            <aoj direction="to">
                <uw head="thing"/>
            </aoj>
        </uw>
        <uw id="uw03" head="use">
            <icl direction="to">
                <uw head="do"/>
            </icl>
            <tense attribute="present"/>
        </uw>
        <uw id="uw04" head="language">
            < icl direction="to">
                <uw head="abstract thing"/>
            </icl>
            <tense attribute="present"/>
            <focus attribute="entry"/>
        </uw>
        <uw id="uw05" head="communication">
            < icl direction="to">
                <uw head="action">
            </icl>
            <convention attribute="pl"/>
        </uw>
        <uw id="uw06" head="network">
            <icl direction="to">
                <uw head="thing">
            </icl>
        </uw>
    <relation label="aoj" uw-id1="uw00" uw-id2="uw01"/>
    <relation label="mod" uw-id1="uw00" uw-id2="uw02"/>
    <relation label="obj" uw-id1="uw03" uw-id2="uw04"/>
    <relation label="pur" uw-id1="uw03" uw-id2="uw05"/>
    <relation label="mod" uw-id1="uw05" uw-id2="uw06"/>
</sentence>
```

Figure 2 Definition a sentence in UCL

IMPLEMENTING AN ENCONVERTER-DECONVERTER

UCL is defined in the meta-language XML, to work with it a XML parser should be used. As the enconverter-deconverter is written in the language Java, the Java API for XML Processing (JAXP) Version 1.1 from Sun, was used (other Java XML parsers could have been used).

As said before, UCL uses the ontology available on the ThoughtTreasure (TT) tool (written in C). This tool includes program libraries to manipulate concepts of the ontology, to do consultations on the net of concepts, and to analyze their hierarchy. An instance of TT can run as a server in a network and communicate with a Java program running in another process. A Java communication API is supplied with TT to handle the low level details of this communication.

The enconverter-deconverter prototype uses the Java communication API to contact a running instance of TT and use its functionality. Those include natural language treatment, ontology queries, etc. A high level Java interface was written to communicate with the TT server (through the API) and implement the high level functions needed by the prototype. This interface is called *UclLanguage*.

Figure 3 presents a diagram with the sequence of events that happens when the prototype makes use of the interface *UclLanguage* to generate UCL messages.

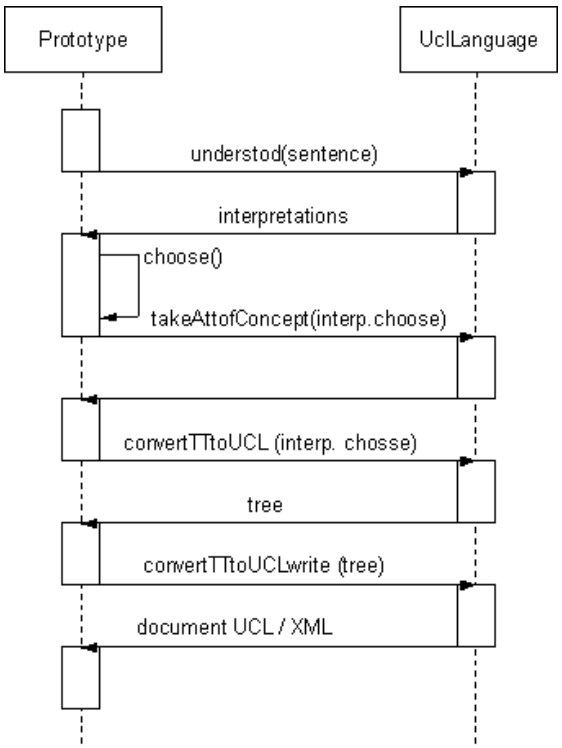


Figure 3: Diagram with the sequence of events during enconvertion.

The process begins when a user enters a natural language sentence into the prototype. The prototype calls the method *understood* of the interface *UclLanguage*. The natural language sentence is interpreted (using TT) and some possible semantic interpretations are returned. The user chooses the most appropriate interpretation. The chosen interpretation is converted to TT's format (method *takeAttofConcept*) and then to UCL format (method *convertTTtoUCLwrite*). The UCL format can be shown on the screen or saved in a file.

The reverse process, transform a UCL message in natural language is easier. The prototype uses the method *deconvertUCLtoTT* to convert the UCL message in a list of TT concepts. Then it uses the method *deconverterTTtoLN* to transform this list of concepts in a natural language sentence, which represents the original UCL message.

Example : Monkey eats bananas

```
======= Input Natural Language ==========
Example: Monkey eats bananas.

============ Choose Option ==============
<0>An ape eats a banana.

Option: 0
============ Message UCL  ===============
<?xml version="1.0" encoding="UTF-8"?>

<sentence>
  <uw id="uw2" head="present-indicative">
    <icl direction="to">
      <uw head="present-tense" />
    </icl>
    <focus attribute="entry" />
  </uw>
  <uw id="uw4" head="eat">
    <icl direction="to">
      <uw head="ingest" />
    </icl>
  </uw>
  <uw id="uw5" head="ape">
    <icl direction="to">
      <uw head="mammal" />
    </icl>
  </uw>
  <uw id="uw7" head="banane">
    <icl direction="to">
      <uw head="fruit-tropical" />
    </icl>
  </uw>
  <relation id="uw1" label="icl" id1="uw2" id2="uw6" />
  <relation id="uw6" label="icl" id1="uw3" id2="uw7" />
  <relation id="uw3" label="agt" id1="uw4" id2="uw5" />
</sentence>

======== Deconverter Message UCL  ===========
=>Debug : [present-indicative [eat ape banane ]]

English: An ape eats a banana.
French : Un singe croque la banane.
```

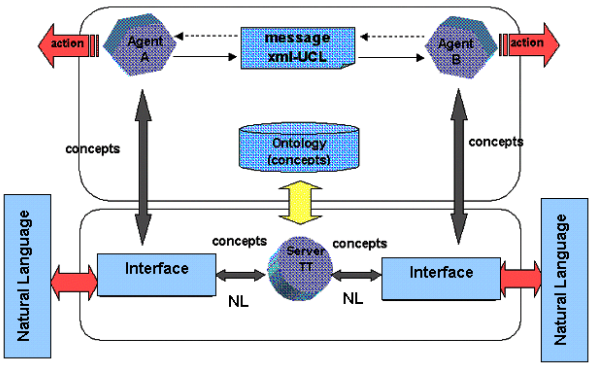


Figure 4: Architecture of a system that uses the language UCL

Figure 4 illustrates the use of UCL (using one TT server) in the communication process between two software agents.

CONCLUSIONS

The definition of the Universal Communication Language (UCL) includes all theoretical concepts of the Universal Networking Language (UNL). This was done to preserve the representative power of this language. The Web community currently regards XML as an important step toward semantic integration. Developing the language UCL using XML yielded some important benefits. The most important is the reuse of existing tools for creating, transforming, and parsing UCL documents.

The UCL enconverter-deconverter prototype shows the need for a shared ontology for the implementation of a successful enconverter-deconverter. UCL was developed to be used as a rich Agent Communication Language (ACL), which would make it easier for humans to communicate with and program software agents (using multiple natural languages). But UCL can be used in the same role as UNL.

The prototype also points out the need for an open shared ontology for UNL. UNL relation and attributes labels have some ontological knowledge already embedded in them. This makes impossible to map all possible UNL (and consequently UCL) constructs into ThoughtTreasure ontology based representation. The prototype can not be expanded into a full featured UCL enconverter-deconverter. For the time being this prototype is good enough to help the development of a prototype UCL interpreter for software agents.

The full power of the approaching of using UCL as an ACL and programming tool for software agents will only be realized, when an open shared ontology for UNL and enconverters-deconverters for many natural languages (using this shared ontology), are available. One will be able to program a software agent using his own native language and share this program with many other people, which will see and interact with the program in their own native languages.

Finally, UCL is still a proposal, but we hope that others in the Web community will help to shape its final format.

ACKNOWLEDGEMENTS

The authors would like to thanks the CNPq – National Council for Research in Brazil for the financial support for this work.

REFERENCES

Connolly, D. (2000). *Extensible Markup Language (XML)*.February 2000. Available on-line: http://www.w3.org/XML/

Dignum, Frank; Greaves, Mark. (ed.) (2000). *Issues in agent communication*. – (Lecture notes in computer science; Vol 1916: Lecture notes in artificial intelligence) Berlin; Heidelberg; New York; Barcelona; Hong Kong; London; Milan; Paris; Singapore; Tokyo: Springer, 2000.

Grosof, Benjamin N.; Labrou, Yannis (1999). *An Approach to using XML and a Rule-based Content Language with an agent communication Language*. IBM Research Report. RC 21491 (96965), 28 May 1999, Available on-line: http://www.research.ibm.com

Mamadou, T. K.; Shimazu, A.; Tatsuo, N. (2000). *The State of the Art in Agent Communication Languages*. Japan Advanced Institute of Science and Technology., Japan, 1999.

Mueller, Erik T. (1998). *Natural Language processing with ThoughtTreasure*. New York: Signiform. Also available on-line: http://www.signiform.com/tt/book/

Ushida, H.; Zhu, M.; Senta, T.D. (1999). *The UNL a Gift for a Millennium*. UNU/IAS, November 1999, ISBN:4-906686-06-0. Also available on-line: http://www.unl.ias.unu.edu/publications/index.htm

UNL, Challenges and Misunderstandings. Some answers

Edmundo Tovar*, Jesús Cardeñosa*

* Validation and Business Applications Research Group
Artificial Intelligence Department
Universidad Politécnica de Madrid
Campus de Montegancedo, s/n, Boadilla del Monte, 28660 Madrid, Spain
etovar@fi.upm.es, carde@fi.upm.es

Abstract

The UNL, either as language or as a system, is not well known due to several reasons. At this moment, UNL is not only the name of a language for computers aiming at supporting multililingual services in Internet. It is also a system, with a defined architecture and a wide panorama of applications and possibilities to support business, institutional and educational applications, all of them going beyond the linguistic barriers. Nobody doubts about the possibilities of this type of system. However, this system today supported by an organization based on a Foundation (placed in Geneva and created only to support this UN initiative) needs the collaboration and financial help of all kind of sources (UN is not financing the initiative at this moment). This is a hard task. Perhaps the more significant case about the impossibility to reach financial support for this initiative and also about the different research and application issues has been the high number of project proposals made for different Call for Proposals of the EC in the area of Econtent and IST, as well. All of them have been rejected, thus creating a wall for the development of the systems of the European languages, that are actually the more advanced within the UNL Consortium. This paper will try to analyze the different evaluations made for the various proposals in order to clear up the real state of this system and also the reasons of the low level of knowledge about this important initiative. Our goal then is to examine the opinions of the EC evaluators giving in the paper the adequate answers to them even by the side of the UNL Consortium too. Dissemination policies and internal organization of the UNL will be clarified for a better analysis in the immediate future.

1. The New Organization for the UNL Program

The Universal Networking Language (UNL in the following) arises by the initiative from the Institute of Advanced Studies (UNU/IAS), of the United Nations University in 1996. The mission of the UNL Programme (UNDL, 2002a) is to develop and promote a multilingual communication platform for Internet, with the purpose of enabling all peoples to share information and knowledge in their native language. The IAS first selected a group of institutions from fifteen countries that were in charge of developing the modules for each corresponding language. The milestones and partial results are revised in yearly meetings. In Brussels, 1999 it was presented more than a project. It was presented an Organization for the future. But the most important fact is perhaps that, after three years of development work of the participating teams, the UNL Language specifications were made public. That is, anyone can develop, potentially, for public or commercial use, components and systems integrable with the UNL system.

The current organization of the UNL Programme (UNLP) has been put on the hands of the recently created UNDL Foundation (UNDL, 2002b), a non-profit international organization created to continue the research and development initiated by the UNU/IAS in Information Technologies, in particular, in the field of interlingua communication, UNL and its applications in all areas of human knowledge and activities. The UNDL deploys the UNL facilities to assist all peoples in bridging the digital and cultural divide, in accordance with the principles and objectives of the United Nations and its Members States.

The new organization of the UNL Programme has a network of persons and institutions under the direction and sponsorship of the UNDL Foundation. The UNLP network consists of the UNL Center, the Language Centers (LC) of each language and other elements as UNL National Units, Permanent Committees and ad-hoc working groups.

The UNL Center has the overall responsibility (UNDL, 2002c) for promoting and coordinating the UNLP. In this organizational structure the LC are considered as an expansion of the UNL Center for a given language and is responsible for the research, development and maintenance of the UNL System in that language. For all these tasks, each LC must, as opposed to the past, procure the necessary financial means for the support of the LC and UNLP operations (article VII.4.e, UNLP Statute). In this paper we explain the efforts addressed by European LC, in the last years trying to get funding from the European Commission (EU) Research Programmes.

2. History of the proposals presented

European LCs have been involved in the last years in the preparation of a lot of proposals in different EU Programmes including UNL as technological base. All of them failed. We think that there are no only specific reasons to reject every proposal. This total coincidence may can be explained additionally by global reasons related to the perception and knowledge of the UNL technology by the EC evaluators as the political and technical actions of the UNL Organization.

Globally speaking, we have collected information of 8 project proposals to the EC programmes with the UNL technology as basis; all of them rejected. However, we are going to focus in this analysis over the last 5 proposals because two reasons: they are the last ones, and we, this research Group as support of the Spanish LC, has had a very active participation in them. The three other proposals were made by consortiums coordinated by companies. In all cases the application of the UNL technology fit very well with the system proposed. These proposals were named HEREIN-ML, and AQUITRA with applications for the International Office of Water. Both proposals were thought for the multilingual support of the

European Heritage Project (www.european-heritage.net). Today the European languages are covering part of the multilingual support with the UNL technology with a high degree of precarity and with direct investment in resources from the Spanish government and the Spanish Language Centre, using resources of free collaborators. However in these proposals the UNL was not accepted at all as alternative for the multilingual support.

The 5 proposals considered (see table 1), in chronological order, will be described through the summary of goals and the action lines of the Programme in which the proposal has been developed.

Date	Identifier	Programme
July 00	QANET 3312	Econtent Preparatory actions
October 00	QANET IST2000-28568	RTD Proposal, IST
January 01	LINGWEB 30130	RTD Proposal, IST
July 01	EU-UNL: 22045	Econtent Programme. Demonstration Project
November 01	MULTIDOC: 34702	RTD Proposal. IST

Table 1: A view of the proposals presented

QANET-3312(econtent): Quality Assurance Procedures for an Internet Multilingual System

Summary of goals: This proposal aimed to make a selection of resources for testing and measures of coverage together with the definition of a common lexicon of general purpose, to address the definition and construction of tools to verify and maintain their resources, to test the cross lingual tools and resources, and to generate the Quality Manuals according to ISO9001 and validation procedures to support the implantation of the Quality System in the UNL Programme.

Action lines of the RTD Programme:

Action Line 3 This proposal fits with this action because this action line mentions explicitly the problems derived from linguistic diversity and from the services to be supplied with an effective infrastructure in order to sooth the consequences of the growing number of languages in Europe and the increase of institutional and commercial relations.

QANET (RTD) IST2000-28568: Quality Assurance System for Internet Multilingual Applications

Summary of goals: This proposal aimed to define the Quality System of the UNL Program of the United Nations to overcome the linguistic barriers in Internet. The definition of the Quality System required the generation of resources (lexicon corpora addressed to this task) and methods

to evaluate the future existing systems integrated with to the UNL system. For that two industrial applications would have been developed. One UNL editor based on existing analysers (demonstrating so the integrability of this approach and the capability to reuse existing systems) and also a target language Generator completely developed from the current public specifications of the UNL language that has real intention to be exploited after the project.

Action lines of the RTD Programme:

Action Line 3.3.3 (2000 WorkProgramme). Multilingual communication services and appliances.

This proposal fits with this key action. It is addressed concretely to services and appliances independently of the language of the user. The core of the system (the UNL System) has been developed to the wider multilingual capabilities system built until this moment. In fact, UNL forms the way to access from any language and generate any other language in the world.

LINGWEB (IST2000-30130): Multilingual Web-site Deployment based on an Interlingua Technology.

Summary of goals: We aimed to deploy the UNL technology that up to that moment we had had several basic components to create the UNL network addressed to support the multilinguism in Internet. Concretely we aimed to obtain a complete implantation of multilingual services in the user web-site based on UNL technology. The user would be the Organisation Barcelona2004. Besides, we would define the materials and contents to support international training courses including the testing and adaptation of the tools involved in the maintenance of the system and the UNL coding like part of the Technology Transfer process. These tools should be adapted for any other uses out of the Language Centers environment.

Action lines of the RTD Programme:

Action Line 3.3.4 (2000WorkProgramme). Trials in multilingual e-service and ecommerce. The proposal is conceived as a Trial. Technology users are at the same time suppliers of the contents to offer an information service in a highly multilingual environment. What underlies the concern of this proposal is the effective evaluation of a new and open technology in a real environment, and to define accurately Technology Transference tasks. EU-UNL was presented as a Demonstration project to prove the economic viability of the service on a specific field.

EU-UNL: European Use of UNL

Summary of goals: EU-UNL focuses in the implementation of the UNL technology for multilingual dissemination of contents on the field of the quality of water and on the field of tourism. The project includes the user corporate web implementation of a multilingual document generation system and definition of procedures for technology transfer, planning and implementation of measures and cost evaluation, as well as the complete set of materials to assure the necessary support for internal training in the organizations. EU-UNL constitutes just a case-study for the viability of the implementation of this technology that can be extended to different languages and different fields.

Action Line of the econtent Programme: Action 2. Enhancing content production in a multilingual and multicultural environment. The overall goal in this action is to investigate and experiment with new strategies, partnerships and solutions for designing e-content products and services that can accommodate local languages and cultural conventions. EU-UNL aimed to demonstrate the capabilities of the UNL technology for multilingual dissemination of e-contents as well as for introducing a new paradigm of creation and management of multilingual web-sites.

MULTIDOC-IST-34702: A system for multilingual document dissemination

Summary of goals: The goal of this proposal was the development and integration of the components for a multilingual dissemination system in the Web using the public and open UNL. This technology constituted the base of the multilingual support for the proposed application. Initially we planned to demonstrate it in a workplace/business scenario, but were equally applicable in a personal dissemination scenario. For providing multilingual functionalities to Internet publishers UNL would be embedded into their current documents. We would also provide the tools needed for processing the new multilingual documents.

Action lines of the RTD Programme:

Action Line III.3.1 (2001WorkProgramme). The Multidoc directly addresses most of the concrete objectives listed under action line III.3.1, such as the advance towards a fuller realisation of the multilingual Internet for personal development and informational purposes, wider availability and more effective production and use of multilingual information, Multilanguage design, authoring and publishing of online (web) multimedia documents, or multilingual generation.

3. The evaluation of the proposals

We aim in this section to reflect the view of the proposers and the EC evaluators for each proposal described. Before, the evaluation criteria and the process followed in the IST Programme are explained.

3.1. Evaluation criteria for the IST Programme

A number of evaluation criteria are common to all the programmes of the fifth framework programmes. Independent experts examine each eligible proposal against these criteria. The specific programme decisions provide further details of these criteria and may also provide for additional evaluation criteria that apply only to the particular programme(s) concerned. Any particular interpretations of the criteria to be used for evaluation and any weights and thresholds to be applied to the criteria are set out in the programme-specific annexes to this document and referred to in calls and all relevant supporting documentation.

For the detailed examination of proposals against the criteria set out in the rules for participation, the experts will generally provide marks and comments. In addition, the experts are asked to examine certain evaluation criteria by answering a set of questions relevant to the specifications referred to in the call. The following questions are addressed at an appropriate moment in the evaluation:

- Does the proposal address the parts of the work programme, including policy issues, open for the particular call? If the proposal is only partially in line with the call, does it have sufficient merit to be considered in its entirety or partially?
- Have relevant ethical issues been adequately taken into account in the preparation of the proposal; is the proposed research compliant with fundamental ethical principles, if relevant? Is the research proposed in line with Community policies, if relevant; have appropriate safeguards/impact assessment regarding Community policies (e.g. environment) been taken into account, where necessary?
- Does the proposal follow the requirements for presentation (notably requirements for anonymity)?

In the case of negative answers to these questions, the experts are required to provide comments to justify their answers. On the basis of the experts' remarks, the Commission reserves the right not to continue with the evaluation of any proposal which is found not to fulfill one or more of the above requirements. In clear-cut cases (for example, a proposal which addresses a research task which is not open in the particular call), a proposal may be ruled out of scope or contrary to clearly stated policy requirements at the moment that the eligibility checks are carried out.

All eligible proposals that conform to the requirements of the call are examined for their quality and relevance by the Commission assisted by external experts. Experts examine proposals and provide marks for the criteria set out below (which are drawn from the decisions on the framework programmes and the "rules for participation" decisions and grouped into five main blocks). In addition, they also provide an overall mark for each block of criteria (unless a proposal fails any thresholds – see below). Experts are required to provide comments to accompany each of their marks in a form suitable for providing feedback to the proposers. These comments must be consistent with any marks awarded.

The blocks of criteria to be applied by all programmes are as follows (EC, 2001):

Scientific/Technological quality and innovation

- The quality of the research proposed and its contribution to addressing the key scientific and technological issues for achieving the objectives of the programme and/or key action;
- The originality, degree of innovation and progress beyond the state of the art, taking into account the level of risk associated with the project;
- The adequacy of the chosen approach, methodology and work plan for achieving the scientific and technological objectives.

Community added value and contribution to EU policies

- The European dimension of the problem. The extent to which the project would contribute to solving problems at the European level and that the expected impact of carrying out the work at European level would be greater than the sum of the impacts of national projects;
- The European added value of the consortium - the need to establish a critical mass in human and financial terms and the combination of complementary expertise and resources available Europe-wide in different organisations;
- The project's contribution to the implementation or the evolution of one or more EU policies (including "horizontal" policies, such as towards SMEs, etc.) or addressing problems connected with standardisation and regulation.

Contribution to Community social objectives

- The contribution of the project to improving the quality of life and health and safety (including working conditions);

- The contribution of the project to improving employment prospects and the use and development of skills in Europe;
- The contribution of the project to preserving and/or enhancing the environment and the minimum use/conservation of natural resources.

Economic development and S&T prospects

- The possible contribution to growth, in particular the usefulness and range of applications and quality of the exploitation plans, including the credibility of the partners to carry out the exploitation activities for the RTD results arising from the proposed project and/or the wider economic impact of the project;
- The strategic impact of the proposed project and its potential to improve competitiveness and the development of applications markets for the partners and the users of the RTD results;
- The contribution to European technological progress and in particular the dissemination strategies for the expected results, choice of target groups, etc.

Resources, Partnership and Management

- The quality of the management and project approach proposed, in particular the appropriateness, clarity, consistency, efficiency and completeness of the proposed tasks, the scheduling arrangements (with milestones) and the management structure. In addition, the tools to be used for monitoring project progress, including the quality of specified indicators of impact and performance, and ensuring good communication within the project consortium;
- The quality of the partnership and involvement of users and/or other actors in the field when appropriate; in particular, the scientific/technical competence and expertise and the roles and functions within the consortium and the complementarity of the partners;
- The appropriateness of the resources - the manpower effort for each partner and task, the quality and/or level and/or type of manpower allocated, durables, consumables, travel and any other resources to be used. In addition, the resources not reflected in the budget (e.g. facilities to carry out the research and the expertise of key personnel). For this criterion, comments may be given rather than marks.

When examining proposals, experts will only apply these criteria, supplemented by any programme-specific criteria contained in the programme decision. These criteria as they apply to the particular programme may be described in greater detail in the programme-specific annex and the work programme. Experts are not be allowed to apply criteria which deviate from those set out and the programme-specific annex.

3.2. The Evaluation of the UNL Proposals

3.2.1. Evaluation Results of QANET (econtent)

The opinion of the EC Experts.

This proposal caused a good impression because, in opinion of the evaluators, showed a well documented overview of the subject, an extensive workplan, the consortium consisted of outstanding relevant experience, with a proposal well structured. However, it presented an R&D approach rather than a feasibility demonstration of econtent, as was required by the present call. For this reason the proposal fell outside the scope of the econtent call. The evaluators recommended the submission of the proposal to a more suitable EU programme.

The opinion of the EU-UNL Consortium.

We accepted the opinion of the expert evaluators.

Actions taken by the UNL partners.

We considered the evaluation of the proposal in an optimistic way. For this reason we decided to remake the proposal to be adapted to the next call of R&D IST Programme incorporating at least a company and a user (new QANET proposal).

3.2.2. Evaluation Results of QANET (RTD)

The opinion of the EC Experts.

This proposal failed to reach the threshold score on two of the criteria.

- *Scientific/technological quality and innovation.* In opinion of the EC experts the proposal did not provide a convincing integration of both aspects, quality assurance in multilingual applications and developing resources for the UNL platform. The detailed study of the state of the art in Machine Translation and NLP systems evaluation had several omissions and did not bring forward clear conclusions.
- *Economic development and S&T prospects*. A commercial partner was willing to take up the exploitation of the project results, but these were highly conditional on the success of the UNL approach. The viability of which was questionable. Likewise, the potential for commercial exploitation of Quality Assurance methodologies for Human Language Technologies is not demonstrated, and would have required a much deeper market analysis than provided in the proposal.

The opinion of the QAnet Consortium.

We proposed to develop a series of resources (corpuses and controlled dictionaries) to be produced during the project as the base of the testing of the UNL Quality System as well as to any other NLP. For this the results are not completely conditional on the success of the UNL approach. The conclusions derived from the state of the art, maybe not enough described, are that we need to produce instantiated quality models for human language technology applications (purpose of this project).

Actions taken by the UNL partners.

We decided to carry on presenting a new proposal.

3.2.3. Evaluation Results of LINGWEB

The opinion of the EC Experts.

This proposal was judged ineligible. The reasons were because:

(1) *Non-existence of technology*. Multilingual website creation technology based on UNL does not exist while it should be a prerequisite for a trial project;

(2) *Excessive resources for development*. A high level of development and integration of new components consumes more than a half of resources;
(3) *Non-study of benefits*. The benefit of the approach chosen even in terms of productivity enhancement or the impact on the management of the lifecycle of multilingual documents is not at all addressed;
(4) *Market study insufficient*. The market perspectives are not convincing despite the intention of the coordinating partner to spin-off the results.

However, as the evaluators said in their report, the idea of using UNL as an interlingua for multilingual website creation is attractive and could be reconsidered in the framework of future generation multilingual web activities.

The opinion of the EU-UNL Consortium.

In this occasion, we felt very surprised by the way of the rejection of this proposal (ineligible) and the reasons that explained this decision. We answer to every one of the arguments previously described:

(1) *Non-existence of technology*. The UNL technology was officially presented in UNL annual meetings at Brussels and Geneva previously to this proposal with attendance of representatives of the EC.
(2) *Excessive resources for development.* There is not any new component in this proposal. According to the requirements of the Call for this proposal we proposed the adaptation of resources and components already existing. For this task we planned 6 man month of the total 75 mm. The rest of the tasks are assigned to produce methodologies.
(3) *Non-study of benefits.* There is a whole workpackage (wp5) that addresses specifically the definition of metrics and methods for evaluating the technical and business performances and its associated costs.
(4) *Market study insufficient.* This is more subjective argument. We proposed several exploitation strategies based on the creation of new Language Centers, the promotion for the creation of new companies from the results obtained of some Business Plan made by the coordinator of the proposal, the expansion of the use of the UNL technology without costs to institutions, segmentation of the market uses, professional training for individuals that are working in the field of translation, the creation of a commercial version of the system at low price for individuals, forming associations for the developing of specific components and/or joint exploitation of specific contents with commercial interest, and by last, through the expansion of number of languages as priority.

In summary, we did not understand and we did not agree with this qualification of proposal "ineligible". What kind of political attitude of the Programme responsible were taken?

Actions taken by the UNL partners.

We collected the last comment of the evaluators concerning to the idea of using UNL as interlingua for multilingual website as an attractive idea and, in spite of the strong hit we received, we kept on our efforts promoting a new proposal in the econtent Programme (EU-UNL proposal).

3.2.4. Evaluation Results of EU-UNL

The opinion of the EC Experts.

This proposal was considered as an interesting approach to the development of an interlingua for the automatic translation of text. However, UNL, in opinion of the experts was not sufficiently established and proven. It bears too many risks and should probably addressed under an R&D Programme. They had serious concerns about UNL, hand-encoding and its long term viability. The overall score was 2 (fair). In brief, the evaluators appreciate good technical knowledge in consortium, and they think that based on this partnership this could be a good research project.

The opinion of the EU-UNL Consortium.

The purpose of the project is to prove the cost-effective feasibility of the integration of a well-proven translation system to a content provider deployment strategy. This would provide a big amount of information in several languages that would serve as base of the knowledge needed. By this reason, one of the main objectives of the proposal included a Methodology for the implementation of the multilingual UNL system, including the testing phase. Effectively, the basic components of the UNL system have been already developed in the latest years. Now, they need to be tested in an integrated way and in real environments to fine-tune the interrelation of every language components such as was planned in the proposal.

Actions taken by the UNL partners.

We followed the recommendations of the evaluators and we promoted a new proposal in a RTD Programme (Multidoc proposal).

3.2.5. Evaluation Results of MULTIDOC

The opinion of the EC Experts.

This proposal only failed to reach the threshold score on one of the criteria.

Scientific/technological quality and innovation. In opinion of the EC experts the innovative value is low as this approach to the translation is not new. The scientific value of the proposal rests on the merits of the technology, UNL, that is being applied. But for these experts UNL is not a well-proven translation system since it is not backed up by solid independent evidence. Thus, this proposal fails to adduce any reference in the literature in support of UNL. By other side, according to their opinions, the proposal does not contain any suggestion how the enormous linguistic complexities of the encoding process can be taken one stage beyond machine aided / validated human effort which renders the approach economically unviable on any scale.

The overall conclusion is that the project intends to employ a technology of insufficiently proven feasibility and questionably economic viability.

The opinion of the Multidoc Consortium.

We propose to use the UNL technology for representing the informal contents of web pages following the XML-compliance of document mark-up languages. It

is true that is not innovative. The innovative aspect in this project is the design and implementation of a multilingual dissemination system that covers all the steps of the publication chain: encoding of contents, generation of multilingual count parts and delivery of language specific versions to readers. The user site and the sites of the technology providers engage in a communication process involving standardized UNL-enriched documents using Internet-based communication software components.

As regards the complexity of the encoding process, in this moment several partners of the consortium have prototyped tools addressing this need.

Actions taken by the UNL partners.

We decided to take a period of reflection. We have taken a lot of man-month dedicated to the elaboration of proposals for the IST Programme without success. This is not a problem of a proposal but the perception of the UNL technology by the EC responsible.

3.2.6. Comparative Analysis of Evaluation Results

We have gathered the scores provided by the evaluators for previous proposals (see table 2). Each column corresponds to the scores obtained by each criterion, with the following meaning:

- Criterion 1: Scientific/technological quality and innovation
- Criterion 2: Community added value and contribution of EC policies
- Criterion 3: Contribution to Community social objectives
- Criterion 4: Economic development and S&T prospects
- Criterion 5: Resources, partnership and management

Proposal	Score Crit.1/3	Score Crit.2/2	Score Crit.3/0	Score Crit.4/3	Score Crit.5/2
QANET	Non numerical score. Global score = 0 (rejected)				
QANET	2	3	2	2	2
LINGWEB	Ineligible				
EU-UNL	Non numerical score. Global score = 2 (fair)				
MULTIDOC	2	3	3	3	2

Table 2: A view of the proposals evaluation

We have included in the table, together with the identifier of criterion, the threshold score required by the EC. An analysis of these results for the previous evaluations shows that the main obstacle for the approval of the proposals refers to the use of the UNL as technology (criterion of the technological quality and innovation). Evaluators do not find attractive and feasible the inclusion of this technology. However, in these proposals, the other criteria are in general well considered, issues such as the adequacy for the problem that address and its contribution to community social objectives, the fitness to the EC policies or a consortium balanced.

4. Conclusions

The initiatives described in this paper show at least two issues by the side of the European UNL LC (proposers). Firstly, proposers have shown a persistent interest to involve the EC in the success and diffusion of a technology for Multilinguality derived from the United Nations. Second, proposers have dedicated lot of resources trying to follow the recommendations of evaluators. Specifically, the Spanish Language Center was the coordinator of the first three proposals and was an active contributor to the rest. We have commented the last five proposals, but there are another three presented with the same results: HEREIN-ML (Towards a methodology for making textual information about European heritage multilingual by using UNL as metadata), AQUITRA and COACH (Company Organization for Automation Customer Help Integrated into ebusiness).

The diagnosis has been done but there are no clear causes. We can speculate with some of them.

- From the viewpoint of the EC evaluators, UNL technology is not feasible maybe because the lack of successful experiments and by the scarce presence of UNL in scientific areas of the sector.
- From the viewpoint of the proposers, we regret the absence or extension of more explanations or advices for the future, maybe at the political and strategic level in order to avoid apparent contradictions in the specific evaluations obtained.

The only view of all this information placed together is speaking by itself. All the proposers have long and intense European projects experience during the last ten years at least. On the other hand, it is not understandable why the interest of the EC in this global initiative of the UN is so low or inexistent. Europe must not be out of this initiative and some of the technical evaluations seem to be made in the best case by persons with a low level of knowledge about this initiative. The reader of this paper can extract conclusions by him/herself according the proposals, and the persistent and sometimes contradictory evaluations of all of them.

5. References

EC, 2001. *Manual of Proposal Evaluation Procedures*, IST Program, ed. 1-10-2001, http://www.cordis.lu/ist.

UNDL Foundation, 2002a. *The Universal Networking Language Programme. Mission*, http://www.undl.org/missionunlp.html.

UNDL Foundation, 2002b. *UNDL Foundation. Mission*, http://www.undl.org/mis sion.html.

UNDL Foundation, 2002c. *The Universal Networking Language Programme. Statute*, http://www.undl.org/statuteUNLP.html.

Data Set
for Designing and Testing an Arabic Stemmer

Ibrahim A. Al kharashi
Tel: 481-3273, fax: 481-3764
Kharashi@kacst.edu.sa

Imad A. Al sughaiyer
Tel: 481-3217, fax: 481-3764
imad@kacst.edu.sa

Computer and Electronics Research Institute
King Abdulaziz City for Science and Technology
P. O. Box 6086, Riyadh 11442, Saudi Arabia

ABSTRACT

Arabic language has unique characteristics that greatly affect its automation. Arabic language exhibits a very complex but very regular morphological structure. Different proposed morphological analysis techniques for the Arabic language are based on heavy computational processes and/or the existence of large amount of associated information. Researchers in the field of Arabic computational linguistics faced with some basic technical difficulties including lack of propre evaluation and testing frameworks. Because of that, researchers in their works provided general description for approaches with almost no effectiveness or efficiency measures.

This work proposed an initiative for a framework to be used for testing and evaluating Arabic morphological and stemming techniques. A new Arabic stemmer is proposed where the generated data set were used to construct, test and evaluate the stemmer.

INTRODUCTION

Stemming and morphological analysis techniques are computational processes that analyze natural words by considering their internal structures. Stemming techniques usually deal with languages with simple morphological systems while morphological techniques are widely used in languages with complex morphological systems. Stemming and morphological analysis techniques can be viewed as clustering mechanisms and usually help in resolving the lexical ambiguity. The main objective of the stemming algorithms and one objective of morphological analysis techniques is to remove all possible affixes and thus reduce the word to its stem. Both processes are very useful in many natural language applications such as information retrieval, text classification and categorization, text compression, data encryption, vowelization and spelling aids and automatic translation (Lovins, 1968; Dawson, 1974).

Semitic languages require more complicated systems for processing their morphology. Arabic language, for example, consists of a very complex but very rich and regular morphological structure. English has a simple morphology compared to other languages. European languages involve more complex morphology than does English (Savoy, 1999).

In Arabic, a root is a single morpheme that provides the basic meaning of an Arabic word. Arabic root is the word's origin before any transformation process. A stem, on the other hand is a morpheme or a set of concatenated morphemes that refers to some central idea while a word is the single isolated lexeme that represents certain meaning.

An affix is a morpheme that can be added before, after or inserted inside a root or a stem as a prefix, suffix or infix respectively to derive new words or meaning. Arabic prefixes are derived from small set of letters and articles, while suffixes are derived from small set of letters, articles and pronouns. Removal of prefixes in Arabic is not harmful process most of the time because, as oppose to English, the process does not reverse the meaning of the word.

A pattern is a model used to study the internal structure of Arabic words. It consists of the three basic Arabic pattern letters that corresponds to the first, second and third letter of the Arabic trilateral root respectively. For the quadrilateral roots, the third letter is duplicated to represent the fourth root letter. In addition, zero or more augmented letters or one or more short vowels are inserted to expand the pattern.

Computational Arabic morphology drew the attention during the last two decades. This, consequently, has led to the emerging of some morphological analysis techniques. Arabic morphological analysis techniques can be categorized into table lookup, linguistic and combinatorial approaches (Ali, 1988; EL-Affendi, 1991; Al-Fedaghi & Al-Anzi, 1989). Some researchers suggested analyzing Arabic words to reach their roots (Ali, 1988) while others suggested analyzing them to their

stems only (Alsuwaynea, 1995; Al-Atram, 1990). Analyzing words to their roots is preferred in linguistic processing-based applications while analyzing words to their stems is more useful in some other applications such as information retrieval-based systems. A simplified system for generating/analyzing Arabic words is shown in Figure 1.

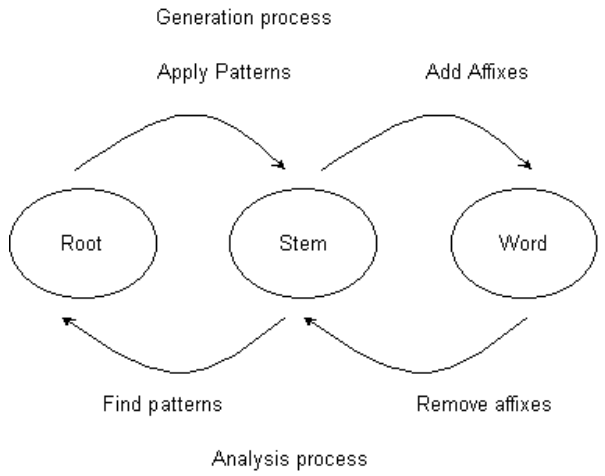


Figure 1. Arabic system for generating/analyzing words

Researchers in the field of Arabic computational linguistics faced with some basic technical difficulties including lack of propre evaluation and testing frameworks. Because of that, researchers in their works provided general description for approaches with almost no effectiveness or efficiency measures.

This work proposed an initiative for a framework to be used for testing and evaluating Arabic morphological and stemming techniques.

ARABIC DATA SET

The framework is based on a data set initially used to design and evaluate a proposed Arabic stemmer. The data set is a collection of about 23,000 Arabic words extracted from 100 short Arabic articles collected randomly from the internet. Extracted words were normalized by removing vowels and then stored in a binary file in the same order as the original natural text. Since word order was preserved, it is very easy to deduce the contextual meaning of any word by listing few words before and after the current word.

Structure of the word record is shown in Figure 2. Each word in the data set was manually investigated to produce morphological components including stem and affixes. In this work, the stem is defined as a singular, masculine and past tense Arabic word without affixes. To guarantee an adequate level of accuracy, an Arabic linguist has been consulted during this stage.

Figure 2. Data structure used in storing words.

If expanded, this data set can be used for other linguistic studies and researches such as morphological analysis techniques, affixation compatibility and different frequency analysis.

Usually, gathered natural text used in modern Arabic is full of spelling errors and spelling variations. Errors corrected partially during the manual processing stage and then completed semi-automatically. Table 1. Lists some examples of errors and spelling variations.

Following is some statistical characteristics of the data set. Figure 3. shows the length distribution of words. Most of the words with length of two letters and some of those with length of three letters are stop words. Furthermore, most of words with the highest lengths are foreign words. Figures 4 and 5 show frequency distribution for word and stem respectively. Figures show normal distribution over the collection.

Spelling mistake	Arabic terminology	Example
Using different spelling variations of foreign words	تهجئة الكلمات الأجنبية	انترنت و انترنيت
Compound nouns. With and without space between parts.	الأسماء المركبة	عبد الرحمن و عبدالرحمن
Confusing between Arabic letters	ه and ة ا and أ ى and ي	هرة – هره استنجار – أنباء علي – على

Table 1. Spelling errors and variations

Table 2. and Table 3. give the frequency of prefixes and suffixes. Such statistics are very useful in different computational and linguistic studies.

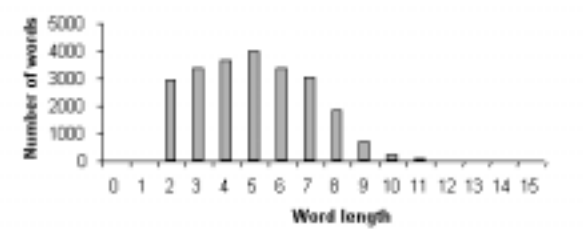


Figure 3. Word length distribution.

Figure 4. Word frequency distribution.

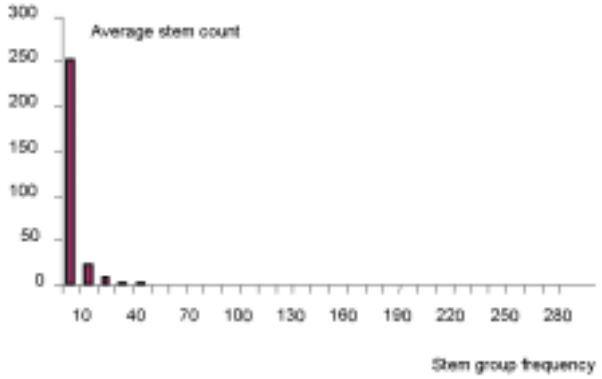


Figure 5. Stem frequency distribution.

Prefix	freq	Prefix	freq	Prefix	freq	Prefix	freq
ال	6890	لل	368	وس	32	ولل	5
و	1484	بال	260	ول	18	كال	5
ب	620	ف	111	وب	16	فال	5
ل	575	س	100	وبال	15	فل	2
وال	476	ك	82	وت	11	فس	2

Table 2. Prefix List Derived from data set

Suffix	freq	Suffix	freq	Suffix	freq	Suffix	freq
ت/ة	1312	وا	44	اتهم	7	وه	2
ات	1136	اتها	34	يون	6	تم	2
ية	1060	ما	32	هما	6	يه	1
ي	698	يات	23	اتية	6	يتها	1
ها	570	تها	21	كم	5	يتنا	1
ه	494	ته	15	ونه	4	وهم	1
ا	445	اته	15	يها	3	وننا	1
ين	256	ان	14	وها	3	تين	1
ون	141	يين	11	هن	3	تموها	1
هم	134	و	9	تهم	3	تان	1
يا	68	ك	9	تنا	3	اها	1
نا	61	ني	8	اتنا	3	اتكم	1

Table 3. Suffix List Derived from data set

PATTERN-BASED ARABIC STEMMER

In this section, a new approach that utilizes the apparent symmetry of generated natural Arabic words is introduced. In this approach, a unique regular expression-based rule is generated for group of similar Arabic words. Rules are used to describe the internal morphological structure of Arabic words and guide the decomposition process of a given word to its basic units i.e. stem, prefix and suffix. A very simple rule parser was developed to perform the analysis to process and extract word morphological components.

Created rules are written from right to left to match script writing direction of Arabic language. Rule pattern may contain up to three distinct parts. The first and last parts describe affixation properties of the word while the middle part controls the stem extraction process. Pairs of angle brackets surround affixation parts. Absence of prefix or suffix in the rule patterns is sometimes denoted by empty angle brackets. This is necessary in order to distinguish them from an angle-bracketed part of the stem.

The complexity of rules varies from very simple passive ones to very complicated rules that deal with complex morphological behaviors. Set of passive rules is created to handle words already in stem forms, isolated articles, proper names and foreign words.

A rule will be fired if it has the same length as the length of the inspected word. A match is achieved if and only if a fired rule produces the correct prefix, stem and suffix. A given word should fire at least one rule and match only one rule.

EXPERIMENTATION

Created data set has been used in the design and implementation stages of the stemmer. The first part of the experiment was designed to study rule growth in a natural text. In this part each word passed to the parser for analysis. The parser has access to list of accumulated rules. The parser tries to fire rules in sequence. On match, the word structure will be updated with number of fired rules, the id of matched rule and its sequence. On mismatch, a new rule should be created and appended to the rule list.

Figure 6 shows the growth of rules. It shows very rapid growth at lower number of words and a tendency to

be stabilized as more words introduced. Figure 7 depicts number of generated rules for every thousand words. It clearly shows that number of generated rules decreases as number of words increases.

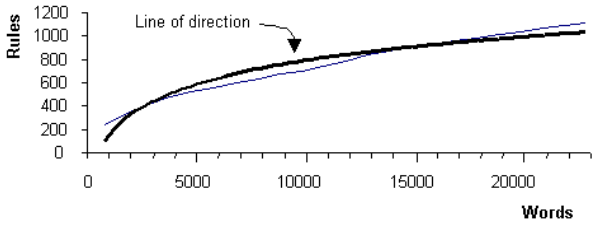


Figure 6. Rules growth per 1000 words

Figure 8 shows the length distribution of words and created rules for the test collection. It can be deduced that majority of rules were generated by words of length 5, 6, and 7 letters. This is a normal phenomenon because words of such lengths are more likely to have diverse kind of affixes. Existence of affixes, consequently, produces more rules. For words with shorter lengths, number of introduced rules were low due to the fact that shorter words are most likely to be particles or words already in stem forms. Fewer rules were introduced for words with longer lengths because most words are either proper names or foreign words. Such type of words is less likely to have affixes.

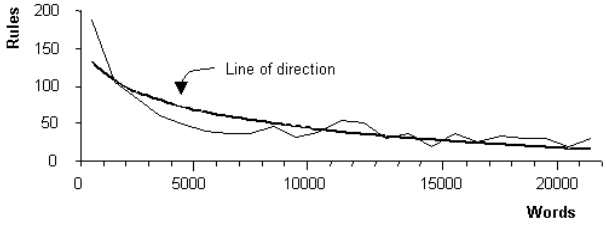


Figure 7. Number of rules generated per 1000 words.

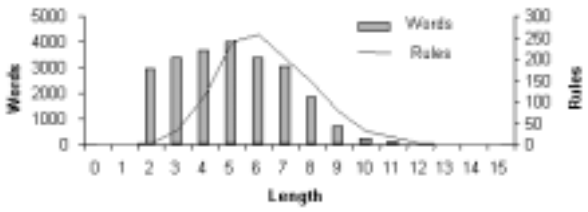


Figure 8. Distribution of rule and word lengths

The order of rule firing plays an important role in the efficiency of the analyzer. For a given word, it is desirable to fire less number of rules and to maintain firing order in such a way that first fired rule is the matched one. Figures 9 and 10 show the relationship between matched and total firings per rule. Having different rule orders will produce different plots. In order to achieve optimized performance the curve of Figure 9. should follow the horizontal line or the scattered points in Figure 10. aligned with the diagonal line. Although it is impractical to achieve such optimum state, it is possible to have certain rule ordering that produces the best performance for such rule set.

Figure 11. indicates that average fired rules is in-line with the conclusion derived from Figure 8. For optimized analyzer, it is desirable to keep average fired rule for each word length class as low as possible. Also, for optimization, it is needed to keep the sequence of the matched rule at the top of firing sequence.

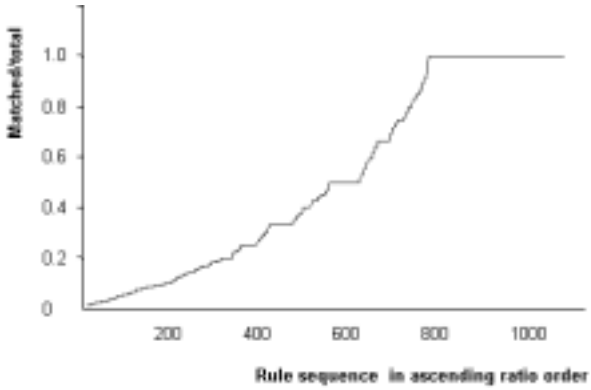


Figure 9. Relation between matched and total firing.

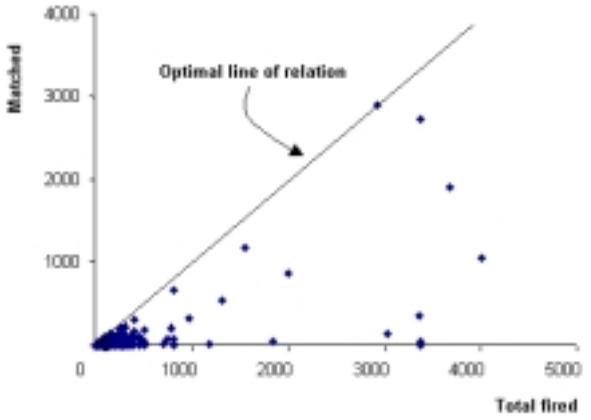


Figure 10. Match vs. total per rule matched and total firing.

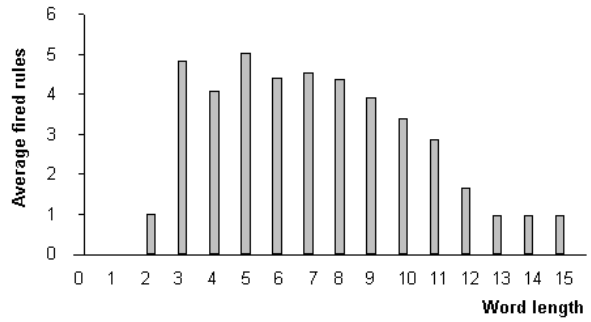


Figure 11. Average fired rules vs. word length.

CONCLUSION

Known Arabic morphological analysis techniques suffer from few problems including the need for testing and evaluating frameworks. This paper proposed an initiative for a framework to be used for testing and evaluating Arabic morphological and stemming techniques. The framework is based on a data set initially used to design and evaluate a proposed Arabic stemmer. The data set is used to construct, test and evaluate the stemmer.

This data set can be used for other linguistic studies and researches such as morphological analysis techniques, affixation compatibility and different frequency analysis.

REFERENCES

Lovins, J. (1968). Development of a stemming algorithm. Mechanical Translation and Computational Linguistics, No. 11, (pp 22--31).

Dawson, J. (1974). Suffix removal and word conflation. *ALLC Bulletin*, 2(3), 33--46.

Savoy, J. (1999). A stemming procedure and stopword list for general French corpora. *Journal of the American Society for Information Sciences*. 50(10), 944--952.

Ali, N. (1988). Arabic Language and Computer. Ta'reeb. (in Arabic)

El-Affendi, M. (1991). An algebraic algorithm for Arabic morphological analysis. The Arabian Journal for Science and Engineering. 16(4B), 605--611.

Al-Fadaghi, S. and Al-Anzi, F. (1989). A new algorithm to generate root-pattern forms. Proceedings of the 11th National Computer Conference, KFUPM. (pp 391--400).

Alsuwaynea, A. (1995). Information Retrieval in Arabic language. King Fahad National Library, (in Arabic).

Al-Atram, M. (1990). Effectiveness of Natural Language in Indexing and Retrieving Arabic Documents. KACST, AR-8-47. (in Arabic).

A Prototype English-to-Arabic Interlingua-based MT system

Abdelhadi Soudi*, Violetta Cavalli-Sforza†, Abderrahim Jamari#

* CLC, Ecole Nationale de L'Industrie Minérale,
Av. Hadj Ahmed Cherkaoui, B-P: 753 Agdal, Rabat, Morocco
asoudi@enim.ac.ma

† Department of Computer Science
San Francisco State Univ., 1600 Holloway Avenue, San Francisco, California, U.S.A.
vcs@sfsu.edu

Institut Universitaire de la Recherche Scientifique
Rabat, Morocco
iramaj5@hotmail.com

Abstract

This paper describes an ongoing research project on English-to-Arabic Interlingua-based machine translation. Section 1 gives a description of the system that generates Arabic sentences from Interlingua representations (IRs). In section 2, we show how basic sentential components are mapped. In this context, we address some of the differences between English and Arabic such as agreement in number which cannot be transferred exactly from the IR of an English sentence. Results and an example translation are provided in section 3. In this context, we address the issue of word order variation in Arabic.

1. The Architecture of the Arabic Generation System

An Interlingual approach to machine translation (MT) has a number of advantages over other approaches, such as the 'transfer' model. In an Interlingua-based architecture, source text analysis and target text generation are divided into separate components. A language-independent intermediate representation (or Interlingua) mediates between these two components. The decoupling of the analysis and generation phases allows the system to handle multiple-language output and avoids the reconfiguration of the system for each new language.

In the KANT Interlingua-based MT system (Nyberg, and Mitamura, 1992), each sentence is first conveyed into tokens. The KANT analyzer uses a lexicon, a morphological analyzer, source language grammar and semantic information in order to parse the tokenized sentence into a feature structure (FS), a list of feature-value pairs that reflects the syntactic structure of the source language (i.e., English). The interpreter then uses mapping rules to convert the FS into an IR. An IR is a tree-structured representation that abstracts away many of the syntactic details of both source and target language, while conveying the meaning of the source language. In section 3 below, we provide an example of a source language FS, the IR produced from this FS and the target language FS produced from the IR.

Generation of the target language sentence begins with the IR. The system which generates Arabic sentences from IRs consists of 4 subsystems: the mapping system, the sentence generation system, the sentence/morphology generation interface and the morphological generation system, as shown in Figure 1 below.

First, the generation mapping rules convert the IR into an FS that reflects the syntactic structure of the target language. The FS is a list of feature-value pairs that reflects the syntactic structure of the target language. Target language lexicon entries are FSs. They are retrieved during mapping and added to the sentence FS under construction. The Genkit grammar analyzer and generator (Tomita and Nyberg, 1988) processes the input FS and generates a preliminary target sentence string, calling MORPHE when it encounters lexical symbols in the generation grammar.[1] This string is optionally run through the CODA post-processing system to produce the final target sentence.

1.1. The Mapping System

The mapping system produces FSs for Arabic from IRs, using a set of mapping rules and a mapping lexicon. The mapper recursively traverses the Interlingua, stopping at each level to examine slots and their fillers (features, concepts and nested Interlinguas). Testing a hierarchy of rule declarations, the mapper performs a structure-building operation called mapping. The goal and result of mapping is a target-language FS whose contents reflect the contents of the Interlingua, expressed in terms of the syntactic and lexical properties of the target language. The mapping process involves three main stages:

- Selecting lexical items for each Interlingua concept;
- Mapping the semantic roles for each Interlingua concept (slots in the Interlingua frame) to grammatical functions (slots in the FS);
- Mapping semantic features for each Interlingua concept to the appropriate syntactic features in the FS.

The mapper's knowledge is represented as mapping rules that are stored in a mapping hierarchy. The use of a hierarchy allows one to write specific rules for specific concept/lexeme pairs and general ruleswhich are inherited.

[1] The morphology/generation interface consists of a lisp program that defines some functions that are used to call the morphological generator from the sentence generator.

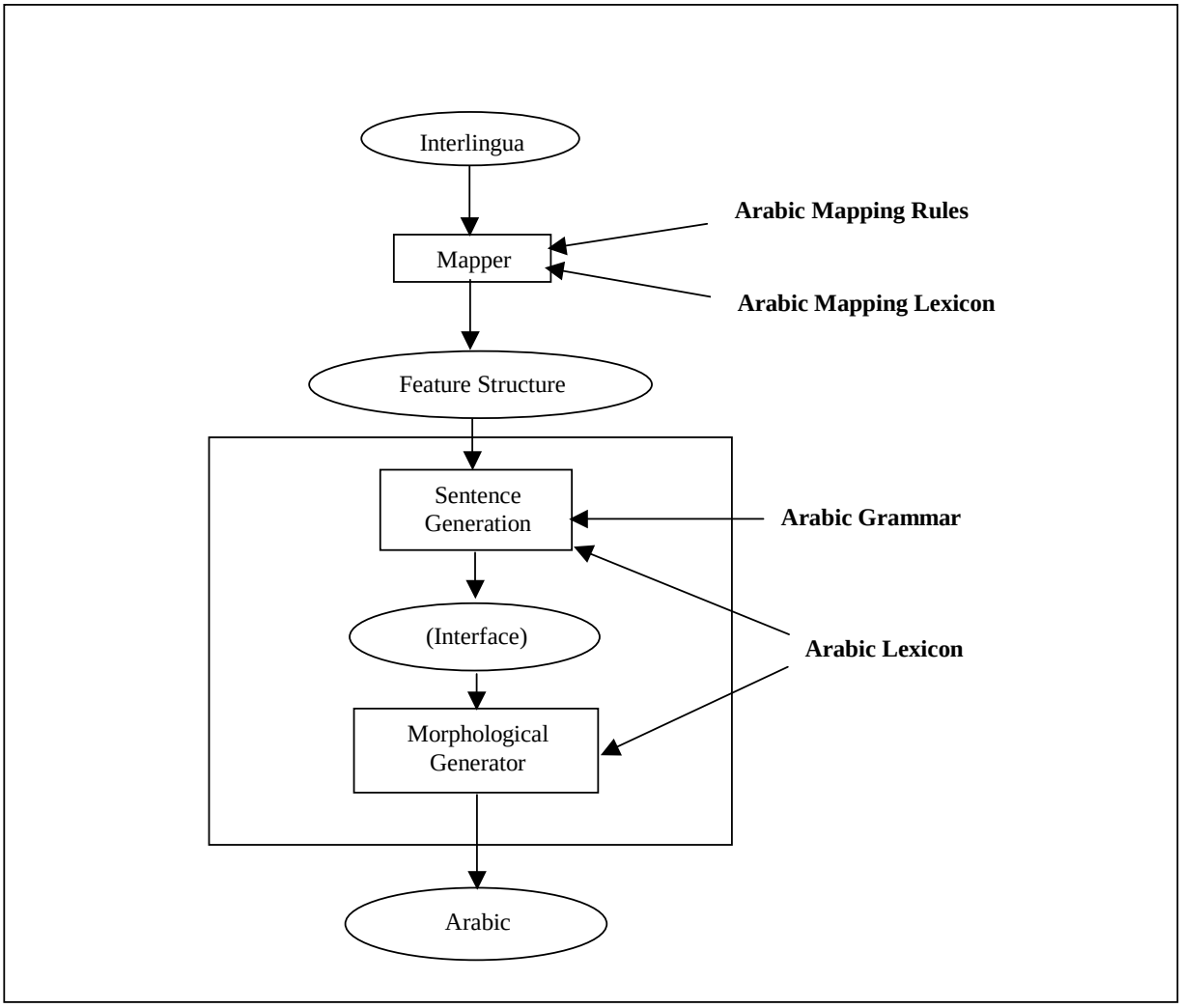


Figure 1. The Architecture of the Arabic Generation System

1.1.1. Concept Encoding Information

Each node in the mapping hierarchy has a name, a list of concepts, and a list of mapping rules to be executed. In addition, it has links connecting to one or more parent nodes. The examples in (1) below show how the concepts *shine* and *house* are encoded:

```
(1)
a. (node ?A-shine
        :parents (VERB)
        :encodes (*A-shine)
        :rules ((:lex "ta?allaq")))

b. (node ?O-house
        :parents (NOUN)
        :encodes (*O-house)
        :rules ((:lex "manzil")))
```

The node names *A-shine and *O-house are arbitrary symbols used to distinguish the nodes. They denote lexical interlingua concepts that would be associated with the lexical entries for the verb 'to shine' and the noun 'house' in the English lexicon. The :parents field specifies the part of speech that these nodes inherit from in the mapping hierarchy. The :encodes field and the :rules field specify which Interlingua concept this node will realize and the mapping rules associated with this node, respectively. ?A-shine and ?O-house denote the names of the lexical nodes used to determine the corresponding Arabic translation.

1.1.2. The Syntactic Lexicon

The syntactic lexicon consists of two parts: templates and entries. The templates specify the default contents of various types of lexical FSs. (2) below illustrates an Arabic syntactic template:

```
(2)  (soft-template conj ((cat conj)))
```

The entries associate each lexeme with a template class and specify the unique features for that particular lexeme, as is illustrated by the following example:

```
(3)  (conj "wa" ((ROOT "wa")))
```

1.1.3. The Mapping Rules

A mapping rule is a set of slots and values that specify operations involved in building an FS from an Interlingua. The lexical nodes in (1a-b) above illustrate a :lex mapping rule, which retrieves a translation from the target language lexicon. Mapping rules may also contain other directives

(e.g. such :map, :test, :add, :force-add, :consume, etc.) for performing other operations on the IR and FS.

For the sake of concreteness, consider the following mapping rule from (Soudi, 1999, pg. 13):

```
(4)  (:test (:sem (number plural)
            :syn (:not (human +)))
     :force-add ((agr ((gender f) (number sg)))))
```

The mapping rule above consists of a set of slots and values associated with the noun mapping hierarchy node. The :test slot specifies a set of conditions that must be passed for the rule to be applied. The :syn subslot specifies a negated condition on the FS, namely the feature (:not (human +), that must be met. The :sem subslot specifies a condition on the IR, namely the FVP (number plural). The slot :force-add indicates that the FS under construction should have feminine as its gender value and singular as its number value. This slot actually overrides information in the IR: the value of the number feature in the IR, namely plural, is overridden here by the singular. The mapping rule above applies to the sound plural feminine in Arabic (i.e., the *-At* class). By way of example, in the IR for the French noun *les animaux* 'animals', we would have, inter alia, the feature-value pairs (number plural) and (gender masculine). This information should be overridden for the corresponding Arabic noun 'Hayawanaat' – which is (human -) – by the feature-value pairs (number singular) and (gender feminine). Note that the information specified by the :force-add slot in the example above relates to subject-verb agreement. Thus, the sound plural noun *Hayawanaat* is plural but has 'singulative' agreement with verbs.

1.2. The Generation Grammar

To generate Arabic sentences, we have used Genkit (Generation Kit) (Tomita and Nyberg, 1988), a system that compiles a grammar written in a formalism called Pseudo-Unification Grammar into a sentence generation program. The generator follows a top-down, depth-first strategy for applying rules during generation.

The following example shows a unification-based grammar rule for generating sentences. The rule consists of a context-free phrase structure rule and a list of pseudo equations.

```
(5)  (<S> ==> (<NP> <VP>)
          (((x1 agr) = (x2 agr))
           (x1 == (x0 subj))
           (x1 case) = nom)
           (x2 = x0)))
```

The non-terminals in the phrase structure part of the rule are referenced in the constraint equations as $x_0 \ldots x_n$, where x_0 is the non-terminal in the left-hand side (here, <S>) and x_n is the *n-th* non-terminal in the right hand side. In these equations, x_1 represents <NP> and x_2 represents *<VP>*. The rule in (5) is for sentences with an *<NP>* and a *<VP>* that agree in number, person and gender. The equation ((x1 agr) = (xs agr)) indicates that the *<NP>*'s agr feature has a value that unifies with the value of the *<VP>*'s agr feature.

2. Arabic Noun and Verb Mappings

The generation of properly inflected Arabic verbs and nouns is a concern of both the mapper and the generator for a partial integration of the Arabic Morphology system into the KANT system). For example, the generation of correct agreement between nouns and their modifiers or other parts of the sentence may be performed either during mapping or during generation. Different cases must be considered:

(a) **Subject-Verb/Verb-Subject Agreement:** In Arabic, agreement in number between subject and verb depends on the nature of the subject of the sentence and word order. On a VS order, verbs do not agree in number with a plural subject. Agreement is always singular. Verbs, however, agree with their subjects in person and gender, as is illustrated by the following rule for generating a VS order sentence (from Soudi, 1999, pg. 16)):

```
(6)  (<s> ==> (<vp> <np>)
          (((x1 agr) = (x0 subj agr))
           ((x1 agr number) <= 'sg)
           (x2 == (x0 subj))
           ((x2 case) = nom)
           (x1 = x0)))
```

(b) **Intrinsic Number:** In most cases, the number feature for a noun is determined by the input sentence, reflected in the IR, and mapped directly from the IR into the FS by the mapper. Some nouns, however, may have agreement constraints already present in the lexicon. While lexical entries for nouns are usually assumed to be singular, certain nouns may be intrinsically plural in terms of agreement. For example, the noun *naAs* 'people', would contain the agreement information (number pl) in the lexicon, and the mapper should not override it with information that may be present in the Interlingua (for example, if the source language were Italian or Spanish, in which the word is a singular collective noun).

(c) **Number-Noun Agreement:** Number-noun agreement is governed by a set of complex rules. With the number 'one', agreement is as expected, but there may be a reversal of word order (e.g. *kitaabun waaHidun* 'one book' (nominative)). The number 'two' is expressed by the dual of the noun. Numbers 'three' through 'ten' require the noun to be plural and the gender of the number to be the opposite of the gender of the singular noun. For example: *xams* 'five' (masculine) *sanawaat* (plural of *sanat* 'year', feminine) but *xamsatu* 'five' (feminine) *kutub* (plural of *kitaab* 'book', masculine). Up to ten (plural of paucity), numbers and nouns agree in case, which is determined by the syntactic construction they appear in. Numbers above ten (plural of multiplicity) require a singular noun in the indefinite accusative. Agreement decisions can be made in the generator with the help of a callout function, but are most easily handled using the mapper.

3. An Example Translation and Results

To demonstrate the function of the components described in section 1, we will use the example sentence below:

(7) Jakarta and Bangkok are shining the most.

In the current system, the mapper takes as input the IR in (8), which is generated by the KANT analyzer and interpreter, and produces the FS for Arabic (9), using a set of mapping rules and a mapping lexicon (Soudi, 1999, pg. 20):

(8) **The Interlingua**

```
(*A-SHINE
  (FORM FINITE)
  (TENSE PRESENT)
  (MOOD DECLARATIVE)
  (PUNCTUATION PERIOD)
  (PROGRESSIVE +)
  (IMPERSONAL -)
  (ARGUMENT-CLASS AGENT)
  (MANNER
  (*M-THE-MOST
    (POSITION POSTVERBAL)
    (UNIT -)
    (DEGREE POSITIVE)))
    (AGENT
      (*G-COORDINATION
        (PERSON THIRD)
        (IMPLIED-REFERENCE +)
        (CONJUNCTION  (*CONJ-AND))
        (CONJUNCTS
          (:MULTIPLE
            (*PN-JAKARTA
              (UNIT -)
              (PERSON THIRD)
              (NUMBER SINGULAR)
              (REFERENCE NO-REFERENCE))
            (*PROP-BANGKOK
              (UNIT -)
              (PERSON THIRD)
              (NUMBER SINGULAR)
              (REFERENCE NO-REFERENCE)))))))
```

(9) **The FS**

```
((ADV ((CAT ADV) (ROOT "?ak#ar")))
 (form 4)
 (CAT V)
 (ROOT "ta?allaq")
 (VOICE ACT)
 (TENSE IMPERF)
 (MOOD INDIC)
 (SUBJ
   ((ELEMENT
      (*MULTIPLE*
          ((AGR ((GENDER F) (PERSON 3) (NUMBER SG)))
           (CAT N) (ROOT "jakarTaa"))
          ((AGR ((GENDER F) (PERSON 3) (NUMBER SG)))
           (CAT N) (ROOT "baankuuk"))))
     (CONJ ((CAT CONJ) (ROOT "wa")))))
 (PUNCTUATION ((ROOT PERIOD))))
```

Most of the linguistic features used in the KANT Interlingua and FS (e.g., punctuation, form, tense, argument class, number, person) should be self-evident. Some other features are artifacts of KANT's evolution as a technical text system. The IMPLIED-REFERENCE feature is used for nouns, such as the proper noun in the example above. G-COORDINATION contains all conjuncts that are coordinated and the conjunction that is used.[2]

The resulting FS serves as input to the Arabic morphological and sentence generator, producing Arabic surface forms:

(10) baAnkuwk wa jakaroTaA tata^alGaqaAni ^ako#ar

A major problem with the current implementation of the system relates to the word order variation in Arabic. Arabic is basically a VSO language, in which constituents can change order according to the constraints of text flow or discourse. The grammatical roles of constituents are identified by explicit morphological case markings. However, the KANT analyzer does not mark constituents as topic or focus. That is, this information is not provided in the IR. For example, there is no information structure for the system to decide whether to generate a VS order (12a) or an SV order (12b) from an IR for the English sentence in (11):

(11) Zayd ate the apple.

(12)

a. ?akala zayd-un t-tuffaaHat-a.
   ate Zayd-nom the-apple-acc

b. zayd-un ?akala t-tuffaaHata
   Zayd-nom ate the-apple.

Currently, the system produces all sentences in the S(=topic)V order.

While there are challenges to be worked out where the source language and target language differ greatly in their morphology and syntax, an Interlingua approach allows for a flexible integration of software modules for languages that differ in their realization of the same unit of meaning. Indeed, most of the morphological and syntactic differences between the source language and the target language can be handled by either the mapper or the generation grammar.

The system is still under construction. It has been tested on 29 different structures and has produced good results.

4. Conclusion

In this paper, we have described an ongoing research project on English-to-Arabic Interlingua-based machine translation. After giving a description of the system that generates Arabic sentences from IRs, we have shown how basic sentential components are mapped. In this context, we have addressed some of the differences between English and Arabic, such as agreement in number which cannot be transferred exactly from the IR of an English sentence. We have also provided an example translation and results.

[2] To promote representational consistency, the same structure is (*G-COORDINATION) is used if there is no explicit conjunction. In this case, the feature CONJUNCTION will have the value NULL.

5. References

Aronoff, M., 1994. *Morphology by Itself: Stems and Inflectional Classes*. Cambridge, Mass: MIT Press.

Beard, R., 1995. *Lexeme-Morpheme Base Morphology: A General Theory of Inflection and Word Formation.* State University of New York Press.

Cavalli-Sforza, V., A. Soudi, A., and T. Mitamura, 2000. Arabic Morphology Generation Using a Concatenative Strategy". *Proceedings of the North American Association For Computational Linguistics (NAACL)*, 2000, Seattle, United States.

Fassi Fehri, A., 1993. *Issues in the Structure of Arabic Clauses and Words*. Dordrecht, Holland: Kluwer Academic Publishers.

Mitamura, T., E.H. Nyberg, and J. Carbonell, 1991. An Efficient Interlingua Translation System For Multilingual Document Production. *Proceedings of the 3rd Machine Translation Summit.*

Nyberg, E.H. and T. Mitamura, 1992. The KANT System: Fast, Accurate, High Quality Translation in Practical Domains. *Proceedings of COLING'92.*

Schramm, G., 1962. An Outline of Classical Arabic Verb Structure. *Language*, 38:360-75.

Soudi, A., 1999. Interfacing an Arabic Morphological Generator with an Interlingua-based Machine Translation System. ms. Carnegie Mellon University, USA.

Soudi, A., V. Cavalli-Sforza, and A. Jamari, 2001. A Computational Lexeme-based Treatment of Arabic Morphology. *Proceedings of The Arabic Processing Workshop, Association For Computational Linguistics*, Toulouse, France.

Soudi, A., V. Cavalli-Sforza, and A. Jamari, 2002. The Arabic Noun System Generation. *Proceedings of the International Conference on Arabic Processing*, University of Manouba, Tunisia.

Timothy, A.B., 1990. Lexicographic Notation of Arabic Noun Pattern Morphemes and their Inflectional Features. *Proceedings of the Second Cambridge Conference on Bilingual Computing in Arabic and English*. No pagination.

Tomita, M., and E.H.Nyberg, (1988). *Generation Kit and Transformation Kit, Version 3.2, User's Manual*. Technical Report, Carnegie Mellon University, Center for Machine Translation.

Wright, W, 1966. *Lectures on The Comparative Grammars of Semitic Languages.* Amsterdam:Philo Press.

Wright, W., 1988. *A Grammar of the Arabic Language.* Cambridge:Cambridge University Press, 3rd edition.

Arabic Document Topic Analysis

Thorsten Brants, Francine Chen, Ayman Farahat

Palo Alto Research Center (PARC)
3333 Coyote Hill Rd, Palo Alto, CA 94304, USA
{brants,fchen,farahat}@parc.com

Abstract

We adopt algorithms for document topic analysis, consisting of segmentation and topic identification, to Arabic. By doing so, we outline the requirements for Arabic language resources that facilitate building, training, and fine-tuning systems that perform these tasks. Our segmentation and topic identification algorithm is based on Probabilistic Latent Semantic Analysis. First results for segmenting Arabic texts are reported.

1. Introduction

Document topic analysis is the task of assigning one or more topics to a document, characterizing the sub-topics discussed in the document, and identifying boundaries between segments discussing the different sub-topics. Most of the work in text retrieval has been on identifying and ranking the most relevant documents, although there is also work on passage retrieval. Topic analysis has applications in enabling retrieval at a finer grain than at the document level, but at a broader level than a passage.

One step in document topic analysis is topic-based segmentation. This task has been addressed by several authors. All methods calculate the similarity between the text before and after a hypothetical segment boundary and assume a segment boundary if the similarity value is small. Hearst (Hearst, 1997) describes TextTiling. She uses a sliding window and computes similarities between adjacent blocks based on their term frequency vectors. Li and Yamanishi (Li and Yamanishi, 2000a; Li and Yamanishi, 2000b) present a structured Finite Mixture Model, which they refer to as a stochastic topic model (STM). Choi et al. (Choi, 2000; Choi et al., 2001) present a model based on Latent Semantic Indexing (LSI) and divisive clustering. We have developed a segmentation method that uses the Probabilistic Latent Semantic Analysis (PLSA) model (Hofmann, 1999) for smoothing the term frequency vectors in a way that better models synonomous terms.

Topic-based segmentation is different from finding story boundaries in the TDT program. There, segmentation is not necessarily topic based, but also can (and commonly does) utilize a large variety of cue phrases which are usually absent in topic-based segmentation.

Figure 1 shows an example topic analysis for a part of an article that appeared in the El Hayat newspaper (the complete article is too long to be printed as an example). Our segmentation algorithm identified two segments, which are represented as non-underlined as underlined text. The first segment is about Israeli military operations in the West Bank, the second segment is about international efforts to defuse the tension. The next steps in topic analysis are topic identification and keyword or key phrase generation. Possible keywords are given to the right of the text (first segment: Israel, occpupy, Palestine; second segment: withdraw, stop, international).

2. Training for Arabic Document Topic Analysis

In this section, we outline the resources that are currently available for performing Arabic document topic analysis, and the resources we ideally would like to have.

2.1. Morphology

Algorithms for English document topic analysis usually depend on a morphological analyzer that associates each full-form of a word with its base form or stem. This significantly decreases the number of distinct word forms in a text by uniquely mapping a full form to some base form.

Stemming Arabic is much more difficult than stemming English. Reduction to roots can be done uniquely in the majority of cases but this would yield a very coarse grained model because words with only remotely connected meanings often share the same root. Reduction to stems (i.e., a root and a pattern) is done by the analyzer presented in (Beesley, 1996), but the output at this level is very ambiguous because of the omission of vowels in writing and the existence of diacritics and clitics. Some researchers resorted to the use of character n-grams instead of words for statistical Arabic models (Sawaf et al., 2001). Systems for uniquely identifying clitics and stems for Arabic are highly desirable as a preprocessing step for document topic analysis.

Preferable resource: Corpus of modern standard Arabic, labeld with uniquely identified stems (root and pattern) and clitics.

2.2. Segmentation

Current segmentation algorithms are trained unsupervised, i.e., no training data with explicitly labelled segment boundaries is provided. But evaluation requires such data. In the absence of documents labelled with segment boundaries, developers of segmentation algorithms use concatenated documents and try to identify the document boundaries (Choi, 2000; Hearst, 1997; Li and Yamanishi, 2000a). However, this is sub-optimal since segment boundaries *within* a document are expected to represent smaller topic shifts than boundaries *between* documents. We expect that the accuracy of a system evaluated on real document segments is lower than on artificially concatenated documents.

Preferable resource: Corpus of modern standard Arabic, labeled with segment boundaries within documents.

Figure 1: Example topic analysis for a document from the El Hayat newpaper, Apr. 11, 2002. Our segmentation algorithm TopSeg-C identified two parts. The first segment (non-underlined) is about Israeli military operations in the West bank, the second segment is about international efforts to defuse the tension. Keywords to identify the topic of the different segments are given to the right of the text.

2.3. Topic Identification

For English, collections labelled with large numbers of topics are available, e.g., in the Reuters-21578 corpus, each document is labelled with one or more of 90 different topics. Such a corpus is currently unavailable for Arabic. ELRA recently made available a collection of documents that are organized in seven domains. TREC-2001 made a step towards more detailed topics giving 25 topic descriptions but only a small number of documents (those necessary for TREC-2001) were manually labelled. Arabic topic analysis systems would benefit from large collections annotated with more fine-grained topics. This would allow topic identification and keyword evaluation as presented in (Li and Yamanishi, 2000a).

Preferable resource: Corpus of modern standard Arabic manually labeled with a fine-grained set of topic labels and keywords for each document as a whole, and for each segment in each document.

3. Topic Based Segmentation

3.1. TopSeg

TopSeg, our text segmentation system, combines the use of the Probabilistic Latent Semantic Analysis (PLSA) model (Hofmann, 1999) with the method of selecting segmentation points based on the similarity values between pairs of adjacent blocks. PLSA represents the joint probability of a document d and a word w based on a latent class variable z:

$$P(d,w) = P(d)\sum_{z} P(w|z)P(z|d) \qquad (1)$$

A model is fitted to a training corpus $\mathcal{D}$ using the Expectation-Maximization algorithm (EM) to maximize the log-likelihood function $\mathcal{L}$:

$$\mathcal{L} = \sum_{d\in\mathcal{D}}\sum_{w\in d} f(d,w)\,log P(d,w). \qquad (2)$$

After a model is trained, the model parameters $P(w|z)$ obtained in the training process are used in the process of folding-in the new (test) documents into the PLSA model to calculate $P(z|q)$ for new documents q. In the folding-in process, the Expectation-Maximization algorithm is used in a similar manner to the training process: the E-step is identical, in the M-step $P(w|z)$ is constant for all w and $P(z|q)$ is recalculated at each iteration. Usually, a very small number of iterations is sufficient for folding-in.

To segment a document, the document is first preprocessed by tokenizing the document and identifying sen-

tence boundaries. For English, two additional steps, downcasing and stemming, are performed. Next the text is broken in *blocks* of sentences. Candidate points of segmentation are identified and correspond to the locations between the text blocks. In our case, blocks are overlapping (as in a sliding window) and consist of h (e.g., $h = 5$) consecutive sentences.

Folding-in is then performed on each block b to compute the distribution among the set of latent classes, $P(z|b)$, where z is a latent variable, and b is a block. The estimated distribution of words for each block b, $P(w|b)$, is then computed as

$$P(w|b) = \sum_z P(w|z)P(z|b) \quad (3)$$

for all words w, where $P(w|z)$ is taken from the PLSA clustering of the training documents. The distribution of words in adjacent blocks b_l and b_r is compared using a similarity metric based on the Hellinger distance (Basu et al., 1997), also known as the Bhattacharyya distance (Kailath, 1967):

$$\mathrm{sim}_{\mathrm{Hel}}(b_l, b_r) = \sum_w \sqrt{P(w|b_l)P(w|b_r)}. \quad (4)$$

Dips are local minima in the similarity of adjacent blocks. We expect larger dips to correspond to stronger changes in topic. In our evaluation task, the number of segments is known is in advance, and we select the locations of the largest dips as segementation points.

3.2. TopSeg Using Combined Models

Training a PLSA model using EM starting with a random initialization yields a locally optimal model that is reached from the given start position. In general, different initializations yield different locally optimal models, which in turn might yield significantly different segmentation error rates.

One possibility to reduce the effect of different initializations is to generate several PLSA models, each with a different initializations. Then similarity values between adjacent blocks are computed according to the different models, and the resulting similarity values are averaged, yielding an *averaged similarity* curve.

The algorithm for combined models, TopSeg-C, generates k different PLSA models from the same training set, starting with different initializations. Each of the k models is used for folding-in the blocks of the test documents, and similarities between the blocks are calculated according to the k models. Now, the k similarity values for each pair of adjacent blocks are combined by calculating the average similarity value, yielding the average similarity curve. Local minima (dips) are determined in this resulting curve, and the largest dips are identified as the segment boundaries.

Similarly, we can use PLSA models with different numbers of latent classes to generate an averaged similarity curve. This was suggested by (Hofmann, 1999). However, we found that averaging over different initializations (with the same number of latent classes) yields slightly better results.

Table 1: The two corpora used in the experiments.

	Reuters-21578 (training set)	Arabic (training set)
Corpus		
# documents	7,769	6,482
# tokens	1,156,828	1,156,156
# types	41,343	70,148
# topics	90	–
Vector Space Model		
# terms	22,142	67,270
# terms $f > 1$	11,042	38,358
# n-grams	–	222,986
# n-grams $f > 1$	–	133,362

4. Topic-Based Segmentation Experiments

We performed first segmentation tests for Arabic using TextTiling and our PLSA-based model, TopSeg. Experiments and results are reported in this section. The TextTiling experiments serve as a baseline for our new segmentation model that we are currently developing for Arabic.

4.1. Data

Most of the resources outlined in section 2. are not available yet. We therefore resort to basic preprocessing and evaluation methods for performing the task of Arabic topic-based segmentation.

We prepare Arabic documents in a similar manner as Li & Yamanishi (Li and Yamanishi, 2000a) prepared documents from the Reuters-21578 corpus. 500 test documents are generated by randomly choosing two documents from the AFP Arabic Newswire Corpus (year 1994) and concatenating them into one. The task is to detect the document boundary. The system uses 6,482 documents for training[1] (training and test set are disjoint). Information about the Reuters-21578 set and the Arabic set that we prepared are provided in table 1. The sizes of the data sets are roughly comparable, but with the Arabic documents longer on average. The difference in the number of terms is even larger since we applied stemming to the English data.

Optimal values for the block size h for each model and the number of clusters Z for TopSeg and TopSeg-C were determined in preliminary experiments. For the following experiments, we set $h = 6$ for TextTiling, and $h = 5$, $Z = 256$ for TopSeg and TopSeg-C.

4.2. Results

We use the probabilistic error measure suggested by Beeferman et al. (1999) to report the results of our experiments. It is the probability p_{err}^{kw} that two *words* at distance k_w words are incorrectly identified to belong to the same/to different segments. For comparison, we present segmentation results on English data using TextTiling and STMs,

[1]The AFP Arabic Newswire Corpus is available from the Linguistic Data Consortium. The document identifiers of our concatenation of the training and test documents are available at `http://www.parc.com/istl/groups/qca/arabic-data/`

Table 2: Segmentation sentence error rate. Results marked with * are taken from (Li and Yamanishi, 2000a; Li and Yamanishi, 2000b).

Corpus	Algorithm	Terms	p_{err}^{kw}	p_{err}^{ks}
* Reuters-21578	TextTiling	stems	-	8.5 %
* Reuters-21578	STM	stems	-	9.2 %
AFP Arabic	TextTiling	fullform	8.09%	9.40%
AFP Arabic	TextTiling	n-grams	5.83%	7.49%
AFP Arabic	TopSeg	fullform	3.10%	3.88%
AFP Arabic	TopSeg	n-grams	3.05%	3.94%
AFP Arabic	TopSeg-C	fullform	**2.26%**	**2.91%**
AFP Arabic	TopSeg-C	n-grams	**2.30%**	**2.94%**

which were given in (Li and Yamanishi, 2000a; Li and Yamanishi, 2000b). They used a slightly different measure, i.e., the probability p_{err}^{ks} that two *sentences* at distance k_s sentences are incorrectly identified to belong to the same/to different segments. We therefore give both measures, p_{err}^{kw} and p_{err}^{ks}, for our results. k_w and k_s are set to be half the average length (in words and sentences, respectively) of a segment.

Table 2 presents the results on English and Arabic documents. We compare three different algorithms: TextTiling, our new algorithm using PLSA (TopSeg), and its variant using several PLSA models that are combined (TopSeg-C). Each of the three algorithms is run using full forms and using n-grams. For TextTiling, n-grams yield significantly better results than full forms (5.83% vs. 8.09% word based segmentation error rate). TopSeg yields almost identical results for n-grams and full forms. This may be explained by the property of PLSA to cluster semantically similar words, which is absent in the TextTiling algorithm. Results for using stems in Arabic are unknown yet, since no Arabic stemmer producing unique stems was available to us. In order to avoid variation that is due to different initializations of the PLSA models, we repeated each experiment using single models four times and report averaged results.

Combined models (using four different initializations) perform significantly better than single PLSA models. The word based error rates are 2.26% vs. 3.10% for full forms, and 2.30% vs. 3.05% for n-grams. Each experiment using four different random initializations is repeated four times, averaged results are reported.

Overall, TopSeg and TopSeg-C perform much better than TextTiling. The best result of 2.26% is a 61% reduction in error rate compared to TextTiling using n-grams.

Error rates for TopSeg using full forms and TopSeg using n-grams are almost identical. However, processing fullforms is much faster because the vocabulary generated from the training set only contains 38,358 different full forms, while it contains 133,362 different n-grams with $f > 1$. Computation times on a 1.7 GHz Pentium-III running Linux are as follows. For full forms, training one PLSA model with 256 classes and 20 EM iterations takes approx. 2 minutes, performing segmentation on the Arabic test set with 500 documents takes approx. 13 minutes. For n-grams, training takes approx. 9 minutes, segmentation approx. 52 minutes.

5. Conclusion

Ideally, Arabic document topic analysis would be based on a uniquely identified stem for each word, on a training collection with long documents with manually assigned segment boundaries, on manually assigned topic labels, and on manually assigned keywords word the document and its segment. Until such resorces are available, we use unlabeled documents and either full-forms or character n-grams instead. We applied our segmentation system TopSeg to Arabic newswire texts, yielding a 61% error reduction compared to TextTiling, a state-of-the-art approach for English. Our best system, using a combination of PLSA models with different random initializations, achieved an error rate of 2.26%. The system achieved approximately the same error rate when using full forms and when using n-grams as terms.

6. References

Ayanendranath Basu, Ian R. Harris, and Srabashi Basu. 1997. Minimum distance estimation: The approach using density-based distances. In G. S. Maddala and C. R. Rao, editors, *Handbook of Statistics*, volume 15, pages 21–48. North-Holland.

Doug Beeferman, Adam Berger, and John Lafferty. 1999. Statistical models for text segmentation. *Machine Learning*, 34:177.

Kenneth R. Beesley. 1996. Arabic finite-state morphological analysis and generation. In *Proceedings of COLING-96*.

Freddy Y. Y. Choi, Peter Wiemer-Hastings, and Johanna Moore. 2001. Latent semantic analysis for text segmentation. In Lillian Lee and Donna Harman, editors, *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing*, pages 109–117.

Freddy Choi. 2000. Advances in domain independent linear text segmentation. In *Proceedings of NAACL-2000*, pages 26–33, Seattle, WA.

Marti A. Hearst. 1997. Texttiling: Segmenting text into multi-paragraph subtopic passages. *Computational Linguistics*, 23(1):33–64.

Thomas Hofmann. 1999. Probabilistic latent semantic indexing. In *Proceedings of SIGIR-99*, pages 35–44, Berkeley, CA.

T. Kailath. 1967. The divergence and bhattacharyya distance measures in signal selection. *IEEE Trans. Commun. Tech.*, COM-15:52–60.

Hang Li and Kenji Yamanishi. 2000a. Topic analysis using a finite mixture model. In *Proceedings of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 35–44.

Hang Li and Kenji Yamanishi. 2000b. Topic analysis using a finite mixture model. *IPSJ SIGNotes Natural Language (NL)*, 139(009).

Hassan Sawaf, Jorg Zaplo, and Hermann Ney. 2001. Statistical classification methods for arabic news articles. In *Proceedings of the ACL/EACL Workshop on ARABIC Language Processing: Status and Prospects*, Toulouse, France.

Evaluating Arabic Retrieval from English or French Queries: The TREC-2001 Cross-Language Information Retrieval Track

Douglas W. Oard[†], Fredric C. Gey[*] and Bonnie J. Dorr[⋆]

[†]College of Information Studies and Institute for Advanced Computer Studies
University of Maryland, College Park, MD 20742
oard@glue.umd.edu
[*]UC DATA
University of California, Berkeley, CA
gey@ucdata.berkeley.edu
[⋆]Computer Science Department and Institute for Advanced Computer Studies
University of Maryland, College Park, MD 20742
bonnie@umiacs.umd.edu

Abstract

The Cross-language information retrieval track at the 2001 Text Retrieval Conference (TREC-2001) produced the first large information retrieval test collection for Arabic. The collection contains 383,872 Arabic news stories, 25 topic descriptions in Arabic, English and French from which queries can be formed, and manual (ground truth) relevance judgments for a useful subset of the topic-document combinations. This paper describes the way in which the collection was created, explains the evaluation measures that the collection is designed to support, and provides an overview of the results from the first set of experiments with the collection. The results make it possible to draw some inferences regarding the utility of the collection for *post hoc* evaluations.

1. Introduction

For the Tenth Text Retrieval Conference (TREC-2001), the U.S. National Institute of Standards and Technology (NIST) developed the first large Arabic information retrieval test collection. This was the eighth year in which non-English document retrieval was evaluated at TREC, and the fifth year in which cross-language retrieval has been the principal focus of that work. Prior TREC evaluations have explored retrieval from Spanish, Chinese, French, German, and Italian document collections. Retrieval from European-language collections is now evaluated in the Cross-Language Evaluation Forum (CLEF) (Peters, 2001), and retrieval from Asian languages is now evaluated at the NTCIR Evaluation (Kando, 2001).

Information retrieval test collections at TREC are designed to model the automatic portion of an interactive search process. They consist of a set of documents to be searched, a set of topics for which relevant documents are to be found, and a set of judgments that identify the documents known to be relevant. In the TREC-2001 Cross-Language Information Retrieval (CLIR) task, the goal of each team was to use English, French, or Arabic queries to rank the set of Arabic documents in order of decreasing likelihood of relevance to the query. In this paper, we describe how the three components of the test collection were created, describe some characteristics of the collection that were observed in TREC-2001 experiments by ten research teams, and and give an overview of the retrieval techniques that those teams explored. The paper concludes with some brief remarks about plans for future development of this test collection.

2. Test Collection

As in past TREC CLIR evaluations, the principal task was to match topics in one language (English or French, in this case) with documents in another language (Arabic) and return a ranked list of the top 1,000 documents associated with each topic. Participating teams were allowed to submit as many as five runs, with at least one using only the title and description field of the topic description. Evaluation then proceeded by pooling the highly-ranked documents from multiple runs and manual examination of the pools by human judges to decide binary (yes/no) relevance for each document in the pool with respect to each topic. A suite of statistics were then calculated, with the mean (over 25 topics) uninterpolated average precision being the most commonly reported.[1]

2.1. Topics

Twenty-five topic descriptions (numbered AR1-AR25) were created in English in a collaborative process between the Linguistic Data Consortium (LDC) and NIST. An example of one of the topic descriptions used in the evaluation is:

<top>
<num> Number: AR22
<title> Local newspapers and the new press law in Jordan
<desc> Description:
Has the Jordanian government closed down any local newspapers due to the new press law?

[1]Uninterpolated average precision is the mean over the ranks of the relevant documents for a topic of the density of relevant documents at or above that rank.

<narr> Narrative:
Any articles about the press law in Jordan and its effect on the local
newspapers and the reaction of the public and journalists toward the new
press law are relevant. The articles that deal with the personal suffering
of the journalists are irrelevant.
</top>

Through the efforts of Edouard Geoffrois of the French Ministry of Defense, the English topics were translated into French and made available to participants which wished to test French to Arabic retrieval. The French version of the topic shown above is:

<top>
<num> Number: AR22
<title> Les journaux locaux et la nouvelle loi sur la presse en Jordanie
<desc> Description:
Le gouvernement jordanien a-t-il interdit un journal local à cause de la nouvelle loi sur la presse?
<narr> Narrative:
Tout article concernant la loi sur la presse en Jordanie et ses effets sur les journaux locaux ainsi que la réaction du public et des journalistes à la nouvelle loi sur la presse est pertinent. Les articles traitant des souffrances personnelles des journalistes ne sont pas pertinents. </top>

The LDC also prepared an Arabic translation of the topics, so participating teams also had the option of doing monolingual (Arabic-Arabic) retrieval. Participating research teams were responsible for forming queries from the topic descriptions using either automatic or manual techniques. Any technique that did not involve human intervention in the formulation of specific queries was classified as automatic. The most common automatic technique was to use all of the words in some set of fields, often the title and description fields. Manual runs were those cases in which people formed queries by hand. All are available on the TREC Web site at http://trec.nist.gov/data.

2.2. Documents

The document collection used in the TREC-2001 CLIR track consisted of 383,872 newswire stories that appeared on the Agence France Press (AFP) Arabic Newswire between 1994 and 2000. The documents were represented in Unicode and encoded in UTF-8, resulting in a 896 MB collection. A typical document is shown in Figure 1. The document collection is distributed by the LDC as Catalog Number LDC2001T55 using one of three arrangements:

- Organizations with membership in the Linguistic Data Consortium (for 2001) may order the collection at no additional charge.[2]

<DOC>
<DOCNO>20000321_AFP_ARB.0001</DOCNO>
<HEADER>ا ر0100 4 ش 8920 قمر /اف-ذرر03 اسرائيل/فلسطينيون</HEADER>
<BODY>
<HEADLINE>جرح ثلاثة اسرائيليين اصابة اثنين منهم خطيرة في هجوم في الضفة الغربية</HEADLINE>
<TEXT>
<P>القدس 12-3 (اف ب) - افادت حصيلة جديدة للجيش الاسرائيلي ان ثلاثة اسرائيليين جرحوا مساء امس الاثنين في هجوم جرى عندما اطلق .عليهم الرصاص من سيارة تجاوزت السيارة المدنية التي كانت تقلهم قرب ترقومية في محيط الخليل بالضفة الغربية</P>
<P>واوضح المتحدث باسم الجيش الاسرائيلي ان سائق السيارة التي كانت تقل الاسرائيليين، وهو من مستوطني الضفة الغربية اصيب بجروح .".طفيفة، ووصف حالة احد الجريحين الاخرين بانها "حرجة" وحالة الثاني بانها "خطيرة</P>
<P>ويشكل الخليل حيث يقيم 004 مستوطن يهودي بحماية الجيش الاسرائيلي وسط 021 الف فلسطيني، بؤرة توتر بين الاسرائيليين والعرب. وقد انسحبت اسرائيل في كانون الثاني/يناير 7991 من 08% من هذه المدينة وابقت على وجود عسكري كبير في الحي الذي يسكنه المستوطنون .</P>
<P>وحرح الاسرائيليون الثلاثة عندما تعرضت السيارة التي كانوا فيها للاطلاق نار من سيارة اجرى تجاوزتها قرب بلدة نرقومية الذي يؤدي اليها "المعبر" الذي يربط بين غزة وجنوب الضفة الغربية مرورا بالاراضي الاسرائيلية "الامن".</P>
<P>وقد نقل الجريحان بسيارة اسعاف ثم بمروحية الى مستشفى حداسا في القدس.</P>
<P>وبدأ الجيش عمليات بحث عن القاتلين واقام حواجز على الطرقات.</P>
<P>وابلغت السلطة الفلسطينية بملابسات الهجوم لتحاول العثور على مرتكبيه.</P>
<P>واشاد المسؤولو الاسرائيليون في الفترة الاخيرة بالتعاون مع اجهزة الامن الفلسطينية في اطار مكافحة الارهاب.</P>
<P>ونشرت بلدية مستوطنة كريات اربع القريبة من الخليل بيان احتجاج على سياسة السلام التي ينتهجها رئيس الوزراء الاسرائيلي ايهود باراك الذي .".يتهمه "بترك المستوطنين رهائن بايدي الفلسطينيين</P>
<P>وقال الجيش الاسرائيلي في تقديرات اولية ان خلية تابعة لحركة المقاومة الاسلامية (حماس) قد تكون وراء الاعتداء.</P>
<P>وتعارض حركة حماس بشدة اتفاقات اوسلو حول الحكم الذاتي الفلسطيني الصرمة عام 3991 وقد اعلنت مسؤوليتها عن عمليات الاعتداءات .الذي استهدفت اسرائيل منذ ذلك الحين</P>
</TEXT>
<FOOTER>شف/ا ا موا004 اف ـــــــــــــــــــ</FOOTER>
</BODY>
<TRAILER>405012 حمت مار 00</TRAILER>
</DOC>

Figure 1: An Arabic document from the collection.

- Non-members may purchase rights (that do not expire) to use the collection for research purposes for $800.
- The Linguistic Data Consortium may be able to negotiate a license at no cost for research groups that are unable to pay the $800 fee, but in such cases the scope and term of the license would be limited to a specific research project.

3. Relevance Judgments

The ten participating research teams shown in Table 1 together produced 24 automatic cross-language runs with English queries, 3 automatic cross-language runs with French queries, 19 automatic monolingual runs with Arabic queries, and 2 manual runs (one with English queries and one with Arabic queries). From these, 3 runs were selected from each team in a preference order recommended by the participants for use in forming assessment pools. The resulting pools were formed from 15 cross-language runs with English queries (14 automatic and 1 manual), and 15 monolingual runs with Arabic queries (14 automatic and 1 manual). The top-ranked 70 documents for a topic in each of the 30 ranked lists were added to the judgment pool for that topic, duplicates were removed, and the documents then sorted in a canonical order designed to prevent the human judge from inferring the rank assigned to a document by any system. Each document in the pool was then judged for topical relevance, usually by the person that had originally written the topic statement. The mean number of relevant documents that were found for a topic was 165. The relevance judgments are available on the TREC Web site at http://trec.nist.gov/data.

Most documents remain unjudged when pooled relevance assessments are used, and the usual procedure is to treat unjudged documents as if they are not relevant. Voorhees has shown that the preference order between automatic runs in the TREC ad hoc retrieval task would rarely be reversed by the addition of missing judgments, and that the relative reduction in mean uninterpolated average precision that would result from removing "uniques" (relevant documents found by only a single system) from the judgment pools was typically less than 5% (Voorhees, 1998). As Figure 2 shows, this effect is substantially larger in the TREC-2001 Arabic collection, with 9 of the 28 judged automatic runs experiencing a relative reduction in mean uninterpolated average precision of over 10% relative when

[2]Information about joining the LDC is available at http://www.ldc.upenn.edu/

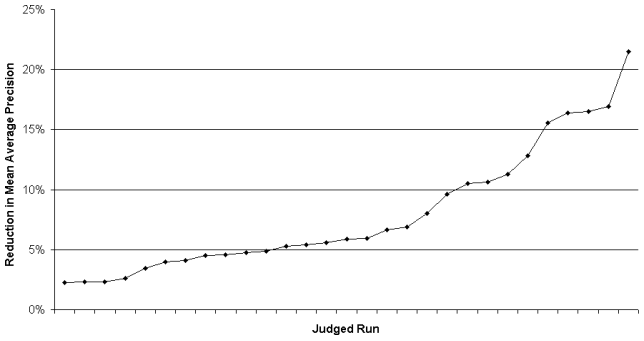


Figure 2: Effect on 29 judged runs of removing "uniques" contributed by that run.

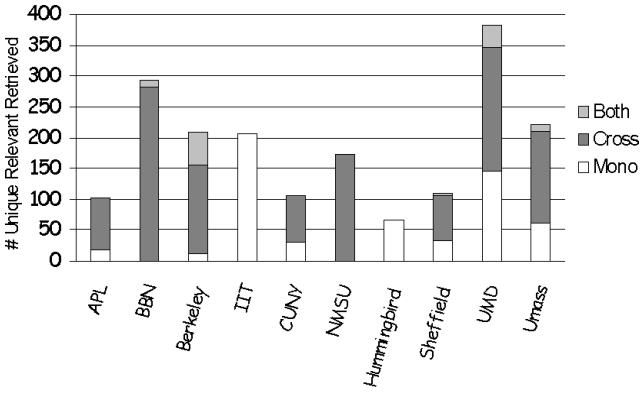


Figure 3: Unique relevant documents, by research team.

the "uniques" contributed by that run were removed from the judgment pool.

Figure 3 helps to explain this unexpected condition, illustrating that many relevant documents were found by only a single participating research team. For 7 of the 25 topics, more than half of the known relevant documents were ranked in the top-70 in runs submitted by only a single research team. For another 6 of the 25 topics, between 40 and 50 percent of their relevant documents were ranked in the top-70 by only one team.

These results show a substantial contribution to the relevance pool from each site, with far less overlap than has been typical in previous TREC evaluations. This limited degree of overlap could result from the following factors:

- A preponderance of fairly broad topics for which many relevant documents might be found in the collection. The average of 165 relevant documents per topic is somewhat greater than the value typically seen at TREC (100 or so).
- The limitation of the depth of the relevance judgment pools to 70 documents (100 documents per run have typically been judged in prior TREC evaluations).
- The diversity of techniques tried by the participating teams in this first year of Arabic retrieval experiments at TREC, which could produce richer relevance pools.
- A relatively small number of participating research teams, which could interact with the diversity of the techniques to make it less likely that another team would have tried a technique that would find a similar set of documents.

Team	Arabic Terms Indexed			
	Word	Stem	Root	n-gram
BBN		X		
Hummingbird		X		
IIT	X	X	X	
JHU-APL	X			X
NMSU	X	X		
Queens	X			X
UC Berkeley		X		
U Maryland	X	X	X	X
U Mass	X	X		
U Sheffield	X			

Table 1: Indexing terms tested by participating teams.

Team	Query Lang	Translation Resources Used			
		MT	Lexicon	Corpus	Translit
BBN	A,E	X	X	X	
Hummingbird	A				
IIT	A,E	X	X		
JHU-APL	A,E,F	X			
NMSU	A,E		X		
Queens	A,E	X			
UC Berkeley	A,E	X	X		
U Maryland	A,E	X			X
U Mass	A,E	X	X		
U Sheffield	A,E,F	X			

Table 2: Translation resources used by participating teams.

The first two factors have occasionally been seen in information retrieval evaluations based on pooled assessment methodologies (TREC, CLEF, and NTCIR) without the high "uniques" effect observed on this collection. We therefore suspect that the dominant factors in this case may be the last two. But until this cause of the high "uniques" effect is determined, relative differences of less than 15% or so in unjudged and post hoc runs using this collection should be regarded as suggestive rather than conclusive. There is, of course, no similar concern for comparisons among judged runs since judgments for their "uniques" are available.

As has been seen in prior evaluations in other languages, manual and monolingual runs provided a disproportionate fraction of the known relevant documents. For example, 33% of the relevant documents that were found by only one team were found only by monolingual runs, while 63% were found only by cross-language runs.

4. Results

Tables 1 and 2 summarize the alternative indexing terms, the query languages, and (for cross-language runs) the sources of translation knowledge that were explored by the ten participating teams. Complete details of each

team's runs can be found in the TREC-2001 proceedings (Voorhees and Harman, 2001), so in this paper we provide only a brief summary of the approaches that were tried. All ten participating teams adopted a "bag-of-terms" technique based on indexing statistics about the occurrence of terms in each document. A wide variety of specific techniques were used, including language models, hidden Markov models, vector space models, inference networks, and the PIRCS connectionist network. Four basic types of indexing terms were explored, sometimes separately and sometimes in combination:

Words. Indexing word surface forms found by tokenizing at white space and punctuation requires no language-specific processing (except, perhaps, for stopword removal), but potentially desirable matches between morphological variants of the same word (e.g., plural and singular forms) are precluded. As a result, word indexing yielded suboptimal retrieval effectiveness (by the mean uninterpolated average precision measure). Many participating research teams reported results for word-only indexing, making that condition useful as a baseline.

Stems. In contrast to English, where stems are normally obtained from the surface form of words by automatically removing common suffixes, both prefixes and suffixes are normally removed to obtain Arabic stems. Participating teams experimented with stemming software developed at three participating sites (IIT, NMSU, and U Maryland) and from two other sources (Tim Buckwalter and Shereen Khoja).

Roots. Arabic stems can be generated from a relatively small set of root forms by expanding the root using standard patterns, some of which involve introduction of infixes. Stems generated from the same root typically have related meanings, so indexing roots might improve recall (possibly at the expense of precision, though). Although humans are typically able to reliably identify the root form of an Arabic word by exploiting context to choose between alternatives that would be ambiguous in isolation, automatic analysis is a challenging task. Two participating teams reported results based on automatically determined roots.

Character n-grams. As with other languages, overlapping character n-grams offer a useful alternative to techniques based on language-specific stemming or morphological analysis. Three teams explored n-grams, with values of n ranging from 3–6.

Term formation was typically augmented by one or more of the following additional processing steps:

Character deletion. Some Unicode characters, particularly diacritic marks, are optional in Arabic writing. This is typically accommodated by removing the characters when they are present, since their presence in the query but not the document (or vice-versa) might prevent a desired match.

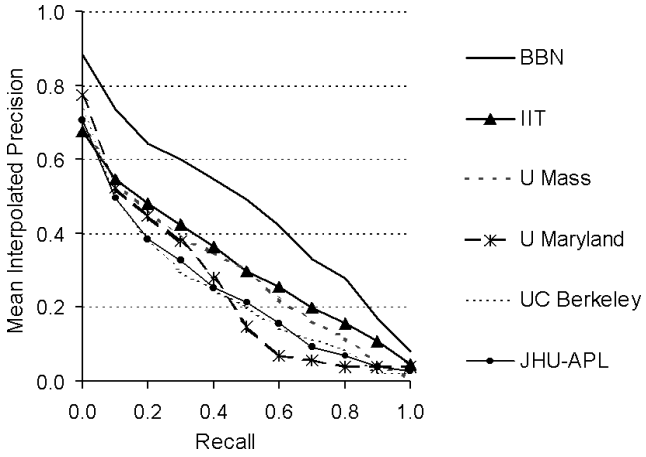


Figure 4: Cross-language retrieval effectiveness, English queries formed from title+description fields, automatic runs.

Character normalization. Some Arabic letters have more than one Unicode representation because their written form varies according to morphological and morphotactic rules, and in some cases authors can use two characters interchangeably. These issues are typically accommodated by mapping the alternatives to a single normalized form.

Stop-term removal. Extremely frequent terms and other terms that system developers judge to be of little use for retrieval are often removed in order to reduce the size of the index. Stop-term removal is most commonly done after stemming or morphological analysis in Arabic because the highly productive morphology would otherwise result in impractically large stopword lists.

Nine of the ten participating research teams submitted cross-language retrieval runs, with all nine using a query-translation architecture. Both of the teams that tried French queries used English as a pivot language for French-to-Arabic query translation, so English-to-Arabic resources were key components in every case. Each team explored some combination of the following four types of translation resources:

Machine Translation Systems. Two machine translation systems were used: (1) a system developed by Sakhr (available at http://tarjim.ajeeb.com, and often referred to simply as "Ajeeb" or "Tarjim"), a system produced by ATA Software Technology Limited (available at http://almisbar.com, and sometimes referred to as "Almisbar" or by the prior name "Al-Mutarjim"). At the time of the experiments, both offered only English-to-Arabic translation. Some teams used a machine translation system to directly perform query translation, others used translations obtained from one or both of these systems as one source of evidence from which a translated query was constructed. A mark in the "MT" column of Table 2 indicates that one or more existing machine translation systems were used in some way, not that they were necessarily used

to directly perform query translation.

Translation Lexicons. Three commercial machine readable bilingual dictionaries were used: one marketed by Sakhr (also sometimes referred to as "Ajeeb"), one marketed by Ectaco Inc., (typically referred to as "Ectaco"), and one marketed by Dar El Ilm Lilmalayin (typically referred to as "Al Mawrid"). In addition, one team (NMSU) used a locally produced translation lexicon.

Parallel Corpora. One team (BBN) obtained a collection of documents from the United Nations that included translation-equivalent document pairs in English and Arabic. Word-level alignments were created using statistical techniques and then used as a basis for determining frequently observed translation pairs.

Transliteration. One team (Maryland) used pronunciation-based transliteration to produce plausible Arabic representations for English terms that could not otherwise be translated.

When multiple alternative translations were known for a term, a number of techniques were used to guide the combination of evidence, including: (1) translation probabilities obtained from parallel corpora, (2) relative term frequency for each alternative in the collection being searched, and (3) structured queries. Pre-translation and/or post-translation query expansion using blind relevance feedback techniques and pretranslation stop-term removal were also explored by several teams.

To facilitate cross-site comparison, teams submitting automatic cross-language runs were asked to submit at least one run in which the query was based solely on the title and description fields of the topic descriptions. Figure 4 shows the best recall-precision curve for this condition by team. All of the top-performing cross-language runs used English queries.

As is common in information retrieval evaluations, substantial variation was observed in retrieval effectiveness on a topic-by-topic basis. Figure 5 illustrates this phenomenon over the full set of cross-language runs (i.e., not limited to title+description queries). For example, half of the runs did poorly on topic AR12, which included specialized medical terminology, but at least one run achieved a perfect score on that topic. Five topics, by contrast, turned out to be problematic for all systems (AR5, AR6, AR8, AR15, and AR23). Examining retrieval effectiveness on such topics may help researchers identify opportunities to improve system performance.

No standard condition was required for monolingual runs, so Figure 6 shows the best monolingual run by team regardless of the experiment conditions. Several teams observed surprisingly small differences between monolingual and cross-language retrieval effectiveness. One site (JHU-APL) submitted runs under similar conditions for all three topic languages, and Figure 7 shows the resulting recall-precision graphs by topic language. In that case, there is practically no difference between English-topic and Arabic-topic results. There are two possible explanations for this widely observed effect:

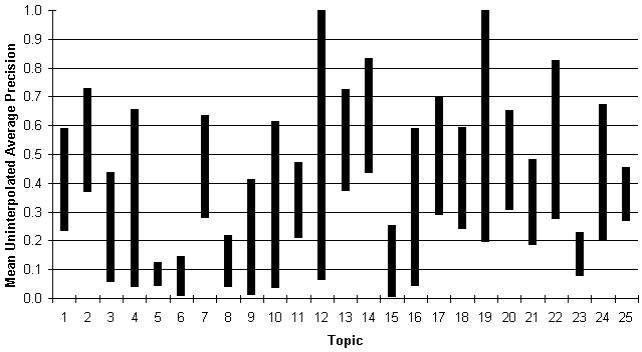


Figure 5: Cross-language topic difficulty, uninterpolated average precision (base of each bar: median over 28 runs, top of each bar: best of the 28 runs).

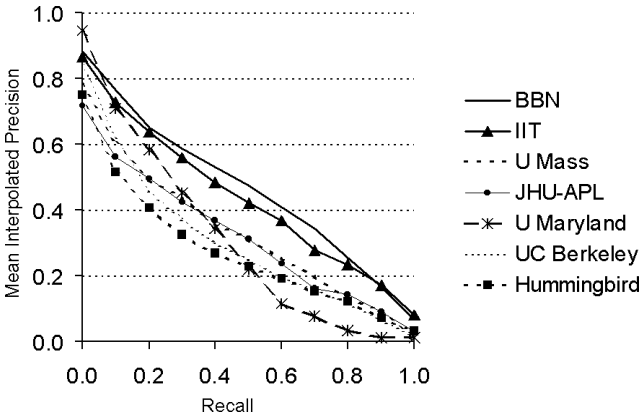


Figure 6: Monolingual retrieval effectiveness, Arabic queries formed from title+description fields (except JHU-APL and UC Berkeley, which also used the narrative field), automatic runs (except U Maryland, which was a manual run designed to enhance the relevance assessment pools).

- No large Arabic information retrieval test collection was widely available before this evaluation, so the monolingual Arabic baseline systems created by participating teams might be improved substantially in subsequent years.
- The 25 topics used in this year's evaluation might represent a biased sample of the potential topic space. For example, relatively few topic descriptions this year included names of persons.

Several teams also observed that longer queries did not yield the improvements in retrieval effectiveness that would normally be expected. One site (Hummingbird) submitted runs under similar conditions for three topic lengths, and Figure 8 shows the resulting recall-precision graphs. In this case, longer queries showed no discernible benefit; indeed, it appears that the best results were achieved using the shortest queries! The reasons for this effect are not yet clear, but one possibility is that the way in which the topic descriptions were created may have resulted in a greater concentration of useful search terms in the title field. For example, the title fields contains an average of about 6 words, which is about twice as long as is typical for TREC.

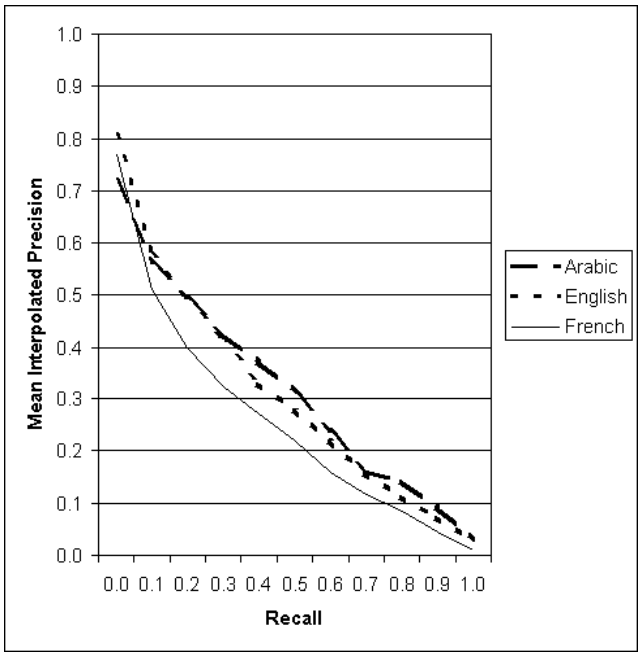


Figure 7: Topic language effect, title+description+ narrative.

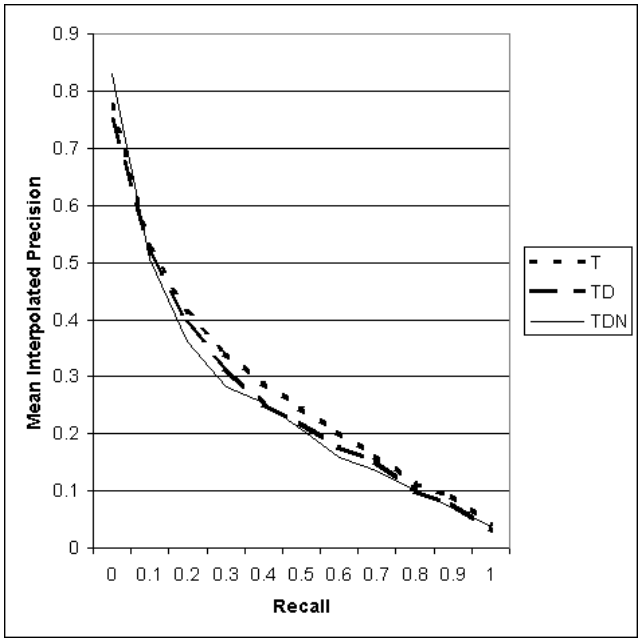


Figure 8: Query length effect, Arabic queries. (T=title, D=Description, N=Narrative).

5. Summary and Outlook

The TREC-2001 CLIR track focused on searching Arabic documents using English, French or Arabic queries. In addition to the specific results reported by each research team, the evaluation produced the first large Arabic information retrieval test collection. A wide range of index terms were tried, some useful language-specific processing techniques were demonstrated, and many potentially useful translation resources were identified. In this paper we have provided an overview of that work in a way that will help readers recognize similarities and differences in the approaches taken by the participating teams. We have also sought to explore the utility of the test collection itself, providing aggregate information about topic difficulty that individual teams may find useful when interpreting their results, identifying a potential concern regarding the completeness of the pools of documents that were judged for relevance, and illustrating a surprising insensitivity of retrieval effectiveness to query length.

The TREC-2002 CLIR track will continue to focus on searching Arabic. We plan to use 50 new topics (in the same languages) and to ask participating teams to also rerun the 25 topics from this year with their improved systems as a way of further enriching the existing pools of documents that have been judged for relevance. We expect that the result with be a test collection with enduring value for post hoc experiments, and a community of researchers that possess the knowledge and resources needed to address this important challenge.

Acknowledgments

We are grateful to Ellen Voorhees for coordinating this track at NIST and for her extensive assistance with our analysis and to the participating research teams for their advice and insights along the way.

6. References

Noriko Kando, editor. 2001. *Proceedings of the Second NTCIR Workshop Meeting on Evaluation of Chinese and Japanese Text Retrieval and Text Summarization (NTCIR-2)*, Tokyo. National Institute of Informatics. http://research.nii.ac.jp/ntcir.

Carol Peters, editor. 2001. *Cross-Language Information Retrieval and Evaluation*. Springer: Lecture Notes in Computer Science: LNCS 2069. http://www.clef-campaign.org.

E. M. Voorhees and D. K. Harman, editors. 2001. *The Tenth Text REtrieval Conference (TREC-2001)*, Gaithersburg, MD. National Institute of Standards and Technology, Department of Commerce. http://trec.nist.gov.

Ellen M. Voorhees. 1998. Variations in relevance judgments and the mesaurement of retrieval effectiveness. In C.J. Van Rijsbergen W. Bruce Croft, Alistair Moffat, editor, *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 315–323. ACM Press, August.

Logical Aspects of Unification Based-Arabic Semantic Construction

Bassam Haddad†, Mustafa Yaseen† and Mudhafar Al-Jarrah†

† Faculty of Information Technology
Amman University, Jordan
{haddad, myaseen, maljarrah}@ammanu.edu.jo

Abstract

This paper addresses issues related to employing logic-based semantic construction as meaning representation formalism for Arabic. Since semantic formalism has to be compositional on the level of semantic representation, λ-conversion based on the Discourse Representation Theory can be utilized for realization of semantic construction for Arabic.

Keywords: Logical Form, Arabic Semantic Representation, λ-DRT, Unification-Based Semantics, HDPSG

1. Introduction

For the last two decades concentration on Arabic processing has focused on Arabic from the *morphological* and *syntactical* points of view. In this field, significant progress has been reported (Beesely 2001; Ouersighni 2001; Ditters 2001; Al-Fedaghi and Al-Anzi 1989) and many others.

Despite the importance of *semantic processing* for achieving the *understanding capability*, there was little work reported on *semantic representation* and *semantic analysis* of Arabic (Haddad and Yaseen, 2001; Khayat 1988; Al-Johar and McGregor 1997; El-Dessouki et al. 1988, Al-Muhtaseb and Mellish, 1997) and others. Therefore, we believe that there is an ultimate need to make more effort to develop an *adequate model* for *semantic processing* for Arabic, even though there is no existing *formal theory* capable to provide a complete and consistent account of all phenomena involved in Arabic semantic processing.

Semantic processing has to carry out different tasks on different levels to achieve the *understanding facility*. One of the most important of these levels is the *construction* and *composition* of *meaning representation formalisms* for Arabic sentences. This semantic level plays a decisive role for other semantic processing steps, i.e. *semantic resolution* and *evaluation.*

Lexical and *unification*, especially HDPSG related system development is ongoing in numerous university and industrial settings for different languages. HDPSG is based on GPSG, LFG and Categorial Grammar. In such a grammar the *lexicon* plays a pivotal role, where *semantics* and *syntax* can be integrated in the same grammar. A central concept of a *unification-based grammar* is the notion *"feature"* or *"attribute"* which is characterized *by feature value pairs.*

Applying the operation of *unification* to *two compatible feature values structures* yields a new *feature value structure* containing *all the information* involved in the two original feature structures. The importance of the concept might be residing in the fact that we can solve the problem of finding the right level of *granularity* in classifying words into categories having *internal structures*. Furthermore, the *unification* allows us to combine information from *multiple feature structures*, as long as it is consistent (Pollard and Sag 1994; Bender et al. 1999).

Simulating the *λ-conversion process* in a *feature logical formalism* within a *unification-based grammar* such as HDPSG enables a realization of *unification based semantic construction formalism for Arabic.*

Inspired by the work of (Bos et al. 1994) we propose in this paper to integrate the semantic construction model presented in (Haddad and Yaseen, 2001) in a *unification-based semantic Grammar.*

2. Logical Semantic Representation

Assuring the *modularity constraint* in a natural language understanding system requires a *compositional semantic formalism* on the level of meaning representation. Despite the fact that *predicate logic represents* well-studied formal representation formalism, it does not provide any *compositional facilities. λ-abstraction* offers an important framework for achieving such a goal in particular for the meaning construction of Arabic sentences (Haddad and Yaseen 2001; Pinkal 1995; Montague 1974).

In this context we have achieved some success in developing a model for *construction of meaning rep-*

resentation forms for Arabic sentences. Based on some *compositional rules* expressing the meaning of *syntactical categories of Arabic*, our approach employs a *λ-conversion process* to construct *logical forms representing the meaning of Arabic sentences* (Haddad and Yaseen, 2001).

In this model *determiners* play a central role in constructing *semantic constituents*. For example, the Arabic determiners such as "ال$_n$", "كل", "بعض", etc., could be considered as *quantifiers*. Generally the meaning of a quantifier, ||Quant||, can be expressed as follows:

$$||\text{Quant}|| \Rightarrow \lambda R \lambda S(Quantifier(R,S) \quad (1)$$

The definite determiner "ال$_1$" combines in general two things together: a *restriction R and a scope S:*

$$||\text{ال}_1|| \Rightarrow \lambda R \lambda S(\text{ال}_1\ (x, R \wedge S)) \quad (2)$$

The following example in *"figure 1, three-branch quantifier tree representation"* might illustrate the basic concept of this approach. Details about this concept are found in [Haddad and Yaseen, 2001].

The *function* of the determiner "ال$_1$" in the sentence "يتعلم الولد العربيه" can be formulated as follows:

$$||\text{VS}|| \xrightarrow{sem} ||\text{Subj}||\ (||\text{Obj}||\ (||\text{Verb}||)) \quad (3)$$

Applying of (3) to ||ال$_1$|| yields the following logical representation:

$$\lambda R \lambda S(\text{ال}_1\ (x, R \wedge S))\ (||\text{ولد}||)\ (||\text{يتعلم ال عربية}||) \quad (4)$$

$$\lambda S(\text{ال}_1\ (x, \text{ولد}(x) \wedge S))\ (||\text{يتعلم ال عربية}||) \quad (5)$$

$$\text{ال}_1\ (x, \text{ولد}(x) \wedge \text{ال}_1\ (y, \text{عربية}(y) \wedge \text{يتعلم}\ (x, y))) \quad (6)$$

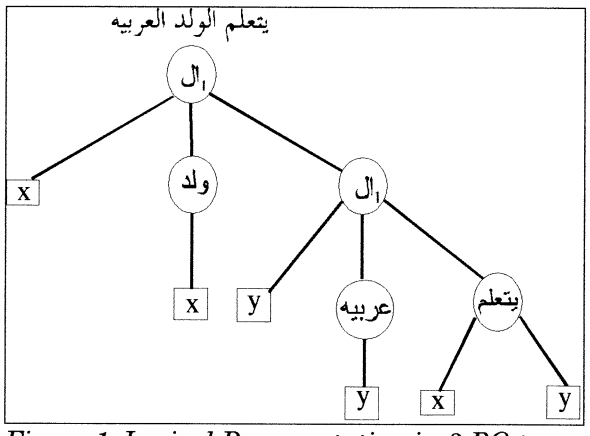


Figure 1: Logical Representation in 3-BQ tree

3. DRT-Based Compositional Aspects for Arabic

Despite the importance of logic-based *compositional models* for achieving Arabic understanding, such methods are rather constructed to deal with Arabic *sentence semantics* and in general they are *inappropriate* for treating *text semantics*.

The *Discourse Representation Theory (DRT)* is capable of capturing problems involved in representing *anaphoric aspects* and *text semantics* (Kamp, 1981; Bende-Farkas and Kamp, 2001).

In this approach the semantic function of sentences consists in constructing of *Discourse Representation Structures (DRS)* by applying certain *DRS construction rules dynamically* within the context of the *referents* in a text.

For instance, the function of a *definite article* seems in the view of *DRT*, not in interpreting it as a *unique quantifier*. It has rather to be understood as a *referent to a certain object in a nominal expression.* Moreover, the interpretation of the *indefinite article* appears *in the first place* not to be as an *existential quantifier. An indefinite article introduces* rather a *new referent to the context.*

In addition, one of the most important aspects of *DRT* is its interesting interpretation of *pronouns*. The interpretation of *a pronoun* is not *a variable,* which has to be *locally bound,* but much more as a *definite label* with the function of *making a reference to a previously introduced discourse referent*. Therefore, *a DRT-based semantic construction of Arabic* has to be in the first place not in constructing the logical meaning in an *isolated mode* but much more in a *dynamic and modifiable one.*

Example:

تدرس ماريا لغه تحبها

(Maria studies a language she likes)

The interpretation of this *discourse* starts with an empty *DRS*. After interpretation of the first part of the sentence "تدرس ماريا لغة" (Maria studies a language), the *DRS* is expanded by adding the next *referents* and *conditions*. The referent *e* represents an *event of studying* "تدرس". The *referent n* is used to denote the *time of speech* (see the following figures):

x y e n
"ماريا" *(x)*
لغة*(y)*
e: تدرس*(x,y)*
$n \subseteq e$
DRS for تدرس ماريا لغة

y e n z w s
"تدرس" *(x)*
لغة*(y)*
e: تدرس*(x,y)*
n ⊆ e
s: تحب*(z,w)*
$n \subseteq s$
$z = x$
$w = y$
DRS for تدرس ماريا لغة تحبها

In the final stage of representation the resulting *Discourse Representation Structures* are interpreted in *model theory* based logical forms.

It is obvious that *DRT-based semantic construction* proceeds from another point of view than the *Montague-style* in the construction process and it is therefore *not compositional*. Furthermore, the semantic construction is given in *top-down manner* and is *not declarative*, that means the processing order effects the binding possibilities (Pinkal, 1995).

3.1 Compositional Semantics for ARABIC

The Integration of *lambda conversion* in *DRT* extends *DRT* to be *compositional* without losing the important feature of representing *text anaphoric*. In this approach the semantic function of sentences consists in constructing of *Discourse Representation Structures* by applying some *DRS construction rules* within the context of the *referents* in a text. The DRS_n, for instance, consists of a pair: *a universe of discourse*, DR_n, i.e. *a set of Discourse Referents and a set of conditions*, $COND_n$, *about the* DR_n. An additional feature of the language of *λ-DRT, we adopted the merge operation* ⊗, which combines *two DRS's* by taking the *union* of the sets of discourses and conditions separately (Bos et al., 1994):

$$\langle DR_1, COND_1 \rangle \otimes \langle DR_2, COND_2 \rangle \equiv \langle DR_1 \cup DR_2, COND_1 \cup COND_2 \rangle \quad (7)$$

For example the meaning representation for constructing the *DRS* for sentence "كل طالب يجتهد" could be represented in terms of *λ-DRT as follows (see also (1)):*

$$||\text{كل}|| \Rightarrow \lambda R \lambda S \langle \{x\}, \{ x: Any \} \rangle \otimes R(x) \xrightarrow{\text{كل}} S(x) \quad (8)$$

$$||\text{طالب}|| \Rightarrow \lambda y \langle \{\}, \{y: Individual, \text{طالب}(y)\} \rangle \quad (9)$$

$$||\text{يجتهد}|| \Rightarrow \lambda z \langle \{\}, \{e: Event, z: Individual, \text{يجتهد}(e, z_{<agent>})\} \quad (10)$$

The *DRS* in (10) means, that there is an *event* "يجتهد" which takes an *individual* as an *argument* and plays the role of an *agent*.

Simulating the basic aspects of the *λ-conversion process* presented in [Haddad and Yaseen, 2001] and applying it to the *DRS's* established above would lead in a simplified form to the following semantic representation:

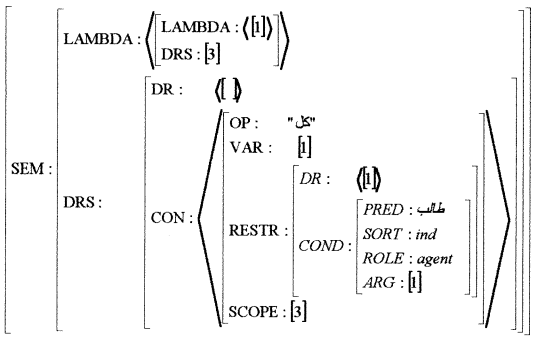
$$\langle \{x\}, \{x: Individual, \text{طالب}(x_{<agent>})\} \rangle \xrightarrow{\text{كل}} \langle \{\}, \{e: Event, \text{يجتهد}(e, x_{<agent>})\} \quad (11)$$

3.2 Unification-based Semantic Construction for Arabic

A *λ-Expression* representing the meaning of an *Arabic constituent* (Haddad and Yaseen, 2001) could be formulated in terms of *feature structures*. Such structures might be represented by a LAMBDA and a DRS *feature structure*. A LAMBDA *feature structure* specifies a *list* of the appropriate *arguments, which are* involved in the expression, while *a DRS feature structure represents the body of the λ-expression.* Furthermore, additional *pragmatic notations* could be *also embedded in the DRS feature structures.* Compositional rules expressing the meaning of syntactical constituents are also integrated in the *lexical entries* of a *DRS*.

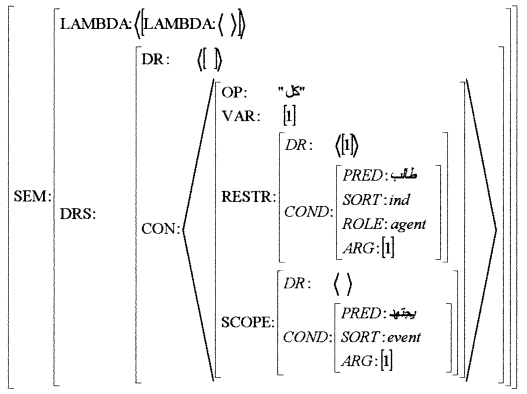
A *unification-based semantic construction* can be achieved by *unifying* the values of a *LAMBDA feature structure* with *the representations of the feature structures involved in the arguments*. And then storing the results of the unification in the *DRS feature structure of processed syntactical constituent*. This process corresponds to *λ-conversion proposed in* (Bos et al. 1994).

Constructing the meaning of "كل طالب" in the sentence "كل طالب يجتهد" requires the application of the *feature structures* involved in (8) to the *feature structures* in (9) (see also "figure 1" and (3), (4), (5), (6)):



(12)

To construct the meaning of the whole sentence "كل طالب يجتهد" , "DRS: [3]" has to be applied to the composed DRS in (12):



(13)

It is obvious that (13) corresponds to the logical form in (11).

4. Conclusion

Semantic processing is a non-trivial topic in natural language understanding. We believe that the progress that has been made in recent years is also *applicable to Arabic. Semantic construction* is a substantial task in achieving the basic steps of Arabic understanding. Problems involved in treating *text anaphoric* can be treated *dynamically* by simulating *λ-conversion presented in (Haddad and Yaseen, 2001) within an adapted DRS for Arabic.*

Additionally, this paper is an attempt to direct the attention of research concerned with Arabic processing, in particular to the techniques concerned with *DRT-based and unification-based semantic processing.* At present, our research is concerned with developing such a model by adopting these additional features.

5. References

Al-Fedaghi and Al-Anzi, 1989. Al-Fedaghi, S., Al-Anzi, F. *A New Algorithm to Generate Arabic Root-Pattern Forms.* Proceedings of the 11th National Computer Conference, Saudi Arabia, 1989, pp. 391-400.

AL-Johar and McGregor, 1997. Al-Johar, B., McGregor, J. *A Logical Meaning Representation for Arabic (LMRA).* Proceedings of the 15th National Computer Conference, Riyadh, Saudi Arabia, 1997, pp. 31-40.

Al-Muhtaseb and Mellish, 1997. Al-Muhtaseb H., Mellish C. *Towards an Arabic Upper Model: A proposal.* Proceeding of the 15th National Conference, Riyadh, Saudi Arabia 1997.

Beesley, 2001. Kenneth R. Beesley. *Finite-State Morphological Analysis and Generation of Arabic at Xerox Research: Status and Plans 2001.* ACL/EACL01: Conference of the European Chapter, Workshop: Arabic Language Processing: Status and Prospects , 2001

Bende-Farkas and Kamp, 2001. Ágnes Bende-Farkas and Hans Kamp. *Indefinites and Binding: From Specificity to Incorporation*, Lecture Notes, 13th European Summer School In Logic, Language and Information, ESSLLI, 2001.

Bender, Ivan Sag and Thomas Wasow, 1999. *Instructor's Manual for Syntactic Theory: A Formal Introduction.* CSLI Publications, 1999

Bos, J, E. Mastenbroek, S. McGlashan, S Millies and Pinkal, 1994. *A Compositional DRS-based Formalism for NLP Applications* Report 59, VerbMobil, Universität des Saarlandes 1994.

Ditters, 2001. Everhard Ditters, *A Formal Grammar for the Description of Sentences Structures in Modern Standard Arabic.* ACL/EACL01: Conference of the European Chapter, Workshop: Arabic Language Processing: Status and Prospects, 2001

El-Dessouki, A., Nazif, A, O, Ahmad, 1988. *An Expert System for Understanding Arabic Sentences.* Proceeding of the 10th National Computer Conference, Jeddah, Saudi Arabia, 1988, pp 745-759.

Haddad and Yaseen, 2001. Bassam Haddad and Mustafa Yaseen. *Towards Understanding Arabic: A Logical Approach for Semantic Representation.* ACL/EACL01: Conference of the European Chapter, Workshop: Arabic Language Processing: Status and Prospects, 2001

Kamp, 1981. *A Theory of Truth and Semantics Representation.* In J. Groendijek, T. J. Stokhof, eds., Formal Methods in Study of Languages. Mathematish Centrum, Amsterdam, 1981

Khayat, 1988. Khayat, M.G., Al-Muhtaseb, H.A., *Knowledge Representation in Natural Language System.* Proceedings of the 10th National Computer Conference, Jeddah, Saudi Arabia, 1988, pp. 667-677.

Montague, 1973. Richard Montague. *The Proper Treatment of Quantification in Ordinary English.* In: Philosophy, Language , and Artificial Intelligence, ed., J. Kulas, J. H. Fetzer and T. Rankin, Kluwer Academic Publishers, 1988.

Ouersighni, 2001. Riadh Ouersighni. *A major offshoot of the Dinar-MBC project: AraParse, a morphosyntactic analyzer for unvowelled Arabic*

texts. ACL/EACL01: Conference of the European Chapter, Workshop: Arabic Language Processing: Status and Prospects, 2001

Pinkal Manfred, 1995. *Sprachverarbeitung: Semantik.* In Einführung in die Künstliche Intelligenz. Ed. Günther Görz. Addison-Wesley, 1995

Pollard and Sag, 1994. Pollard C. and I.A. Sag. *Head-Driven Phrase Structure Grammar.* Stanford, CSLI, 1994.

Arabic Character Recognition using Approximate Stroke Sequence

Professor Mohammed Zeki Khedher* **Dr. Gheith Abandah¶**
*Dept. of Electrical Engg khedher@ju.edu.jo
¶Dept. of Computer Engg abandah@ju.edu.jo
University of Jordan, Amman – Jordan

Abstract

Arabic character recognition of handwriting is addressed. A novel approach for the Arabic Character Recognition is presented based on statistical analysis of a typical Arabic text is presented. Results showed that the sub-word in Arabic language is the basic pictorial block rather than the word. The method of approximate stroke sequence is applied for the recognition of some Arabic characters in their stand-alone form. This method could be extended further for more accurate results. It is recommended that research in Arabic OCR systems in the future is based on the basis of the sub-word as the basic block rather than the word.

1. Introduction

Automatic recognition of handwriting has become a mature discipline at the beginning of the 21st century. On-line systems are now available on handheld computers with acceptable performance. Off-line systems are less accurate than on-line systems. However, they are now good enough for specialized systems such as interpreting handwritten postal addresses on envelopes and reading currency amounts on bank checks (Plamondon 2000).

The recognition of Arabic characters is particularly difficult due to the necessity of segmentation even for printed text. In order to get an insight into the Arabic word structure, it becomes necessary to do some statistical analysis on some typical Arabic text in order to assess the nature of problems facing the workers on Arabic OCR systems. For this purpose, a reasonable size of Arabic text was selected and analyzed. Based on the results of this analysis, a new procedure is suggested for building Arabic OCR systems. As a first step in the implementation of such systems, recognition of Arabic characters in their stand-alone form is addressed. The method of approximate stroke sequence matching is applied and the results are shown.

The paper gives some literature survey on previous work done in the field. It then gives the main characteristics of Arabic writing, presenting the importance of the sub-word structure of the Arabic word, showing the statistical results proving this phenomena, and proposing a new procedure for Arabic OCR system. The newly proposed method suggests the treatment of the sub-word as the basic block in the recognition of Arabic characters. The size of the sub-word should be treated as a decisive factor in the method of recognition of the characters contained in the sub-word. The method of approximate stroke sequence matching is described and then applied to an example of unknown character and compared with two standard characters. A text containing different shapes of Arabic characters was written by 48 different persons and samples of these characters under test were copied for this study. Some results of applying this procedure onto different characters is given. The paper discusses the results obtained and ends up with some conclusions and suggestions for future work.

2. Previous Work in Arabic Character Recognition

Several good literature review papers were published for various research topics on Arabic character recognition (Jambi, 1991; Ahmed, 1994 and Amin, 1997). A recent comprehensive one is given by Ahmed in 2000. Here we shall give some elaboration on some of the effort spent in this direction.

Classical optical off-line recognition of handwriting is composed of a pre-processing stage, character segmentation, feature extraction and a classification stage (Casey 1996). Pre-processing consists of several operations like thresholding, noise removal, page orientation, skewing of lines removal, line segmentation, word segmentation, and pictures and figures removal. There is little difference in these processes between Arabic language and Latin-alphabet-based languages (Abdulla, 1988; Mahmoud, 1991; Hussain & Cowel, 2000; Hussain & Zalik, 2000).

Work on isolated printed Arabic characters and numerals took a lot of shapes. Overall vertical level of the character compared with the baseline was studied (Talba 1987). The chain code describing the sequence of character strokes using the 8-direction strokes was followed by the majority of researchers (Alshebeli, 1997). However, hexagonal sampled procedure was also found (Khellah, 1994). Different methods for matching the unknown character with the standard characters were followed.

Segmentation of characters is an important step in character recognition for cursive writing whether hand written or printed. There are three strategies for segmentation: the classical approach in which segments identification is based on “character-like” properties, the recognition-based segmentation strategy, in which the system searches the image for components that match classes in its alphabets, and the third strategy is the holistic method, in which the system seeks to recognise words as a whole, thus avoiding the need to segment into characters (Casey 1996) Different algorithms were followed to apply one of the above methods or the other, like hierarchical syntactic procedure (Haj-Hassan, 1990), quadratic discriminating functions (Udpa, 1992), the method of moment invariant algorithm (El-Khaly, 1990), accumulative invariant moment was used as an identifier

in character recognition (El-Dabi, 1990), and even segmentation of printed Arabic characters was tried without the thinning process (El-Sheikh, 1988). Use of clustering technique was chosen for classification (El-Desouky, 1992) or tree representation for the description of various characters (Al-Waily, 1989; Saleh, 1994 & Saleh, 1996). Use of tree representation and fuzzy constrained graph models which tolerate large varieties in writing styles were reported also (Abuharba, 1994).

Recognition of different fonts of Arabic printed text was tried using pre-processing and structural feature extraction (Kavianifar,1998). Parallel Arabic OCR systems were also proposed (Alherbish, 1997).

Hidden Markov models which proved to be very successful in the area of automatic speech recognition was tried in the area of omnifont, open-vocabulary Arabic OCR system (Bazzi, 1999).

Work on limited hand-written Arabic text database was tried. A system based on four types of basic features, namely the end points, corners, the strokes and the branch points gave reasonable results (Jambi, 1991).

Recognition of Typewritten Arabic characters gave good results using external features such as character area ratio, n-th quadrant ratio, vertical line ratio, horizontal line ratio, number of upper edges, and other similar features (Al-Ohali, 1995).

Online character recognition uses the feature extraction process results in a sequential manner which is called the chain code. Treatment of secondary characters (mainly the points above and below the characters) is definitely an integral part of the recognition process (El-Gwad, 1990).

Neural Network was used in some work (Said, 1998 , Al-Kadi, 1995, Altuwaijri, 1995 and Al-Sharaidah, 2000).

3. Main characteristics of Arabic Writing

Arabic text is written from right to left and is always cursive. The shape of an Arabic character changes according to its location in the word. An Arabic character has up to four different shapes; the shape of a character depends on the type of character to its right and its position within the word. Table 1 shows the Arabic character set in the four different shapes.

The Arabic character set is composed of 28 basic characters. Fifteen of them have dots and 13 are without dots. Dots above and below the characters play a major role in distincting some characters that differ only by the number or location of dots. Take the example the letters: ب ت ث ي ن . In their middle form, all these five letters are written the same way as: بـ تـ ثـ يـ نـ . They differ only by the number or the locations of the dots.

There are four characters which may take the secondary character "Hamzah ء". Those are "Alif أ إ ", "Waw ؤ" , "Yaa ئ" and "Kaf ك".

There are also some other secondary characters used above and below the characters to indicate vowels but we shall exclude them now from our discussions.

Arabic characters do not have fixed width or fixed size, even in printed form.

3.1 An important phenomena in Arabic writing

Arabic writing is known to be cursive even in printed form. However, it differs from cursive handwriting of English in that some characters can be connected from one side only. Out of the 28 basic Arabic characters, six can be
connected from the right side only while the other 22 can be connected from both sides. These six characters are:dal (د), raa (ر), waw (و), alef (ا), thal (ذ) , and zay (ز). These six characters have only two forms, the stand-alone form and the final form. Whereas the rest of the characters can appear in any of four forms: the initial, the middle, the final, and the stand-alone form. Consequently, an Arabic word may consist of one or more sub-words. A sub-word can be defined as the basic stand-alone pictorial block of the Arabic writing. Any optical character recognition of Arabic characters should treat the sub-word as the basic block for processing whatever the method it uses for preprocessing, segmentation, recognition, or classification.

This is because each sub-word is separated from other sub-word by a space. Although spaces between sub-words are usually shorter than those between successive words, still they are surrounded by space. A word may contain one or more sub-words. Some of these sub-words may even consist of a single character in its stand-alone form. Hence, their recognition does not need segmentation.

Shape of the letter in the text differs according to the location of the character in the sub-word, i.e. a character at the end of sub-word, has exactly the same shape when it comes at the end of a full word. Take the example of the

Letter	Stand-alone	Initial	Middle	Final	Other shapes
Alef	ا			ـا	ى ى
Ba'	ب	بـ	ـبـ	ـب	
Ta'	ت	تـ	ـتـ	ـت	ة ـة
Tha'	ث	ثـ	ـثـ	ـث	
Jeem	ج	جـ	ـجـ	ـج	
H'a'	ح	حـ	ـحـ	ـح	
Kha'	خ	خـ	ـخـ	ـخ	
Dal	د			ـد	
Thal	ذ			ـذ	
Ra'	ر			ـر	
Zai	ز			ـز	
Seen	س	سـ	ـسـ	ـس	
Sheen	ش	شـ	ـشـ	ـش	
Sad	ص	صـ	ـصـ	ـص	
Dhad	ض	ضـ	ـضـ	ـض	
Tta	ط	طـ	ـطـ	ـط	
Dha'	ظ	ظـ	ـظـ	ـظ	
Ain	ع	عـ	ـعـ	ـع	
Gahin	غ	غـ	ـغـ	ـغ	
Fa'	ف	فـ	ـفـ	ـف	
Qaf	ق	قـ	ـقـ	ـق	
Kaf	ك	كـ	ـكـ	ـك	
Lam	ل	لـ	ـلـ	ـل	
Meem	م	مـ	ـمـ	ـم	
Noon	ن	نـ	ـنـ	ـن	
Ha'	ه	هـ	ـهـ	ـه	
Waw	و			ـو	
Ya'	ي	يـ	ـيـ	ـي	

Table1: The Different Forms of Arabic Alphabets

Char per sub-word	Sub-words	Sub-words %	Stand-alone characters	Initial characters	Middle characters	Final characters
1	263,065	45.80%	263,065	0	0	0
2	159,995	27.90%	0	159,995	0	159,995
3	90,068	15.70%	0	90,068	90,068	90,068
4	43,124	7.50%	0	43,124	86,248	43,124
5	13,433	2.30%	0	13,433	40,299	13,433
6	3,633	0.63%	0	3,633	14,532	3,633
7	818	0.14%	0	818	4,090	818
8	247	0.04%	0	247	1,482	247
Total	574,383	100.00%	263,065	313,318	236,719	313,318
% characters			23.4%	27.8%	21%	27.8%

Table 2: Sub-words and Shapes Statistics

word رجال . It consists of 3 sub-words, containing 1, 2, and 1 character respectively.

3.2 Test Sample and Results

In order to give a fair idea about sub-words, a sample of Arabic text consisting of about 1.4MB was collected. It was randomly selected from old books, modern books, newspapers, and other available sources on the web. Statistics presented here about this sample text may give an idea about the structure of Arabic words in terms of sub-words and the four character shapes. Table 2 shows the analysis for this sample.

The sample consists of 262,647 words with 1,126,420 characters. This means that the average word length is 4.3 characters per word. The number of sub-words is 574,383. This means that on the average there is 2.2 sub-word per word. The number of sub-words consisting of one character is 263,065 which makes about 45.8% of the total number of sub-words (and 23.4% of the total number of characters). This means that, in the process of optical character recognition, slightly less than one half of the sub-words need no segmentation at all (whether printed or handwritten). The number of sub-words that consist of two characters is 159,995 which makes 27.9% of the total number of sub-words. This means that about 30% of the total number of sub-words need segmentation into two characters only. The table also shows that on the average the four different shapes of characters are almost equal with the middle form slightly less (23.4% stand-alone, 27.8% for each of the initial form and the final form, as the number should be equal, and 21% for the middle form).

3.3 Proposal for a New Procedure for Recognition of Arabic Characters

According to the above discussion, the approach suggested here is to separate the text into three groups:

1. Sub-words consisting of one character that is in the stand-alone form. This is to be recognised directly without any segmentation.
2. Sub-words consisting of two characters. The first one is in the initial form and the second one is in the final form. This needs segmentation in two parts only. If there is a pre-knowledge of the number of characters to be segmented in the sub-word, then the task becomes easier.
3. Sub-words consisting of more than two characters. The first one is in the initial form, the last one in the final form, and the rest are in the middle form.

Figure 1 shows the flow diagram for this procedure.

4. Approximate Stroke Sequence String Matching

Given two character images, there is no universally accepted definition for similarities or differences between them. If the two images are converted into a one-dimensional string, then the task will be easier to define. Distance between two histograms of angular measurements was the subject of some literatures (Cha 2000).

The stroke sequence is based on the 8-direction stroke convention shown in Figure 2.

Stroke sequence string matching (Cha 1999) is based on the individual distance $d_{i,j}$ between the i'th stroke in letter a_1 and the j'th stroke in letter a_2 where

$$d_{i,j} = |a_1(i) - a_2(j)| \quad \text{if } |a_1(i) - a_2(j)| \le 4$$

This value is to be modified so that the $d_{i,j}$ value is replaced by the value $8\text{-}d_{i,j}$ when it exceeds the value of 4. This is equivalent to taking the smallest angle between the two directions whether the rotation is clockwise or counter-clockwise. Hence $d_{i,j}$ gives the minimum number of necessary steps to turn from the direction given by $a_1(i)$ and $a_2(j)$.

Allowing a cost function of c=2 which allows a mini-

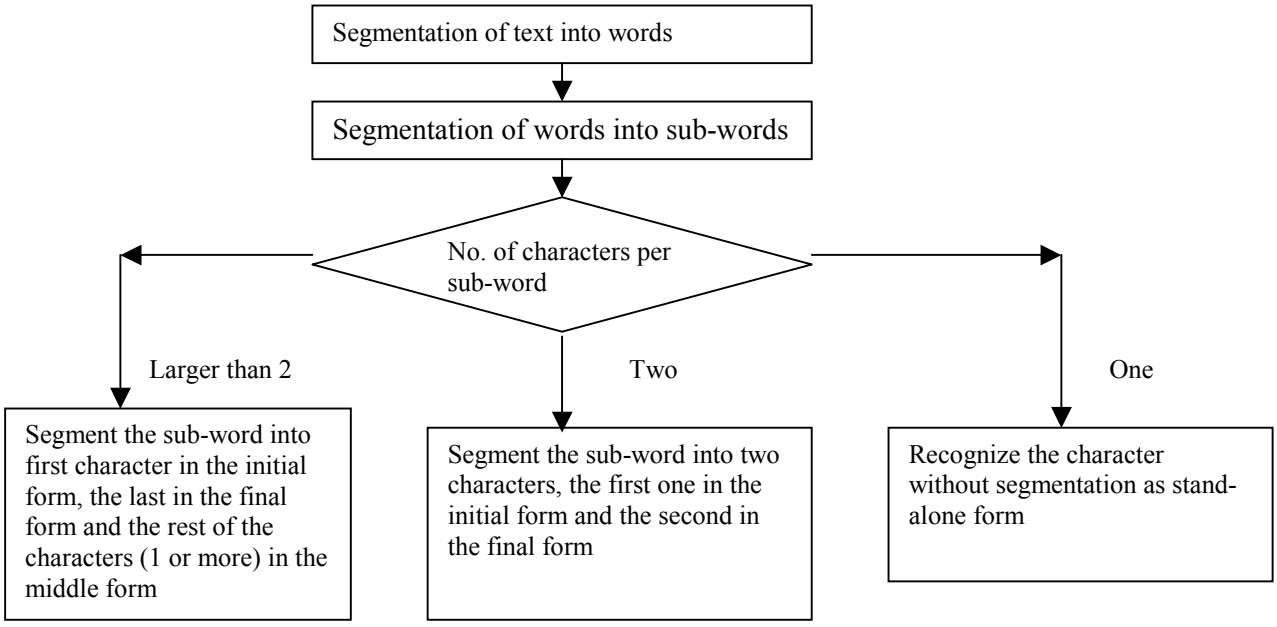


Figure 1: Proposed Procedure for Arabic Character Recognition

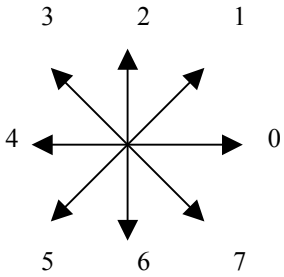


Figure 2: The 8-direction stroke convention

mum edit distance between two stroke sequence strings.

Figures 3a and 3b show the Arabic characters ح and ع taken as standard shapes. The stroke sequences corresponding to each of them is shown in the upper row of Tables 3 and 4, respectively. Figure 3c shows a supposed unknown character to be compared to each of them. However, this is a hand-written letter ع. The stroke sequence for this character is shown in the left-hand column of both Tables 3 and 4. They also show the computed distance tables between the unknown character and each of the two characters which are nearest in shape to it. Its distance to the letter ع is 8 while its distance to the letter ح is 10. This gives the result that the unknown character is ع and not ح .

The individual strokes on the top and left hand side of the table (starting row and column) are t_i and l_j simultaneously. The calculations involved in the shown Tables takes into consideration that

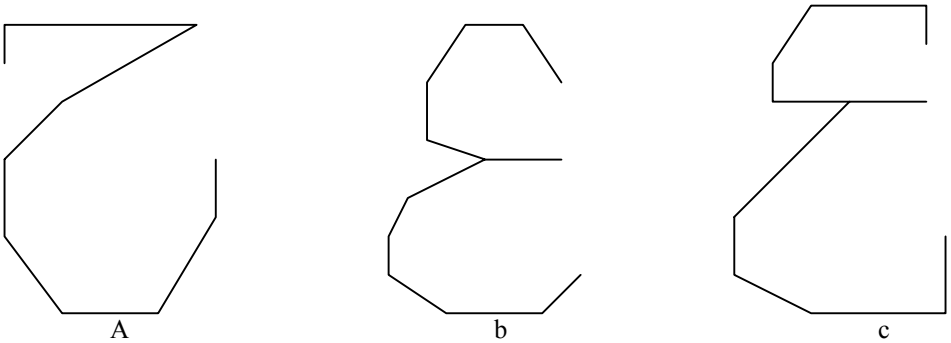
T(i,j) is the minimum value of T(i-1 , j-1) $+d_{i,j}$ and the two values T(i-1, j) +c when t_i is missing and T(I, j-1) +c when l_j is missing.

		2	0	0	5	5	6	7	0	1	2
	0	2	4	6	8	10	12	14	16	18	20
2	2	0	2	4	6	8	10	12	14	16	18
4	4	2	4	6	5	7	9	11	13	15	17
5	6	4	5	7	6	5	7	9	11	13	15
6	8	6	6	7	8	7	5	7	9	11	13
0	10	8	6	6	8	9	7	6	7	9	11
4	12	10	8	8	7	9	9	8	9	10	11
5	14	12	10	10	8	7	9	10	11	12	13
6	16	14	12	12	10	9	7	9	11	13	15
7	18	16	14	14	12	11	9	7	10	12	14
0	20	18	16	14	14	13	11	9	7	9	11
2	22	20	18	16	16	15	13	11	9	8	9
4	24	22	20	18	17	17	15	13	11	10	10

Table 3: String matching between the unknown character and character ح

		3	4	5	6	7	0	4	5	5	6	7	0	1
	0	2	4	6	8	10	12	14	16	18	20	22	24	26
2	2	1	3	5	7	9	11	13	15	17	19	21	23	25
4	4	3	1	3	5	8	9	11	13	15	17	19	21	23
5	6	5	3	1	3	5	7	9	11	13	15	17	19	24
6	8	7	7	3	1	3	5	7	9	11	13	15	7	19
0	10	9	9	5	3	2	3	5	7	9	11	13	15	17
4	12	11	9	7	5	4	5	3	5	7	9	11	13	15
5	14	13	11	9	7	6	7	5	3	5	7	9	11	13
6	16	15	13	11	9	8	8	7	5	4	5	7	9	11
7	18	17	15	13	11	9	9	9	7	6	5	5	7	9
0	20	19	17	15	13	12	11	11	9	8	7	6	4	7
2	22	21	19	17	15	14	13	13	11	10	9	8	7	6
4	24	23	21	19	17	16	15	13	13	12	11	10	9	8

Table 4:String matching between the unknown character and character ع



A b c

Figure 3: a: Character ح b: character ع d: unknown character to be matched with a and b

Figure 4: The letter ع hand written by 48 different persons

1. Results and Discussion

The above algorithm was applied to various Arabic stand-alone characters collected from 48 different persons. Figure 4 shows such character from the different handwriting of the 48 persons for the letter ع. These persons were asked to copy the same text without any restrictions on their writing style. The handwritten pages were scanned and then normalized to nearly the same overall sizes. The stand-alone characters were then copied and analysed by the above algorithms. Results showed recognition rate of about 80% for characters such as ا , ص , ه while the rate of recognition was much less than that for similiar characters such as ب & ف or ق & ن.

6. Conclusions

A statistical analysis on a sample of Arabic text, showed that the average Arabic word contains about 4.3 characters with an average of 2.2 sub-words. This shows that the basic block to be dealt with in Arabic OCR systems should be the sub-word rather than the word. The size of the sub-word varies from a single character up to 8 characters. The method of recognition of sub-words of different lengths ought to be different. For sub-words with a single character, the matching for recognition may be made only with characters of the stand-alone form. For sub-words with two characters only, a single shot segmentation has to be made dividing the sub-word into two characters. The first one is in the initial form and the second one in the final form. Sub-words of lengths longer than 2 characters need to be segmented into three characters or more. The first is of initial form, the last of final form and the rest

of middle form. Design of Arabic OCR system when taking these facts into account would be much simpler. However, the classification of sub-words according to the number of the characters they contain, still ought to be addressed.

As a first step, the recognition of the sub-words with a single character of a stand-alone form has been treated using the approximate stroke sequence string matching. Promising results are shown. Further refinement of the algorithm used need to be carried out for better rate of recognition.

7. Acknowledgement

The authors are grateful to the grant offered by The Higher Council for Science and Technology, Jordan and to the University of Jordan for offering the facilities to do the research. The authors also thank Mr. Samer Alayysh for helping in getting some of the results

8. References

Abd El-Gawad A.O., M.M. Salem, F.E.Z. Abou Shadi and H. Arafat, 1990. Automatic Recognition of Handwritten Arabic Characters. *10th International Conference on Pattern Recognition*.

Abdulla W.H., A.O.M. Saleh and A.H.Morad, 1988. A Preprocessing Algorithm for Hand-Written Character Recognition. *Pattern Recognition Letters.* 7 , 13-18.

Abo Samra G., K. Jambi, H. Al-Barhamtoshy, R. Amer and I. Al-Bidewi, 1997. A Comprehensive Algorithm for Segmenting Handwritten Arabic Script in Off-line Systems. *17th NCC, KFUPM.* 1-13.

Abuharba I.S.I., S.A. Mahmoud and R.J. Green, 1994. Recognition of Handwritten Cursive Arabic Characters. *HLL Transaction of Pattern Analysis and Machine Intelligence.*16:6. 664-672.

Ahmed P. and Y. Al-Ohali, 2000. Arabic Character Recognition: Progress and Challenges. *Journal of King Saud University.* 12, Comp. & Inf. Sci. 85-116

Ahmed P. and M. A. A. Khan, 1994. Computer Recognition of Arabic Script Based Text: the State of the Art. *Proc. of the 4th International Conference and Exhibition on Multi-lingual Computing (ICEMCO 94, Arabic and Roman Script)*, London, 22.1-22.15..

Al-Ohali Y.A.M, 1995. *Development and Evaluation Environment for Typewritten Arabic Character Recognition*. M.Sc. Thesis, King Saud University. Riyadh.

Al-Sharaidah, M. A., 2000. *Recognition of Handwritten Arabic Characters via Neural Networks* M.Sc. Thesis, Jordan University.

Al-Yousefi H. and S.S. Udpa, 1992. Recognition of Arabic Characters*," IEEE Transaction on Pattern Analysis and Machine Intelligence.* 14: 8. 853-857.

Altuwijiri M.M. and M.A. Bayoumi, 1995. A New Recognition System for Multi-Font Arabic Cursive words. *Proceedings of ICECS'95*, Amman, Jordan. 298-303.

Alshebeili S.A., A.A.F Nabawi and S.A. Mahmoud, 1997. Arabic Character Recognition Using 1-D Slices of the Character Spectrum. *Signal Processing, Elsevier.* 56. 59-75.

Alherbish J. and R.A. Ammar, 1997. Arabic Character Recognition in a Multi-processing Environment. *Proceedings of the 2nd IEEE Symposium on Computers and Communications.*

Al-Kadi Z. and S. Serhan, 1994. The Use of Learning Networks in Recogniton of Arabic Numeral. *Dirasat*, Jordan University. 22B:4. 933-949.

Al-Waily R.S.A., 1989. *A Study on Preprocessing and Syntactic Recognition of Hand-Written Arabic Characters*. M.Sc. Thesis, Basrah University, Iraq.

Amin A., 1997. *Handbook of Character Recognition and Document Image Analysis," Chapter 15* World Scientific Pub. Co. 397-420.

Bazzi I., R. Schwartz and J. Makhoul, 1999. An Ominfont Open-Vocabulary OCR System for English and Arabic. *IEEE Transaction on Pattern Analysis and Machine Intelligence.* 21: 6. 495-504.

Casey R.G. and E. Lecolinet, 1996. A Survey of Methods and Strategies in Character Segmentation. *IEEE Transaction on Pattern Analysis And Machine Intelligence.* 18:7.

Cha S.H., Y.C. Shin and S.N. Srihari, 1999. Approximate Stroke Sequence String Matching Algorithm for Character Recognition and Analysis. *5th International Conference on Document Analysis & Recognition* . Banglore, India

Cha S.H. and S.N. Srihari, 2000 "Distance between Histograms of Angular Measuremnts and its Application to Handwritten Character Similarity," *Proc. of International Conference on Pattern Recognition.*

El-Desouky A.I., M.M. Salem A.O. Abd El-Gwad and H.Arafat, 1992. A Handwritten Arabic Character Recognition Technique for Machine Reader. *International Journal of Mini and Microcomputers.* 14:2. 57-61.

El-Khaly F. and M.A. Sid-Ahmed, 1990. Machine Recognition of Optically Captured Machine Printed Arabic Text. *Pattern Recognition.* 23: 11, 1207-1214.

El-Sheikh T.S. and R.M. Guindi, 1988. Computer Recognition of Arabic Cursive Script. *Pattern Recogniton.* 21:4. 293-302.

Hussain F., B. Zalik and S. Kolmanic, 2000. Intelligent Digitisation of Arabic Characters. *Proceedings of the Conference on Information Visualization.* 337-342.

Hussain F. and J. Cowell, 2000. Character Recognition of Arabic and Latin Scripts. *Proceedings of the Conference on Information Visualization.* 51-56.

Haj-Hassan F., 1990. Arabic Character Recognition. *Computers and the Arabic Language*, Hemishere Publishing Co. 113-118.

Jambi K. M., 1991. Arabic Character Recognition: Many Approaches and One Decade. *The Arabian Journal for Science and Engineering.* 16: 4B. 499-508.

Jambi K.M., 1991. *Design and Implementation of a System for Recognizing Arabic Handwritten Words with Learning Ability*. Ph.D. Thesis , Illinois Institute of Technology.

Kavianifar M. and A. Amin, 1998. Preprocessing and Structural Feature Extraction for a Multi-Font Arabic/Persian OCR. *Proceedings of 5th International Conference of Document Analysis and Recognition.*

Khellah F., 1994. Recognition of Hexagonally Sampled Printed Arabic Characters. *The Arabian Journal for Science and Engineering.* 19:4A. 565-585.

Mahmoud S. A., I. AbuHaiba and R.J. Green, 1991 Skeltonization of Arabic Characters Using Clustering Based Skeltonization Algorithm. *Pattern Recognition.* 24: 5. 453-464.

Nouh A., A.N. Ula and A. Sharaf Eldin, 1988 Algorithm for Feature Extraction: A case Study for the Arabic Character Recognition. *Proceedings of 10th National Computer Conference*, Saudi Arabia. 653-666.

Plamondon, R., 2000. On-Line and Off-line Handwriting Recognition, A Comprehensive Survey. *IEEE Transaction on Pattern Analysis And Machine Intelligence,* 22:1.

Said F. N., R.A. Yacoub and C.Y. Suen, 1998. Recognition of English and Arabic Numerals Using a Dynamic Number of Hidden Neurons. *Proceedings of 5th International Conference of Document Analysis and Recognition.*

Salih A.O.M. and R.S. Al-Waily, 1994. Tree Representation of Hand-writen Arabic Characters. *Mu'tah Journal for Research and Studies.* 9:3. 125-140.

Salih A.O.M. and R.S. Al-Waily. 1996. Recognition of Hand-Writen Arabic Characters sing Tree Grammars *2nd Jordanian Conference in Electrical Engineering, Mutah University.* 402-413.

Sami S.E., R. Ramsis and A., 1990. Arabic Character Recognition System, A Statistical Approach for Recognizing Cursive Typewritten Text. *Pattern Recognition*. 23: 5, 485-495.

Talba M.F., S.A. Wahba and A. Salim, 1987. A Recognition Algorithm for Arabic Printed Characters," *Proceedings of the International Symposium of Applied Informatics,* Switzerland. 128-131

Challenges in Arabic NLP

Achraf Chalabi

Sakhr Software Co.
Sakhr Building, Nasr City, Free Zone,Cairo, Egypt
ac@sakhr.com

Abstract

Arabic language poses many challenges to Natural Language Processing . The present paper briefly describes some of those characteristics of Arabic language adding extra complexities and challenges to NLP compared to its English counterpart . Among others, the omission of diacritics in written text , the relatively free word-order, the presence of elliptic personal pronouns and the rich morphology, represent the main language features making Arabic NLP an even more sophisticated task.

1. Introduction

Arabic language is formally written using characters and diacritics. By analogy with Latin languages, characters in Arabic represent Latin consonants, while diacritics represent vowels. There are 3 basic diacritics in Arabic, namely, fatha (a), kasra (i) and damma (ou). In addition to these basic diacritics, there is a shadda , which is used in combination with any of them to produce three new "geminated" vowels. Moreover, each of the three basic vowels, could be doubled to produce a "nunated" vowel. While the basic three vowels and their corresponding geminated ones have direct impact on the linguistic interpretation of Arabic words, "nunated" vowels have a musical rather than linguistic role, and are involved more with the sound accompanying Arabic text.

2. Lack of Diacritics

Although Arabic should be written with full diacritics to avoid misinterpretations , or with mandatory diacritics to minimise ambiguities, most of written Arabic text, except for religious domain, lack diacritics fully. A "human" person reading Arabic text is performing contextual analysis in permanence, and sometimes even backtracking in order to reach the correct interpretation of each word and hence the right diacritics to be applied to the word prior to pronouncing it. Therefore, in reading Arabic text, the application of "reading" rules is the easiest part, and unlike Latin languages, must be preceded by analysis, disambiguation and interpretation.

Just to feel the task, read the following English sentence : " jst t fl th tsk, rd th fllwng nglsh sntnc"
which is the non-vowelized version of the sentence before, and in which
"fl", could represent (file/foil/fool/feel/fly/flee/…..)
"rd" could represent (rod/road/read/red/raid/ride…)
Therefore, computational processing of Arabic language is considered one order of magnitude more complex than its Latin counterpart, assuming only the absence of diacritics problem, which increases significantly the amount of morphological and lexical ambiguities , hence resulting in a combinatorial chain of syntactic ambiguities.

3. Rich Morphology

Arabic language has a very rich morphology. Words in Arabic are constructed out of prefixes, stem, infixes and suffixes. The stem itself is composed of two basic elements : the root and the morphological pattern (or diacritical pattern). The root could be a "tri root", consisting of three characters, or a "quad root" composed of four characters. The application of a morphological pattern to a root generates a stem, which is the basic form of an Arabic word token. Arabic language consists of about 6 000 roots and 700 morphological patterns. Not all patterns could be applied to a given root. The actual valid root-pattern combinations in Arabic generate around 150,000 stems. Two thirds such stems are considered classical Arabic and only the remaining 50 000 stems are those actually used in daily life.

Arabic stems allow attachment of a multitude of prefixes and suffixes, hence constructing final form word tokens. Just to name some, prefixes in Arabic could be:

- Prepositions ل → to / for
  ب → with
  ك → as
- Conjunctions و → and
  ف → then
- Adverbs ف → so
- Auxiliary verb س → will
- Interrogative أ → did-have

On the other side of the word, suffixes have a much wider range of values, such as:

- Case ending suffixes (nominative/ accusative/ genitive)
- Number suffixes (dual/ proper plural)
- Gender suffixes (masculine/ feminine)
- Personal pronouns
- Object pronouns
- Genetive pronouns
- Possessive pronouns

By catenating prefixes and suffixes to a stem, a whole English sentence could be represented in one single Arabic word.
Ex.

Arabic : فسأقابلكما

English : Then I will meet both of you

The rich morphological nature of Arabic words represents an additional major obstacle for computational processing, especially on the morphological level:

Ex.

فضل → (فَضْلٌ)
(فضَّلَ)
(فَ + ضَلٌّ)
.
.
.

In the above example, among the seven possible morphological interpretations for the input non-diacritized word (فضـــل), two will consider the first character an adverbial prefix, assuming a root totally different from the other five alternatives.

While the highly inflectional nature of Arabic poses many complexities in morphological analysis and disambiguation, it does however, in combination with the syntactic constraints of verb-subject and noun-adjective agreement, provide on the syntactic level, many useful clues serving structural analysis and disambiguation.

4. Free Word-Order

Arabic has a relatively free word-order syntax. Tokens constituting an Arabic sentence could be freely moved, without affecting the syntactic validity or the semantic interpretation of the sentence.

Ex.

Ate the man the apple	أكل الرجل التفاحة
Ate the apple the man	أكل التفاحة الرجل
The man ate the apple	الرجل أكل التفاحة

One major problem arising from such flexible word-order, is the ability to swap subjects and objects, which would enable interpretations such as (the apple ate the man) that are syntactically valid but semantically wrong. Therefore, processing Arabic sentences requires syntactic analysis to permanently work in a "hand-shake" mode with semantic analysis. Any attempt to attach a noun phrase to a verb as a subject, object, complement or even adverb, has to consult the semantic analyser prior to attachment.

Many years of intensive work were needed in order to build the selection restrictions data used in ***Sakhr's*** proprietary Arabic semantic analyser. In recent years, the increasing availability of Arabic text in electronic form enabled the construction of very large Arabic text corpora, which in turn, opened new horizons for Arabic Natural Language Processing by combining statistical methods with rule-based techniques.

Arabic free word-order requires also a much more complex formal grammar, compared with its Latin counterpart, in order to reach a comprehensive coverage of valid Arabic structures. A PS-grammar has been written in ***Sakhr*** to drive its single-stack multi-level parser which runs simultaneously with the semantic analyser, and form together the core engine for Arabic sentence processing.

5. Elliptic Personal Pronoun

One main characteristic of Arabic Language is the presence of an elliptical personal pronoun, which is always single but may be masculine or feminine. In combination with the omitted diacritics, elliptical pronouns create one of the most complex problems to Arabic computational processing. Consider the word "أكل" , without diacritics, this word could be one of eight possible words. Two such possibilities are

(i) أَكَلَ (eat),

(ii) أَكَّلَ(feed).

Although both word forms have a common root (ء ك ل), making them belonging to the same semantic cluster, different morphological patterns have resulted in totally different meanings and lexico-syntactic features. One such feature, relevant to our context, is "transitivity". While (أكَـــلَ) could be intransitive or transitive; (أكَّـــلَ) can be transitive or ditransitive. Based on the above, the sentence: أكـــل الولـــد تفاحـــة Could have the following two interpretations :

(i) the boy ate an apple أَكَلَ الوَلَدُ تفاحةً

(ii) (He) fed the boy an apple أَكَّلَ الولدَ تفاحة

Where (i) has assumed the transitive alternative for the verb (أكـــل) and (ii) the ditransitive one. Resolving this issue necessitates robust and intelligent syntactic analyser, supported by a mandatory pronominal reference resolver.

6. Conclusion

Challenges imposed by Arabic language nature push NLP to the extreme, motivating creativity and exhaustive exploitation of every single bit of already available techniques and linguistic resources. The recent emergence of Arabic electronic texts (newspapers, magazines, books, Web sites,…etc) is paving the road to the implementation and integration of new statistical-based modules within the originally rule-based system resulting in a more powerful and accurate hybrid system.

Sakhr has been involved in Arabic NLP since 1985 and has released the first Arabic morphological analyser two years later. It has since, pursued and intensified its Research and Development efforts in this line, building a solid infrastructure for Arabic NLP , including the Arabic Lexicon , Grammar, Parser ,Selection Restrictions, Corpus, Automatic diacritizer …etc, and is now proceeding in the development of its bidirectional Arabic<>English Machine Translation engine powering Sakhr's free on-line translation Web site at http://www.tarjim.com.

OrienTel – Arabic speech resources for the IT market

Rainer Siemund[1], Barbara Heuft[1], Khalid Choukri[2], Ossama Emam[3], Emmanuel Maragoudakis[4], Herbert Tropf[5], Oren Gedge[6], Sherrie Shammass[6], Asuncion Moreno[7], Albino Nogueiras Rodriguez[7], Imed Zitouni[8], Dorota Iskra[9]

[1] Philips Speech Processing, [2] ELDA, [3] IBM, [4] University of Patras, [5] Siemens, [6] Natural Speech Communication, [7] Universitat Politècnica de Catalunya, [8] Lucent Technologies, [9] SPEX

c/o Rainer Siemund, Philips Speech Processing, Kackertstr. 10, D-52072 Aachen, Germany
rainer.siemund@philips.com
http://www.orientel.org

Abstract

A survey of the language resources market clearly shows that the Arabic language is still a stepchild of international R&D efforts in the field of speech recognition. *OrienTel* for the first time makes an effort to create speech data on a large scale. It does so by profiting from the experience of previous *SpeechDat* projects and from the European Commission's policy to embrace non-EU Mediterranean and surrounding countries. The participants of *OrienTel* will collect Standard and Colloquial varieties of Arabic in Saudi Arabia, the UAE, Egypt, Israel + Palestine, Tunisia and Morocco, supplemented by other languages of the region. Help in creating an Arabic network of speech experts is appreciated.

1. Introduction and goal of the paper

Like all other members of the *SpeechDat* family of data collections, *OrienTel* is driven by an international industrial and academic consortium.[1] This time the co-ordinator is Philips Speech Processing, the other participants being ELDA, IBM, Knowledge, the University of Patras, Siemens, NSC, Universitat Politècnica de Catalunya, and Lucent Technologies. *OrienTel* resembles previous *SpeechDat*-like undertakings[2] insofar as the recordings are supposed to serve the broadest possible application areas, ranging from simple command and control services to unified messaging, information retrieval, customer care, banking, WAP and service portals. It is different from previous projects, however, as it takes *SpeechDat* to a variety of non-European languages such as Arabic and Hebrew, which require far-reaching adaptations to database design and annotation standards. The aim of the present paper is to introduce the broad setup of *OrienTel*, give an account of the present status of database design, and to call for a joint effort in producing language resources for the Mediterranean and the Middle East. In addition to the rather general description of the project presented in the LREC2002 Proceedings, the present paper provides some Arabic-specific problems.

2. Why *OrienTel* now?

The *OrienTel* region is a region of extremes. While Israel ranks as one of the World Bank's 26 so-called 'developed' economies, its next door neighbour, the Palestine Authorities, are having to cope with a GDP per capita figure of just US$ 1,634.[3] The contrasts are even more extreme among the Gulf states. The United Arab Emirates possess the highest GDP per capita of the region - which at US$ 16,800 is on one level with most Western European countries - yet just a few hundred kilometres away Yemen reports an average GDP figure of only US$ 304. From a commercial point of view, therefore, not all countries in the *OrienTel* region are currently of equal interest to the present consortium. In anticipation of high growth rates for future mobile communication services, however, the project boldly aims at covering the whole area between Morocco in the West and Kuwait in the East, from Turkey in the North to Yemen in the South of the *OrienTel* region. Through the development of dialect adaptation techniques it will be possible in the near future, we hope, to adapt acoustic models of one language variety of Arabic to a related one. For the time being, a preselection process based on linguistic and commercial judgements as well as on considerations of sheer manpower has picked 9 out of potential 19 countries in which *OrienTel* will become active.

Despite the diversity between individual markets certain general trends are apparent. Mirroring what has happened in the rest of the developed world, cellular telephony has grown rapidly, particularly in markets where the fixed line infrastructure is inadequate. A case in point is Egypt. With 62 million inhabitants Egypt is the second most populous country in the *OrienTel* region: its fixed line teledensity of 10.97% places it at place 13 in the regional league table of lines per 100 inhabitants, yet it comes third in terms of number of mobile subscribers. While the Egyptian government has attempted to improve the availability of fixed line telephony by setting ambitious targets for state-owned Telecom Egypt, the private sector has been allowed a relatively free rein in the mobile sector. The result has been an explosion in the

[1] Thanks go to the European Commission, who are funding *OrienTel* as an R&D project under the 5th Framework Programme (Contract IST-2000-28373).

[2] Infos on *SpeechDat* and related projects can be gathered from http://www.speechdat.org . Publications focusing on specific members of the *SpeechDat* family are, for example, Höge/Tropf 1996 (*SpeechDat M*), Höge et al. 1999 (*SpeechDat II*), Pollak et al. 2000 (*SpeechDat East*), Moreno et al. 2000a (*SpeechDat Car*), and Moreno et al. 2000b (*SALA*).

[3] All figures of the present section were taken from CIT Publications (2000).

number of cellular subscribers. During 1999 the number of subscriptions rocketed from 187,000 to 890,000. According to Egypt's two mobile operators, France Télécom-backed MobilNil and Vodafone AirTouch's Misrfone, the market doubled in size again during 2000 to 1.8 million. Turkey is in a similar situation. With a fixed line network of 17.4 million lines, Turkey's teledensity of 26.6% places it sixth in the regional ranking table, yet its mobile sector has experienced nothing short of phenomenal growth: at the end of 1996 mobile penetration stood at just over 1.2%, but three years later it had increased ten-fold to 12%, and rose to just below 20% by the end of 2000.[4] Company-internal considerations of some *OrienTel* partners bear further evidence of the current interest in speech applications gradually extending from Europe towards some of the countries covered in the project. A survey of the needs for future language development undertaken by ELRA points into the same direction: particularly Arabic and Turkish are currently on the wish list of many companies active in the field of language and speech. Largely due to such infrastructural and commercial considerations, the *OrienTel* consortium chose nine out of a potential set of 19 countries between Morocco in the West and the Gulf States in the East. The countries treated in *OrienTel* so far are depicted in Table 1:

Country	Partner
UAE	Philips
Saudi Arabia	Lucent
Israel/Palestine	NSC
Egypt	IBM
Tunisia	UPC/ELDA
Morocco	ELDA/UPC
Turkey	Siemens
Cyprus	Knowledge/Patras Univ.

Table 1: *OrienTel* countries and partners

More countries may follow in case new partners decide to join the project.

3. Linguistic settings

From a linguistic point of view, too, the *OrienTel* region is far more diverse than any region covered in previous projects of a similar scope such as the various members of the *SpeechDat*-family or *SpeeCon* (Siemund et al. 2000, cf. also http://www.speecon.com). In order to treat the linguistic peculiarities of the area adequately, *OrienTel* follows a different strategy than previous *SpeechDat* projects. As Table 1 shows, each partner in the consortium is not responsible for a single language but for a whole country. The difference is an important one, since, as will be outlined below, in most *OrienTel* countries everyday-life is governed by more than a single language. One of the first project tasks was therefore to determine the various languages spoken in the *OrienTel* region, taking into account both linguistic and commercial criteria. From the consortium's point of view Arabic, Turkish, Hebrew and Cypriote Greek turned out to be of most immediate concern with Farsi being on the wish list for future stages of the project. Furthermore, English and French turned out to be of commercial interest as the dominant business languages in some *OrienTel* countries and because non-native varieties of European languages constitute a hitherto grossly neglected domain in linguistic research. This is also the reason why *OrienTel*'s language portfolio is complemented by German as spoken by Turks in Germany, who represent the largest linguistic minority of the country.

The most complex linguistic picture of the *OrienTel* region, however, is no doubt presented by Arabic and its variants. Arabic of the *OrienTel* area can be subdivided into four broad dialect regions, as outlined in Table 2:

Dialect region	Countries
Mahgreb Arabic	Morocco, Algeria, Tunisia, parts of Libya
Egyptian Arabic	Egypt, parts of Libya
Levantine Arabic	Syria, Lebanon, Israel + Palestine Authorities, Jordan
Gulf Arabic	Kuwait, Qatar, Bahrain, UAE, Saudi Arabia, Oman

Table 2: Dialect regions of Arabic

In order to represent all dialect regions adequately, all areas are attended to by at least one partner. In each country, the variety of languages spoken is rather large. In Morocco, for example, the official language is Modern Standard Arabic, the rather formal language of religion, the media and of public institutions. In everyday interaction though, people either tend to speak a local colloquial variant of Arabic that is only remotely related to the Standard (not to mention the various non-Arabic languages such as Berber) or, when it comes to commercial interaction, French as the language inherited from Morocco's colonial past. All three (or even more) languages have their place in everyday life and user-friendly applications have to take into account each country's linguistic diversity and its users' preferences. The databases produced in *OrienTel* are depicted in Table 3:

Country	1st language	2nd language	3rd language
UAE	Mod. Std. Arabic	Modern Coll. Arabic	English
Saudi Arabia	Mod. Std. Arabic	Modern Coll. Arabic	English
Israel/Pal. Auth.	Mod. Std. Arabic	Mod. Coll. Arabic	Hebrew
Egypt	Modern Std. Arabic	Modern Coll. Arabic	English
Tunisia	Mod. Std. Arabic	Modern Coll. Arabic	French
Morocco	Mod. Std. Arabic	Modern Coll. Arabic	French
Turkey	Turkish	-	German
Cyprus	Cypriote Greek	-	English

Table 3: *OrienTel* languages

[4] The market analysis is based on *Telecommunications Markets in the Middle East*. Exeter: CIT Publications, 2000.

As can be gathered from Table 3, the *OrienTel* consortium will produce a set of 22 databases in 8 countries, all of which will be made publicly available after the end of the project and a commercially reasonable quarantine period.

4. Linguistic research and dialect adaptation

The rather complicated linguistic situation in the *OrienTel* countries calls for innovative approaches to speech recognition techniques. Thorough research will be conducted into multilingual acoustic modelling and the development of multilingual lexicons, including descriptions of phonetic inventories. An important goal will also be the development of phonetic and orthographic transcription strategies. By default written Arabic and Hebrew orthography depict consonants only. Even though it is possible to render vowels by supra- and supersegmental markers, fluent reading of such "annotated" words is awkward even for native speakers. Strategies will therefore be developed to prompt speakers reliably even if the meaning of, for example, a single command word cannot be gathered from the context of whole sentences. The problem of vowels is of particular importance especially since it is largely the vowels on which current Hidden Markov Modelling heavily relies (cf. Rabiner/Juang 1993). Once parts of the various databases become available, it will therefore be one of the main research tasks to assess the linguistic features of dialect clusters and develop techniques of dialect adaptation across the Arabic-speaking world.

5. Foreign accent adaptation

Apart from the databases representing Standard varieties and local dialects, a separate set of data will be produced for foreign accent adaptation. Due to the *OrienTel* countries' colonial, protectorate or migration history, the most prominent foreign languages in the region are French, English and, for different reasons, German. On the one hand, collecting data of this kind will ensure true multilinguality of applications in the *OrienTel* countries. On the other hand, French, English and German services already under operation in the EU can be adapted to foreign accent variation.

6. Demonstrator development

In order to show that the multilinguality approach taken in *OrienTel* is feasible, the project will produce two demonstrator applications. The exact kind of services will be specified at a later stage of the project. Considerations will, however, take into account the convergence of internet, WAP and voice for service portals, unified messaging, customer care applications, directory assistance and banking. The two demonstrators will reflect two different types of services and will account for two different linguistic regions.

7. Dissemination of information and results

In order to keep the speech recognition community informed about the *OrienTel* efforts, the project will contribute to scientific discussions concerning the languages of the *OrienTel* region at conferences, in publications and through relevant mailing lists. It will furthermore continuously update the project's website with information on *OrienTel* activities and publish the results (cf. section 9 below). The 22 databases will be made publicly available through the European Language Resources Association (ELRA) in due course after the project has ended.

8. Database specification

Due to the linguistic heterogeneity of the region, questions of database specification such as corpus composition, orthographic and phonetic transcription strategies constitute a crucial part of the project. Particularly Arabic and Hebrew pose interesting problems for speech recognition that were never tackled in projects of the *OrienTel* scale before. Cases in point are the rendering of vowels, the right-to-left writing system and the transcription of oral or colloquial speaking styles. While at the time of writing the present paper quite a few design details are still under discussion (the design phase is due for completion before the LREC2002 conference starts), some of the cornerstones can already be reported at the present stage.

8.1. Recording scenarios and platforms

All *OrienTel* databases will be recorded from fixed and mobile networks via ISDN lines and multiple channels, i.e. either through a Basic Rate Interface or a Primary Rate Interface (cf. Senia 1998). A dialogue will be implemented by the application driving the recordings. The dialogues will be designed to make the caller speak and act comfortably.

8.2. Corpus and vocabulary

Data collections will rely on three separate sets of prompt sheets, namely one each for

- the 'foreign' languages in Arabic-speaking countries, i.e. English and French, including Turkish, Greek, Hebrew and German
- Modern Standard Arabic
- Modern Colloquial Arabic

While the specifications for English, French, Greek and German are largely based on previous *SpeechDat* projects and *SpeeCon*, the design for Arabic and Turkish presents a novelty. All three sets of prompt sheets, however, contain the following items, though in varying quantities with at least 47 items per sheet:

- isolated digits
- digit and number strings
- natural numbers
- currency amounts
- yes/no questions
- dates
- times
- application keywords and phrases
- word spotting phrase using embedded application words

- directory assistances names (proper names, place names, company names)
- spellings
- phonetically rich words and sentences
- spontaneous utterances

8.3. Transcription and annotation

The *OrienTel* transcription and annotation conventions are largely based on conventions used by the Linguistic Data Consortium and ARPA in producing the ATIS CD-ROMs[5], and the simplifications made for the *SpeechDat*-predecessors of this project, and *SpeeCon*. The goal of the specification document that should be finished by the time of the present workshop is to define a coarse transcription that can be performed quickly, but covers adequately the acoustic events most important for the training and testing of automatic speech recognisers. The transcription is orthographic (cf. the lexicon section below for phonetic renderings) and includes a few markers representing audible acoustic events (speech and non-speech) present in the corresponding waveform files. The phoneme symbol set aims at the localisation of the main acoustic events according to a coarse categorisation rather than a full description of all possible sounds that may appear during a recording. Extra marks contained in the transcription aid in interpreting the text form of the utterance; markers for non-speech acoustic events and distortions have been chosen such that they can be automatically removed or modified to yield the base transcription. The overall aim is to keep as much speech in the corpus as possible and to avoid the need for deleting recordings from the corpus due to some extra noises, disfluencies, etc. All items for all languages covered will be transcribed in standard orthography and will be Romanized in the label files. A Sampa transliteration will be generated and discussed with the Department of Phonetics and Linguistics at UCL (cf. http://www.phon.ucl.ac.uk/home/sampa/home.htm) if need be. Administrative information on speech files and their properties will be stored in SAM files (cf. http://www.icp.grenet.fr/Relator/standsam.html).

8.3.1. Strategies for recording colloquial Arabic

Dialectal Arabic is exclusively a spoken language and can very rarely be found in the written form. This fact imposes constraints on the recording procedure. There are a number of possibilities with regard to collecting colloquial Arabic speech:

1. recording spontaneous speech only;
2. presenting audio prompts to the speaker who then only needs to repeat what has been prompted;
3. presenting written prompt sheets to the speaker that he/she needs to read. The prompts can be
    a. written in vowelized Arabic script or
    b. transcribed using Latin alphabet.

The first option is likely to provide the most natural results. However, transcription of such spontaneous material would be very difficult as well as time- and money-consuming. That is why it has been decided to include a limited number of spontaneous items in the recordings. An example of such a spontaneous item is asking the speaker to answer a question on, e.g., the sightseeing sites of his/her town. Furthermore, spontaneous speech will be recorded as response to questions concerning dates, natural numbers and proper names. An example is enquiring the speaker's date and place of birth. Such questions are expected to produce short and simple utterances.

Typically, a *SpeechDat*-like database contains a number of items whose content has been defined in advance, such as phonetically rich sentences and application words. For this type of recordings option 2 or 3 need to be taken into account (the audio prompts or prompt sheets). In order to determine the best approach a number of basic experiments have been carried out with Moroccan speakers. These have enabled us to dismiss option 3b (transcription with Latin characters). During the experiments using this prompting approach none of the speakers was able to pronounce words naturally. On the other hand, the best results were obtained using audio prompts. However, this option has been dismissed too since it poses too many practical problems with such a high number of speakers and items which need to be recorded. Colloquial Arabic is difficult to read for most of the speakers since they are not used to reading it. Nevertheless, the majority of the speakers manage to come up with a correct pronunciation after having analysed the prompted text for a moment. Because of that, however, it is important to grant the speakers some extra time to become acquainted with the script before the recording starts.

8.3.2. Strategies for recording standard Arabic

Conventionally, the orthographic representation of standard Arabic relies on consonants. Although it is possible to represent vowels using diacritics (located super- and suprasegmentally), fluent reading of such scripts remains a challenge; most of the speakers are not used to reading texts containing vowel diacritics. In complete sentences it is possible for the speaker to deduce the vowels from the context. However, for isolated words it is necessary to mark the vowels in order to disambiguate between the different options. This situation is similar to that of colloquial Arabic in which the speakers need extra preparation time to be able to read vowelised scripts without hesitation and pauses.

8.4. Specification of speakers

The number of speakers to be recorded is 2000 per country. This number is distributed between the set of databases to be collected. Table 4 on the following page shows the minimum number of speakers per country and recorded language. A maximum overlap of 15% in the total number of speakers per country between the different databases is allowed.

[5] Cf. http://www.ldc.upenn.edu/, http://www.arpa.gov/, and http://www.atis.org , respectively.

Country	Colloquial	Standard	Business
Morocco	1000	500	500
Tunisia	1000	500	500
Egypt	1000	500	500
UAE	1000	500	500
Saudi Arabia	1000	500	500
Turkey	-	1700	300
Israel	500	500	1000[6]
Cyprus	1000	-	1000[7]

Table 4: Number of speakers per database

8.4.1. Gender

The distribution of male and female speakers should be 50% each per database, with an allowed deviation of 5% for the whole database per language. There is no gender restriction for "Age" and "Dialect". For "Environment", the gender distribution must be 30-70% for each sub-category.

8.4.2. Age

Table 5 presents the distribution of speaker age:

Age	16-30	31-45	46-60
Proportion	≥ 30%	≥ 20%	≥ 10%
Requirement	Mandatory	Mandatory	Mandatory

Table 5: Distribution of speaker age

Naïve speakers should be recorded rather than experienced or trained speakers to guarantee more natural speaking styles, voices and dialects.

8.4.3. Dialect

Many (though not all) of the languages spoken in the *OrienTel* regions are not the speaker's actual mother tongue. In such cases, we consider a person who spent most of his/her childhood, or who grew up in the concerned region, as having no foreign accent. Language-specific cases should be documented in the LSPs.

The specific number of dialects relevant to each country should be discussed in the LSP documentation. The distribution of speakers over dialect regions refers only to the colloquial varieties of the language. Speech should be collected from a minimum of three different dialect regions (if possible), with at least 20 speakers recorded for each defined dialect.

The speaker's dialectal region is determined by asking the question *"in which district did you grow up" or "where did you spend most of your childhood"*, not the question *"where do you live"*. The allocation of city/district names to the corresponding dialect region can be determined according to the information provided by each partner in the LSP documentation.

8.4.4. Distribution of environments

The speaker distribution for the mobile network should be between 65 to 75% of the total number of speakers in the database; e.g., if there are 1000 speakers in the database, between 650 and 750 of them should be recorded through the mobile network. At least 30% of each gender must be recorded in each environment.

Both the fixed and mobile networks are further divided into specific environments. Speaker distribution over each environment is shown in Table 6:

	Environment	Speaker distribution
Fixed network 30% ± 5%	Home/office	≥ 75%
	Public place/booth	
Mobile network 70% ± 5%	Home/office	≥ 20%
	Public place/street	≥ 20%
	Vehicle	≥ 15%
	Hands-free car kit (optional)	≥ 5%

Table 6: Distribution of recording environments

8.5. Specification of the lexicon

The lexicon is an alphabetically ordered table of distinct lexical items that occur in the corpus with the corresponding pronunciation information. Each distinct word should have a separate entry, which will be laid down in the order orthography ⇨ frequency ⇨ transliteration (for Arabic and Hebrew) ⇨ phonetic transcription ⇨ variants (optional).

The lexicon is derived from the annotated database and is set up as follows:

- Standard Language: Arabic & Hebrew script, both vocalized and not vocalized.
- European languages: Latin script
- Colloquial Language: Region specific Arabic script (same as in orthographic annotations)
- Acronyms such as *IBM* should appear as complete words in the lexicon, i.e. as letters with no spaces in between. The reason is that there are often different ways of pronouncing them (spelled and expanded).

The phonetic alphabet used will be SAMPA, and is thus case-sensitive. While a Hebrew SAMPA alphabet is currently under negotiation for standardization as part of the *SpeeCon* project, *OrienTel* will make an effort to further standardize Arabic and Turkish SAMPA alphabets. Sampa symbols for each language are defined in the language-specific documents accompanying each database and are considered as a standard set of phonemes for that language.

9. Disclaimer and Contact

Since the specifications outlined in this document are still being discussed at present and are thus still subject to revision, the latest state of the *OrienTel* art can always be gathered from the continuously updated *OrienTel* website at http://www.orientel.org. The co-ordinators of the project can be contacted either via the internet pages or through rainer.siemund@philips.com.

[6] Hebrew.
[7] Greek.

10. References

CIT Publications (2000). *Telecommunications markets in the Middle East*. Exeter: CIT Publications.

Höge, H., C. Draxler, H. van den Heuvel, F.T. Johansen, E. Sanders, H. Tropf (1999). Speechdat multilingual speech databases for teleservices: Across the finish line. In *Proceedings of EUROSPEECH '99*, vol. 6 (pp. 2699–2702). Budapest: ESCA.

Höge, H., H. Tropf (1996). SpeechDat (M) Final Report (D06/D07). Available from http://www.speechdat.org.

Moreno, A., B. Lindberg, C. Draxler, G. Richard, K. Choukri, S. Euler, J. Allen (2000a). SpeechDat-Car. A large speech database for automotive environments. In *Second International Conference on Language Resources and Evaluation. Proceedings vol. II* (pp. 895--900). Athens: ELRA.

Moreno, A. R. Comeyne, K. Haslam, H. v. d. Heuvel, H. Höge, S. Horbach, G. Micca (2000b). SALA: SpeechDat across Latin America. Results of the first phase. In *Second International Conference on Language Resources and Evaluation. Proceedings vol. II* (pp. 877–882). Athens: ELRA.

Pollak, P., J. Cernocky, J. Boudy, K. Choukri, H. v.d. Heuvel, K. Vicsi, A. Virag, R. Siemund, W. Majewski, J. Sadowksi, P. Staroniewicz, H. Tropf, J. Kochanina, A. Ostrouchov, M. Rusko, M. Trnka (2000). SpeechDat(E) - Eastern European Telephone Speech Databases. In *Proceedings LREC'2000 Satellite workshop XLDB - Very large Telephone Speech Databases, 29 May 2000* (pp. 20–25). Athens: ELRA.

Rabiner L.R., and B. Juang (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. In *IEEE Transactions on Acoustics, Speech and Signal Processing* 77(2), 257–286.

Senia, F., I. Chatzi (1997), Installation of the recording device and documentation. *Deliverable SD2.1 of the SpeechDat II project LE2-4001-SD2.1*. Available from http://www.speechdat.org/speechdat/deliverables/public/SD21V22.doc .

Siemund R., H. Höge, S. Kunzmann, and Marasek K., (2000). *SpeeCon* - speech data for consumer devices. *Second International Conference on Language Resources and Evaluation. Proceedings vol.* II (pp. 883—886). Athens: ELRA.

The Semi-automatic Tagging of Arabic Corpora

Mark Van Mol
professor Catholic University Leuven
ILT
Dekenstraat 6
B 3000 Leuven
Mark.VanMol@ilt.kuleuven.ac.be

Abstract

At the Institute of Living Languages of the Catholic University of Leuven we developed a system to encode Arabic corpora which enables us to identify strings of characters and to analyse them and disambiguate words. At the institute we developed two kinds of databases, one word-oriented and one sentence-oriented. The word-oriented database contains until now 26,000 Arabic lemmata with all the grammatical information. The second database contains a text corpus of approximatively 4,000,000 tagged Arabic words of which 1,200,000 from spoken Arabic language resources. Both databases will be used in the future in order to develop a semi-automatic tagging of raw Arabic corpora. In order to make Arabic electronic corpora useful for a large variety of purposes a pre treatment seems to be necessary. This treatment comprises three main phases. In the first place the uniformisation of Arabic corpora. The second phase involves the identification of strings of characters and the third phase involves the disambiguation of words on the basis of information coming from both sources. Once the corpus tagged this way, it containes enough detailed information to make scientific searches and analyses.

1. Introduction

In recent years more and more attention has been paid to gather Arabic corpora. Besides the ELRA initiative, numerous groups take the initiative in compiling all kinds of corpora. Many of these initiatives however, suffice with the compilation of raw corpus materials. Unlike other languages, however, the Arabic written language is ambiguous in many respects. The ambiguity of Arabic lies in the first place in the fact that the language is not vocalised. Often it is stated that languages with a rich morphology open much more facilities for tagging. The first problem in Arabic, however is, that written texts are not vocalized except in schoolbooks from primary schools and Coranic texts. All other material remains unvocalised which, of course, raises the level of ambiguity.

2. Levels of ambiguity in Arabic

The ambiguity of Arabic lies on different levels. The first level is the core word itself. Many core words can contain in it different grammatical categories. Below we give a few examples of possible combinations of grammatical categories of unvocalised words. Even if we limit ourselves to the main part of speech categories we find many ambiguous words.

2.1. First level: Core word

2.1.1. Noun - adjective.

Many Arabic word patterns can stand both for a noun and an adjective. Without being exhaustive we mention first the pattern which is most often an adjective, but which can also be a noun. For example, the word (*small*), which, of course, more exceptionally, can be used as a noun meaning the small one. The predictability of the grammatical category of those patterns is not always self-evident. One might suppose that the word pattern most often stands for an adjective, but this is not always the case. Take for instance the word of which it is quite clear, at first sight, that it is a noun, but which in Modern Standard Arabic (especially in North Africa) is often used as an adjective meaning principal.

The same goes for almost all the words ending in a so-called nisba. Indeed, most of those words are, as far as their Arabic pattern is concerned, unpredictably a noun or an adjective. Take, e.g., the word which means *politician* (noun) as well as *political* (adjective). Of course, a completely trained tagged corpus might shed some light on the chance rate of those grammatical categories, but the pattern itself does not say anything on the grammatical category of the word, except that it excludes to some extent the labeling of other categories, such as a verb or a particle. But even this remains in many cases problematic, because the *nisba* characteristic is in many cases not sufficient to exclude other grammatical categories, such as the verb or the particle, especially when unvocalised words are involved. Indeed, forms such as verbs ending with ya' , for instance, the verb (*to stay*) or the particle (*any*) could on the basis of the ending characteristic wrongly be interpreted as being an adjective or a noun.

Other ambiguous word patterns that cover both nouns and adjectives are the patterns · (e.g. noun: *drunk* - adjective: *intoxicated*), (e.g.: noun: *mason* – adjective *constructive*), (e.g. noun: *idler* - adjective: *lazy*)

2.1.2. Participles

Another word pattern, which covers both nouns and adjectives, is the pattern of both active and passive participles , and dervatives. These cases are sometimes even more complicated because they can also be classified from time to time as a preposition (for example: within) but even sometimes as a participle with the function of a verb, such as in he is going inside.

2.1.3. Verb - adjective.

Many verbs have the same shape as adjectives. Often an unvocalised verb with three radicals has the same pattern as an adjective. The three radicals , for example, can both stand for the verb · and the adjective .

2.1.4. Verb - noun.

The most important mingling of word patterns between verbs and nouns occurs with the verbal nouns (*masdar*). The verbal nouns of the fifth and the sixth form often raise confusion. For example (Vth form) can both be a verb (*to meddle*) and a noun (*interference*) and also (VIst form) can both be a verb (*to help*) and a noun (*cooperation*). However, the verbal nouns of the Vth and VIst form are easily detectable in a written text. The verbal nouns of the Ist form, on the other hand, are much more difficult to define as verbal noun, because these forms can often also be used as a noun. But also nouns are mixed up with verbs, such as, for example, the shape which can be a noun (*delegation*) or a verb (*to arrive*).

2.1.5. Verb - noun - adjective

The pattern is even more complicated. This pattern offers at least three possibilities, viz. a noun, an adjective or a verb. The word , for instance, means both *white* as a *white* (a member of the white race). However, it can also have the function of a verb in the sentence *what is his face white*! in which, according to the Arab grammarians, the form is considered to be a verb.

2.1.6. The *taa marbuta* element

One morphological element, which might seem to help to disambiguate words is the *taa marbuta* (Khoja 2002), which is considered in grammar to be the indication of a feminine noun par excellence. There are however exceptions, for instance, the rare forms, such as and (*excellence*) which are masculine and the pattern · in which represents an adjective meaning very learned.

The above elements show that it is not sufficient to take a lexicon and tag it. Many ambiguities are not resolved that way. Only the completely unambiguous forms will be tagged, but it is clear that most of the others will not. This does not mean that the tagging of words in a lexicon is not helpful. One might suppose that when going into more detail, word patterns which can contain two or more grammatical categories, and which are for that reason ambiguous, lose in quite a number of cases this ambiguity when they are translated in their practical word form. The above mentioned word (*calamitous*) for instance is clearly an adjective.

This however remains very tricky, because a word in Arabic, which in its concrete form is clearly an adjective but of which the theoretical form is ambiguous, can always by one Arab author or another be used as substantive. Arab authors often renew the style of the language and the language itself precisely by enlarging the meaning of already existing forms. The case of illustrates this clearly. As one can discover in the dictionary of Hans Wehr, the word (*president*) is definitely only a noun. No other meanings are given in this dictionary. However, corpus analysis of radio texts of Algeria revealed that this word in this pattern is often used as an adjective, meaning principle. Even when basing ourselves on existing lexicons, we cannot guarantee the distinct definition of parts of speech for Arabic words.

2.2. Second level: Derived word forms or conjugated forms

Not only on the level of the core forms there are many ambiguities, the same goes for derived word forms or conjugated forms. Due to the lack of vocalization the conjugation of verbs yields many ambiguous word patterns that are quite difficult to interpret without any valid context. can have four possible meanings*: I wrote*, *you wrote* (m. and f.) or *she wrote*. New ambiguities arise with the conjugated forms of verbs, not only within a conjugational level but also between different conjugational levels. The verb forms in the past tense of the first person singular, the second person masculine singular, the second person feminine singular and third person feminine singular all have the same shape. But new ambiguities arise between, for example, imperative forms and indicative forms. The shape can mean either I write or the imperative form write, but it can also be the third person in the past tense of a verb of the IVth form (*to dictate*).

Also, these derived forms interfere often with similar forms from other words, which makes the correct indication of the tag even more complex. Here too different grammatical categories mix up. In some cases the form does not only lead to the confusion mentioned above, but can even have a form that goes beyond the grammatical categories of a verb such as an elative.

The character combinations of nouns also can have the same shape of conjugated verbs. For example, the first person of the jussive form of the verb (*to build*), which becomes and hence is a pattern of consonants which mixes up with the noun (*son*). Derivate forms of adjectives too can have the same shape as nouns. Many feminine forms of adjectives ending with the nisba do correspond in their shape with feminine nouns. For example the feminine adjective (*personal*) which corresponds to the noun (*personality*).

2.3. Third level: Agglutinative forms of words

Not only isolated morphological forms can be dubious, but also the agglutinative character of the language provokes unexpected ambiguities between strings of

characters between two blanks. The combination of the conjunction with the particle corresponds to the verb (*to take fire*). These new ambiguities can occur with all combinations of particles or conjunctions that are being written directly to the word. The combination of the conjunction with the verb (*to cut*) corresponds to the verb *(to be detestable*). Both the particle and provoke the same kind of ambiguities. For example, the preposition in combination with (*hand*), which corresponds to the subordinate conjunction (*however*). And the preposition , for example, in combination with (*part*), which corresponds to the verb (*to regard*).

3. Automatic vocalization, a solution?

One might argue that the vocalization of an Arabic corpus might solve the problems of tagging. This is only true to a certain extent. First of all, the above shown ambiguities indicate that it is not at all self-evident to make a tagger which disambiguates Arabic raw texts by vocalizing them. Even then, algorithms will have to be written in order to apply the correct grammatical categories to the different lemmas in a text. But even so, in a completely vocalized text, ambiguities remain, as far as grammatical part of speech tagging is concerned, be it that overall ambiguity in a vocalized text is quite lower than in an unvocalized text. It is clear that on all three discussed levels a degree of ambiguity remains.

3.1. Ambiguities on the first level

On the first level, which is the level of the core word, ambiguity remains in the forms , , and in all the words which have the form of a participle, both active and passive, such as those of the form , and all their derived forms. All those forms can be both an adjective and a noun even when they are completely vocalized. On the same level ambiguities remain also between, for example, some verbs and nouns, such as the noun in the meaning of totality, and the verb with the meaning to be tired.

3.2. Ambiguities on the second level

On the second level this is valid for many word forms ending in a nisba followed by a taa marbuta. The complete vocalized word for instance, does not give any more information on the exact grammatical part of speech to apply. The same goes for every word with this pattern.

3.3. Ambiguities on the third level

On the third level also, new problems arise. Word forms, which were not ambiguous on the first level in their core form, become ambiguous and mix up with other words. The very frequent collocation (conjunction - preposition) (and in) has the same shape as the adjective (faithful). There are many other agglutinated word combinations, which mix up with existing core word forms. Another example is (verb = *to brand*), which mixes up with (conjunction + verb = and *he poisoned*). Exclusively basing the tagging of Arabic texts on a lexicon is therefore not sufficient. Indeed, the analysis of in detail tagged corpora gives additional information which might be of great use for the tagging of raw corpora.

4. Information to be derived from tagged corpora

The additional information, to be derived from tagged corpora, is both a statistical one and a grammatical one. Both kinds of information can make a high contribution for the tagging of corpora. Both remain to a certain extent probabilistic.

4.1. Statistical information

The statistical element is evident. Many core word forms in Arabic are no longer used in MSA. The dictionary of Hans Wehr contains many words which seem to be out of use nowadays. In compiling our dictionary MSA-Dutch * Dutch-MSA (Van Mol & Berghman, 2001), which is based on a corpus of 3,000,000 words both from oral and written resources, we discovered that many root patterns did not occur in our corpus. Our corpus contains only words in texts dated from 1980 onward. For instance, no word relating to the stem occurs in the corpus.

The fact that in MSA some word forms are less used is very important for the automatic tagging of corpora. Let us take the form as an example. At first sight this shape is a preposition and a personal pronoun meaning for you. The shape occurs very frequently in MSA. There is, however, an identical form which is a verb meaning to hit with the fist. This form however did not occur at all in our corpus. This means that a count of words in a completely disambiguated corpus can give much relevant information as far as automatic disambiguation of words is concerned or at least it can give a hint about the probability in tagging certain shapes according to the word count statistics.

4.2. The Leuven approach

4.2.1. The encoding system

In order to make preparations for the automatic tagging of Arabic corpora, we developed at the Institute of Living Languages of the Catholic University of Leuven a system to encode Arabic corpora. This system not only enables us to identify strings of characters and to analyze them, it also disambiguates words and makes it possible to label all kinds of strings of characters by the appropriate grammatical information. The disambiguation of words is made by using the Arabic diacritical signs in a special structured systematic way.

As an example we take the root . This shape can be a verb (*to accept*), an adverb (*before*), a noun (*front* part), a preposition *near*) or another preposition (*before*) and even a verb of the second form, if the sjadda is omitted (*kiss*). In order to disambiguate between these different shapes we apply the diacritical signs according to a systematic description. Basically these rules can briefly be summarized as follows:

the basic form of a verb is never vocalized (e.g.). The first consonant of a noun is always vocalized (e.g.). The last consonant of a preposition is always vocalized (e.g.). If there is more than one preposition with the same shape, the second consonant is vocalized as well (e.g.). Adverbs normally take the alif, if not, the last consonant is vocalized, such as in . Derived verb forms, such as those of the second form are always written with the sjadda.

4.2.2. The lexical database

At the institute we develop two kinds of databases, one word-oriented and one sentence-oriented. The word-oriented database contains until now 26,000 Arabic lemmas with all the relevant grammatical information. The words in this database were all disambiguated by way of our encoding system. After every word has been disambiguated by using the diacritical signs in a selective way, the grammatical categories are allocated for those words. Until now this has been done for approximately 20,000 words in this database. Linked to this database is a dictionary Arabic - Dutch v.v. which has recently been published in book form (Van Mol & Berghman, 2001)

4.2.3. The corpus

The second database contains a text corpus of approximatively 4,000,000-tagged Arabic words of which 1,200,000 from spoken Arabic language resources. Both databases will be used in the future in order to develop a semi-automatic tagging of raw Arabic corpora. In order to do so several steps have to be taken, the first of which is uniformisation.

4.2.4. The tagging preparations

In order to open up Arabic electronic corpora for a large variety of purposes a pre treatment seems to be necessary. This treatment comprises three main phases.

4.2.4.1. The uniformisation of Arabic corpora

Uniformisation means that all possible shapes of one word in a raw text are reduced to one identical shape. Due to the fact that, contrary to other languages, there is quite a large freedom in typing Arabic language, the ambiguity or the variety in writing Arabic is quite large. As the computer only recognizes ASCII codes, a minimal amount of standardization seems to be prerequisite.

Even when we have a detailed lexical database of which there are minimum two kinds of information, viz. all the words in their vocalized form, but also in their neutral unvocalised form, it is not always self-evident to find the right matches between words occurring in the database, and words occurring in a raw corpus. A few examples can make this clear. One of the problematic Arabic characters is the *alif*. The *alif* can be written, without a *hamza* or with a *hamza*. This means that when in a database the word (boys) is stored as the unvocalised word form for the vocalized form it is not always certain that this will match with a corresponding word form in the raw corpus, such as, for example, the form , because the ASCII code of both *alif*-forms differs.

Another example is the use of the *alif maqsura*, for example in the word , in Egyptian newspapers, whereas in most other countries the *ya'* is used (e.g.). Other elements which ought to be uniformized is the use of the *sjadda*. Words of which the word pattern contains both alif and sjadda are even more complicated. In those cases uniformization is not a matter of reducing two forms to one. Without counting the vowels, words with alif and sjadda can have up to eight different forms in a raw text. For example, the word (*European*), which can be written , , , , , or .

4.2.4.2. The identification of strings of characters

The second phase involves the identification of strings of characters. In order to do so we develop a two-level approach.

The first approach departs from the word-oriented database from which all possible minimal basic forms of words are generated. For every word we generate all possible, what we might call, minimal basic forms. The minimal basic form contains all the possible prefixes and suffixes which can be added to a word, but which still is part of it. On the other hand we also produce for every word all, theoretic possible, maximal basic forms, which correspond to the possible word combinations between two blanks. All of these forms are given a minimal encoding so that every added linguistic element has an unambiguous shape.

The second approach departs from the sentence-oriented database from which all possible maximal basic forms are retrieved.

The third phase involves the disambiguation of words on the basis of information coming from both sources. Once the corpus tagged this way, it contains enough detailed information to make scientific searches and analyses.

Conclusion

In our view, the best way to make preparations for the automatic annotation of Arabic corpora will be by using a completely in detail annotated corpus which will give a more detailed insight in the distribution of the different Arabic word patterns and their corresponding grammatical category. We hope to give in the near future much more details on the degree of ambiguity on the three word levels, the core word level, the derived word forms or conjugated forms and agglutinative forms of words. Those data will be compared with the data retrieved from the annotated test corpus.

References:

Cantarino, V. (1974). Syntax of modern Arabic prose, 3 Vol, Bloomington, London.

Ditters, E. (1992). A formal approach to Arabic Syntax: the noun phrase and the verb phrase, Amsterdam.

Van Mol, M. & Berghman, K. (2001a). Leerwoordenboek Modern Arabisch - Nederlands, The Dutch Language Union, Amsterdam, Bulaaq

Van Mol, M. & Berghman, K. (2001b). Leerwoordenboek Nederlands - Modern Arabisch, The Dutch Language Union, Amsterdam, Bulaaq.

We thank The MITRE Corporation for their
administrative support
in the organization of the workshop

The MITRE Corporation
202 Burlington Road
Bedford, MA 01730

MITRE

Multimodal Resources and Multimodal Systems Evaluation

Workshop Program

Saturday, June 1, 2002

Palacio de Congreso de Canarias

8:00 a.m. Welcome
Mark Maybury (*MITRE, USA*) and Jean-Claude Martin (*LIMSI-CNRS, France*)

Resources and Annotation: Multimodal

8:30 a.m. Data Resources and Annotation Schemes for Natural Interactivity
Laila Dybkjær and Niels Ole Bernsen
University of Southern Denmark, Denmark

8:50 a.m. Metadata Set and Tools for Multimedia/Multimodal Language Resources
P. Wittenburg, D. Broeder, Freddy Offenga, and Don Willems,
Max Planck Institute for Psycholinguistics, The Netherlands

Resources and Annotation: Gesture and Speech

9:10 a.m. FORM: A Kinematic Annotation Scheme and Tool for Gesture Annotation
Craig Martell, Chris Osborn, Jesse Friedman, and Paul Howard,
University of Pennsylvania, USA

9:30 a.m. Cross-Linguistic Studies of Multimodal Communication
P. Wittenburg, S. Kita, and H. Brugman,
Max Planck Institute for Psycholinguistics, The Netherlands

Resources and Annotation: Facial Expressions, Speech, Integration

9:50 a.m. Development of User-State Conventions for Multimodal Corpus in SmartKom
Silke Steininger, Susen Rabold, Olga Dioubina, and Florian Schiel,
Ludwig-Maximilians University, Munich, Germany

10:10 a.m. Integration of multi-modal data and annotations into a simple extendable form: the extension of the BAS Partitur Format
Florian Schiel, Silke Steininger, Nicole Beringer,
Ulrich Tuerk, and Susen Rabold,
University of Munich, Germany

10:40 a.m. Multimodal Resources Group Discussion
All

11:00 – 11:20 a.m. Morning Break

Annotation Tools

11:20 a.m. Multimodal Corpus Authoring System
Anthony Baldry, Univ. of Pavia, Italy, and Christopher Taylor, Univ. of Trieste, Italy

11:40 a.m. The Observer Video-Pro: Professional System for Collection, Analysis and Presentation of Observational Data
Niels Cadée, Erik Meyer, Hans Theuws, and Lucas Noldus, Noldus Information Technology, The Netherlands

11:20 a.m. Data Resources and Annotation Schemes for Natural Interactivity
Laila Dybkjær and Niels Ole Bernsen
University of Southern Denmark, Denmark

11:40 a.m. Metadata Set and Tools for Multimedia/Multimodal Language Resources
P. Wittenburg, D. Broeder, Freddy Offenga, and Don Willems, Max Planck Institute for Psycholinguistics, The Netherlands

13:00 p.m. Lunch

Multimodal Fusion

14:30 p.m. Prosody based co-analysis of Deictic Gestures and Speech in Weather Narration Broadcast
Kettebekov Sanshzar,Yeasin Mohammed, Krahnstoever Nils, SharmaRajeev, Dept. of CS and Engineering, Pennsylvania State University, USA

14:50 p.m. A Generic Formal Description Technique for Fusion Mechanisms of Multimodal Interactive Systems
Philippe Palanque and Amélie Schyn, LIIHS – IRIT, Université Toulouse, France

Research Infrastructure

15:10 p.m. Eye Bed
Ted Selker, Winslow Burleson, Jessica Scott, and Mike Li, MIT Media Lab, Cambridge, USA

15:40 pm. MUMIN: A Nordic Network for MUltiModal Interfaces
Patrizia Paggio, Center for Sprogteknologi, Copenhagen, Kristiina Jokinen, University of Art and Design, Helsinki, and Arne Jönsson, University of Linköping

System Evaluation

16:00 pm PROMISE - A Procedure for Multimodal Interactive System Evaluation

Nicole Beringer, Ute Kartal, Katerina Louka, Florian Schiel, Uli Türk, University of Munich, Germany

16:30 - 17:00 p.m. **Afternoon Break**

17:00 p.m. Final Group Discussion
All

18:00 p.m. Close

Table of Contents

Table of Contents (Concluded)

Preface

Motivation

Individual organizations and countries have been investing in the creation of resources and methods for the evaluation of resources, technologies, products and applications. This is evident in the US DARPA HLT programme, the EU HLT programme under FP5-IST, the German MTI Program, the Francophone AUF programme and others. The European 6th Framework program (FP6[1]), planned for a start in 2003, includes multilingual and multisensorial communication as major R&D issues. Substantial mutual benefits can be expected from addressing these issues through international cooperation. Nowhere is this more important than in the relatively new areas of multimedia (i.e., text, audio, video), multimodal (visual, auditory, tactile), and multicodal (language, graphics, gesture) communication.

Multimodal resources are concerned with the capture and annotation of multiple modalities such as speech, hand gesture, gaze, facial expression, body posture, graphics, etc. Until recently, only a handful of researchers have been engaged in the development of multimodal resources and their application in systems. Even so, most have focused on a limited set of modalities, custom annotation schemes, within a particular application domain and within a particular discipline. Until now, the collection and annotation of multimodal corpora has been made on an individual basis; individual researchers and teams typically develop custom coding schemes and tools within narrow task domains. As a result, there is a distinct lack of shared knowledge and understanding in terms of how to compare various coding schemes and tools. This makes it difficult to bootstrap off of the results and experiences of others. Given that the annotation of corpora (particularly multimodal corpora) is very costly, we anticipate a growing need for the development of tools and methodologies that enable the collaborative building and sharing of multimodal resources.

Increased International Attention

Recently, several projects, initiatives and organisations have addressed multimodal resources with a federative approach:

- At LREC2000, a workshop addressed the issue of multimodal corpora, focusing on meta-descriptions and large corpora http://www.mpi.nl/world/ISLE/events/LREC%202000/LREC2000.htm
- NIMM is a working group on Natural Interaction and Multimodality under the IST-ISLE project (http://isle.nis.sdu.dk/). Since 2001, NIMM has been engaged with conducting a survey of multimodal resources, coding schemes and annotation tools. Currently, more than 60 corpora are described in the survey. The ISLE project is developed both in Europe and in the USA (http://www.ldc.upenn.edu/sb/isle.html).
- In November 2001, ELRA (European Language Resources Association) conducted a survey of multimodal corpora including marketing aspects (http://www.icp.inpg.fr/ELRA/).
- In November 2001, a Working Group at the Dagstuhl Seminar on Multimodal Fusion and Coordination received 28 completed questionnaires from participating researchers; 21 announced their intention to collect and annotate multimodal corpora in the future. (http://www.dfki.de/~wahlster/Dagstuhl_Multi_Modality/)
- Several recent surveys have focused specifically on multimodal annotation coding schemes and tools (COCOSDA, LDC, MITRE).

[1] http://www.cordis.lu/rtd2002/fp-debate/fp.htm

Other recent initiatives in the United States include:

- NIST Automatic Meeting Transcription Project (http://www.nist.gov/speech/test_beds/mr_proj): "The National Institute of Standards and Technology (NIST) held an all-day workshop entitled "Automatic Meeting Transcription Data Collection and Annotation" on 2 November 2001. "The workshop addressed issues in data collection and annotation approaches, data sharing, common annotation standards and tools, and distribution of corpora. ... To collect data representative of what might be expected in a functional meeting room of the future, [NIST has] created a media- and sensor-enriched conference room containing a variety of cameras and microphones."
- ATLAS (http://www.nist.gov/speech/atlas): Also at NIST, "ATLAS (Architecture and Tools for Linguistic Analysis Systems) is a recent initiative involving NIST, LDC and MITRE. ATLAS addresses an array of applications needs spanning corpus construction, evaluation infrastructure, and multimodal visualisation."
- TALKBANK (http://www.talkbank.org): TALKBANK is funded by the National Science Foundation (NSF). Its goal "is to foster fundamental research in the study of human and animal communication. TalkBank will provide standards and tools for creating, searching, and publishing primary materials via networked computers." One of the six sub-groups is concerned with communication by gesture and sign.

Objective

The primary purpose of this one day workshop (feeding into a subsequent half day Multimodal Roadmap workshop) is to report and discuss multimodal resources, annotation standards, tools and methods, and evaluation metrics/methods, as well as strategize jointly about the way forward. The workshop consists of short presentations and facilitated sessions with the intent of jointly identifying grand challenge problems, a shared understanding of and plan for multimedia resources and applications, and identification of methods for facilitating the creation of multimedia resources.

Scope

The workshop focuses on multimodal resources, annotation and evaluation. Workshop participants were encouraged to annotate multimodal corpora samples using their own coding scheme or tool and report results at the workshop. Topics in the call for papers, listed in its entirety at http://www.lrec-conf.org/lrec2002/lrec/wksh/Multimodality.html, included but were not limited to:

- Guidelines, standards, specifications, models and best practices for multimedia and multimodal LR
- Methods, tools, and procedures for the acquisition, creation, management, access, distribution, and use of multimedia and multimodal LR
- Methods for the extraction and acquisition of knowledge (e.g. lexical information, modality modelling) from multimedia and multimodal LR
- Integration of multiple modalities in LR (speech, vision, language)
- Ontological aspects of the creation and use of multimodal LR
- Machine learning for and from multimedia (i.e., text, audio, video), multimodal (visual, auditory, tactile), and multicodal (language, graphics, gesture) communication
- Exploitation of multimodal LR in different types of applications (information extraction, information retrieval, meeting transcription, multisensorial interfaces, translation, summarisation, www services, etc.)
- Multimodal information presentation
- Multimedia and multimodal metadata descriptions of LR
- Applications enabled by multimedia and multimodal LR
- Benchmarking of systems and products; use of multimodal corpora for the evaluation of real systems
- Processing and evaluation of mixed spoken, typed, and cursive (e.g., pen) language processing
- Evaluation of multimodal document retrieval systems (including detection, indexing, filtering, alerting, question answering, etc.)
- Automated multimodal fusion and/or multimodal generation (e.g., coordinated speech, gaze, gesture, facial expressions)

Table 1 below lists the papers included in the workshop, the primary task focus of the article, the kinds of modalities focused on, and the multimodal research issues addressed in the papers.

TABLE 1. Overview of Contributions

Focus	Contribution	Author(s)	Modality	Research Issues
Resources and Annotation	Data Resources and Annotation Schemes for Natural Interactivity: Purposes and Needs	Laila Dybkjær and Niels Ole Bernsen	multimodal	Natural interactivity, data resources, coding schemes, coding purposes, coding needs
Resources and Annotation	Metadata Set and Tools for Multimedia/Multimodal Language Resources	P. Wittenburg, D. Broeder, Freddy Offenga, Don Willems	multimodal	Metadata
Resources and Annotation	FORM: A Kinematic Annotation Scheme and Tool for Gesture Annotation	Craig Martell, Chris Osborn, Jesse Friedman	gesture and speech	Gesture, Gesture Annotation, Multimodal Annotation, Annotation Tools, Annotation Graph Formalism
Resources and Annotation	Multimodal Annotation Sample	Craig Martell	gesture	Gesture annotation
Resources and Annotation	Cross-Linguistic Studies of Multimodal Communication	P. Wittenburg, S. Kita, H. Brugman	Gesture and speech	Cross-linguistic studies of multimodal communication
Resources and Annotation	Development of the User−State Conventions for the Multimodal Corpus in SmartKom	Silke Steininger, Susen Rabold, Olga Dioubina, Florian Schiel	multimodal (facial expressions and speech prosody)	multi−modal, annotation, user−states, human−machine interaction, coding conventions
Resources and Annotation	Integration of multi-modal data and annotations into a simple extendable form: the extension of the BAS Partitur Format	Florian Schiel, Silke Steininger, Nicole Beringer, Ulrich Tuerk, Susen Rabold	multimodal	integration, multimodal, annotation Quick Time, BAS Partitur Format
Annotation Tools	Multimodal Corpus Authoring System	Anthony Baldry, Christopher Taylor	multimodal	Multimodality, cocordancing, text, resources, translation
Annotation Tools	The Observer Video-Pro: Professional system for collection, analysis and presentation of observational data	Niels Cadée	multimodal	methods, tools, and procedures for the acquisition, creation, management, access, distribution, and the use of multimedia and multimodal language resources

TABLE 1. Overview of Contributions (Continued)

Focus	Contribution	Author(s)	Modality	Research Issues
Multimodal fusion	Prosody based co-analysis of Deictic Gestures and Speech in Weather Narration Broadcast	Kettebekov Sanshzar,Yeasin Mohammed, Krahnstoever Nils, SharmaRajeev	speech and gesture	Multimodal, gesture, prosody, modality integration, speech gesture co-occurrence
Multimodal fusion	A Generic Formal Description Technique for Fusion Mechanisms of Multimodal Interactive Systems	Philippe Palanque, Amélie Schyn	multimodal	Formal description techniques, multimodal systems engineering, fusion mechanisms.
Gaze interaction	A Test-Bed for Intelligent Eye Research	Ted Selker	gaze	Gaze interaction system
Multimodal System Evaluation	PROMISE - A Procedure for Multimodal Interactive System Evaluation	Nicole Beringer, Ute Kartal, Katerina Louka, Florian Schiel*, Uli Türk	multimodal	Multimodality, SmartKom, dialogue system evaluation, evaluation framework
Research Infrastructure	MUMIN: A Nordic Network for MUltiModal INterfaces	Patrizia Paggio, Kristiina Jokinen, Arne Jönsson	multimodal	Multimodal integration, cognitive and usability studies, multimodal dialogue, multimodal research and resources in the Nordic Countries

Any international workshop demands the selfless contributions of many individuals. We first thank the authors and participants for their important contributions. We next thank the Organizing Committee for their time and effort in providing detailed and high quality reviews and counsel. And we thank Paula MacDonald at MITRE for her excellent administrative workshop support.

Mark Maybury and Jean-Claude Martin
Workshop Co-chairs

Workshop Organizers

Mark Maybury (Co-chair)
The MITRE Corporation
Bedford, MA USA
maybury@mitre.org

Jean-Claude Martin (Co-chair)
LIMSI-CNRS, LINC-University Paris 8
Orsay, France
martin@limsi.fr

Workshop Program Committee

Niels Ole Bernsen
NISLab
University of Southern Denmark
Odense, Denmark
nob@nis.sdu.dk

Dybkjaer Laila
NISLab
University of Southern Denmark
Odense, Denmark
laila@nis.sdu.dk

Harry Bunt
Tilburg University
Harry.Bunt@kub.nl

Catherine Pelachaud
University of Rome "La Sapienza"
Italy
cath@dis.uniroma1.it

Lisa Harper
The MITRE Corporation
USA
lisah@mitre.org

Oliviero Stock
IRST
stock@irst.itc.it

Michael Kipp
DFKI
Germany
kipp@dfki.de

Wolfgang Wahlster
DFKI
Germany
wahlster@dfki.uni-sb.de

Steven Krauwer
ELSNET
steven.krauwer@elsnet.org

Antonio Zampolli
Consiglio Nazionale delle Ricerche
pisa@ilc.pi.cnr.it

Author's Index

Data Resources and Annotation Schemes for Natural Interactivity: Purposes and Needs

Laila Dybkjær and Niels Ole Bernsen

Natural Interactive Systems Laboratory
University of Southern Denmark
Science Park 10, 5230 Odense M, Denmark
{laila, nob}@nis.sdu.dk

Abstract

This paper reports on work carried out in the ISLE project on natural interactivity and multimodal resources. Information has been collected on a large number of corpora, coding schemes and coding tools world-wide. The paper focuses on corpora and coding schemes and the purposes for which they were developed or which they could serve.

1. Introduction

The long-term vision of natural interactivity envisions that humans communicate, or exchange information, with machines (or systems) in the same ways in which humans communicate with one another, using thoroughly co-ordinated speech, gesture, gaze, facial expression, head movement, bodily posture, and object manipulation [Bernsen 2001]. The idea of multimodality is to improve human-system interaction in various ways by using novel combinations of (unimodal) input/output modalities [Bernsen 2002]. Natural interactivity is by nature (mostly) multimodal. Across the world, researchers and companies are beginning to tap the potential of natural interactive and multimodal systems. This emerging community needs information about what is already there, how they might access it, what they might use it for, etc., in order that fewer people try to re-invent the wheel than would otherwise risk being the case. In many ways, we are only at the start of what could be a revolution in human-system interaction. It will be some time before a new community of researchers and developers, coming from what is currently an archipelago of widely dispersed areas and specialties, has consolidated in this most exciting field of exploration.

This paper provides an overview of selected aspects of the information on data resources (corpora) and annotation schemes that was collected in the European Natural Interactivity and Multimodality (NIMM) Working Group of the joint EU-HLT/US-NSF project International Standards for Language Engineering (ISLE).

ISLE is the successor of EAGLES (European Advisory Group for Language Engineering Standards) I and II and includes three working groups on lexicons, machine translation evaluation, and NIMM, respectively. The NIMM Working Group (isle.nis.sdu.dk) began its work in early 2000 and has now completed three comprehensive surveys. The surveys address NIMM data, annotation schemes, and annotation tools, respectively. Focus has been on producing descriptions which are systematically organised, follow standard formats, have been verified by the resource creators themselves, and provide interested parties in research and industry with the information they need to decide if a particular resource matches their interests. Each resource (data, coding scheme or tool) comes with contact information on its creator(s) and on how to get access to it. To our knowledge, the surveys significantly contribute to our common knowledge of the state of the art in data, coding schemes, and tools for natural interactivity and multimodal interaction. It appears that no other published work has produced comparatively large collections of information on NIMM resources.

The survey of NIMM data resources [Knudsen et al. 2002a] includes a total of 64 resources world-wide, 36 of which are facial resources and 28 are gesture resources. Several data resources combine speech with facial expression and/or gesture. The report also includes a survey of market and user needs produced by ELRA (the European Language Resources Agency) and 28 filled questionnaires collected at the Dagstuhl workshop on Coordination and Fusion in Multimodal Interaction held in late 2001.

The survey of NIMM corpus annotation schemes [Knudsen et al. 2002b] includes 7 descriptions of annotation schemes for facial expression and speech, and 14 descriptions of annotation schemes for gesture and speech. In addition, the survey draws some conclusions on current coding best practices based on the collected material.

The survey of NIMM corpus coding tools [Dybkjær et al. 2001a] describes 12 annotation tools and ongoing tool development projects, most of which support speech annotation combined with gesture annotation, facial expression annotation, or both. Conclusions on requirements to be met by a general-purpose NIMM annotation tool are made and further refined in [Dybkjær et al. 2001b].

Based on the above ISLE NIMM reports, in particular [Knudsen et al. 2002a and 2002b], this paper reviews the purposes for which the surveyed data resources and coding schemes have been used or are intended to be used, and discusses annotation best practices.

2. Purposes of data resources

This section provides an overview of the purposes for which, according to their creators, the data resources collected in ISLE NIMM have been applied or are intended to be applied (Section 2.1). A summary is then presented of selected results from a market study performed by ELRA and included in [Knudsen et al. 2002a] (Section 2.2).

2.1. Data resources

Many of the 64 reviewed NIMM data resources were found via the web. Others were found through proceedings of specialised conferences and workshops [Knudsen et al. 2002a]. When a resource can be downloaded from the web, this is indicated in the report. For each data resource, contact information is provided so that the resource creators can be contacted and asked how to obtain the resource if it is not directly accessible.

The collected data resources reflect a multitude of needs and purposes, including the following (in random order):

- automatic analysis and recognition of facial expressions, including lip movements;
- audio-visual speech recognition;
- study of emotions, communicative facial expressions, phonetics, multimodal behaviour, etc.;
- creation of synthetic graphical interface characters, including, e.g., talking heads;
- automatic person identification;
- training of speech, gesture and emotion recognisers;
- multimodal system specification and development.

In many cases, the people working with the data, in particular those working with static image analysis, have created their own resource databases. Algorithms for image analysis are sometimes dependent on lighting conditions, picture size, subjects' face orientations, etc. Thus, computer vision research groups may have had to create their own image databases with good reason. Image analysis using computer vision techniques remains a difficult task, and this may be the reason why we have primarily found static image resources produced by workers in this field.

In other areas, (dynamic) video recordings - mostly including audio - are needed. For example, studies of lip movements during speech, co-articulation, audio-visual speech recognition, temporal correlations between speech and gesture, and relationships among gesture, facial expression, and speech, all require video recordings with audio.

Across the collected data resources, re-use is a rare phenomenon. If a resource has been created for a specific application purpose, it has usually been tailored to satisfy the particular needs of its creators, highlighting, e.g., particular kinds of interaction or the use of particular modality combinations. Figure 1 provides an overview of the data resources reviewed, including the purpose(s) for which they were created or have been used.

Modalities	**Name of data resource**	**Purpose(s)**
Dynamic face	LIMSI Gaze Corpus (CAPRE)	Track face, nose and eyes.
Dynamic face, audio	Advanced Multimedia Processing Lab	Lip reading, speech-reading techniques for higher speech recognition accuracy.
	ATR Database for bimodal speech recognition	Research, speech recognition and speech-to-lip generation (animated agents, talking face), observations on the differences in lighting conditions, size of lips, and inclination of a face.
	The BT DAVID Database	Research on audio-visual technologies in speech or person recognition, synthesis, and communication of audio-visual signals.
	Data resources from the SmartKom project	Collect data for the training of speech, gesture and emotion recognisers, to develop dialogue and context models and to investigate how users interact with a machine that has far greater communication skills than at present.
	FaceWorks	Enable multimedia developers to create digital personalities.
	M2VTS Multimodal Face Database	User authentication, lip tracking, face recognition, extend the scope of application of network-based services by adding novel and intelligent functionalities enabled by automatic verification systems combining multimodal strategies (secured access based on speech, image and other information).
	M2VTS Extended Multimodal Face Database – (XM2VTSDB)	Lip tracking, eye coordinate determination, face and speech authentication. Large multi-modal database, which will enable the research community to test their multi-modal face verification algorithms on a high-quality large dataset.
	Multi-talker database	Quantitatively characterize optical speech signals, examine how optical phonetic characteristics relate to acoustic and physiological speech production characteristics, study what affects the intelligibility of optical speech signals, and apply the knowledge obtained to optical speech synthesis and automatic speech recognition.

	VIDAS (VIDeo ASsisted with audio coding and representation)	Devise suitable methodologies and algorithms for time-correlated representation, coding and manipulation of digital A/V bit streams.
	/'VCV/ database	Study lip shape characterisation during speech.
	ATR Database for Talking Face	Research.
	Audio-Visual Speech Processing Project	Research.
	Video Rewrite	Facial animation system to automate all the labelling and assembly tasks required to resynchronise existing footage to a new soundtrack.
Dynamic face, audio, gesture	NITE Floorplan Corpus (Natural Interactivity Tools Engineering)	Test resource for cross level, cross modality analysis of natural interactive communication.
	Scan MMC (Score Analysed MultiModal Communication)	Research on facial expression and gesture.
	Multi-modal dialogue corpus	Research on multi-modal dialogue.
Static face	3D_RMA: 3D database	Validation of facial 3D face acquisition by structured light, recognition experiments by 3D comparison.
	AR Face Database	Create a better resource for face recognition and expression recognition.
	AT&T Laboratories Database of Faces	Face recognition research.
	CMU Pose, Illumination, and Expression (PIE) database	Collect material for the design and evaluation of face recognition algorithms (facial expression detection, temporal issues of facial expressions and other kinds of analysis of facial expressions).
	Cohn-Kanade AU-Coded Facial Expression Database	Develop and test algorithms for facial expression analysis.
	FERET Database Demo	Face recognition.
	Psychological Image Collection at Stirling (PICS)	Psychological research (visual perception, memory and processing).
	TULIPS 1.0	Test lip-tracking algorithms.
	UMIST Face Database	Examine pose-varying face recognition.
	University of Oulu Physics-Based Face Database	Face recognition under varying illuminant spectral power distribution.
	VASC – CMU Face Detection Databases	Train and test face detection algorithms.
	Visible Human Project	Studies of anatomy, creation of synthetic models and test image segmentation algorithms.
	Yale Face Database	Research on face recognition.
	Yale Face Database B	Face recognition under various poses and illumination.
	3D Surface Imaging in Medical Applications	Medical applications.
	Facial Feature Recognition using Neural Networks	Face recognition.
	Image Database of Facial Actions and Expressions	Train neural networks to classify facial behaviours based on FACS.
	JAFFE Facial Expression Image Database	Research on facial expression.
	Photobook	Tool for performing queries on image databases based on image content.
Gesture	MPI Experiments with Partial and Complete Callosotomy Patients Corpus	Research on split-brain patients.
	National Center for Sign Language and Gesture Resources	Support research on sign language.
	ATR sign language gesture corpora	Creation of an inventory of the most important words of Japanese sign language as a basis for the development and evaluation of gesture recognition systems.
Gesture, audio	ATR Multimodal human-human interaction database	Provide a source for analysing the relation between speech and gesture.

	CHCC OGI Multimodal Real Estate Map	Compare the linguistic differences and relative ease of processing multimodal input compared with unimodal input.
	GRC Multimodal Dialogue during Work Meeting	Study the patterns of multimodal communication during a work session about collaborative conception.
	LIMSI Pointing Gesture Corpus (PoG)	Basis for specification of a recognition system
	McGill University, School of Communication Sciences & Disorders, Corpus of gesture production during stuttered speech	Study relations between gesture and stuttered speech.
	MPI Historical Description of Local Environment Corpus	Research.
	MPI Living Space Description Corpus	Research.
	MPI Locally-situated Narratives Corpus	Research.
	MPI Narrative Elicited by an Animated Cartoon "Canary Row" Corpus 1	Research.
	MPI Narrative Elicited by an Animated Cartoon "Canary Row" Corpus 2	Research.
	MPI Narrative Elicited by an Animated Cartoon "Maus" and "Canary Row" Corpus	Research.
	MPI Natural Conversation Corpus	Research.
	MPI Naturalistic Route Description Corpus 1	Research.
	MPI Naturalistic Route Description Corpus 2	Research.
	MPI Traditional Mythical Stories Corpus	Research.
	MPI Traditional Mythical Stories with Sand Drawings Corpus	Research.
	National Autonomous University of Mexico, DIME multimodal corpus	Build and test an interactive multimodal Spanish spoken - graphics system to assist human users in a geometric design task (kitchen design).
	RWC Multimodal database of gestures and speech	Build a speech and video database that can be shared among different research groups pursuing similar work that will promote research and development of multimodal interactive systems integrating speech and video data.
	University of Chicago Origami Multimodal corpus	Study origami, study learner gestures (with and without speech, collaborative gestures), learner gestures in relation to instructor gestures.
	IRISA Georal Multimodal Corpus	Study how people use speech and gestures on a tactile screen to interact with a graphical tourist map.
	LORIA Multimodal Dialogues Corpus	Research.
Gesture, gaze, audio	VISLab Cross-Modal Analysis of Signal and Sense Data and Computational Resources for Gesture, Speech and Gaze Research	Understanding relationships between speech and gesture.
	LIMSI Multimodal Dialogues between Car Driver and Copilot Corpus	Study of multimodal communication between a driver and a co-pilot in different settings.
	University of Venice Multimodal Transcription of a Television Advertisement	Understanding the properties and functions of dynamic genres, including verbal and written discourse, gesture, gaze, colour, voice quality.
Gesture, face, audio	University of California Video Series on Nonverbal Communication	Research on non-verbal communication, including facial expressions, tones of voice, gestures, eye contact, spatial arrangements, patterns of touch, expressive movement, cultural differences, and other "nonverbal" acts.

Figure 1. The reviewed data resources and their purposes.

2.2. Market study

A market study on data resources and user needs was performed by ELRA. A questionnaire was sent to more than 150 people, including ELRA members and people from both industry and academia. 25 responses were received. Among others, the questionnaire included questions on (1) the types of data resources needed, used by, or offered by, respondents, (2) the kinds of task for which data resources are well suited, and (3) the areas in which data resources are being used.

2.2.1. Types of data resources needed or offered

The NIMM data resources in which the respondents seem most interested include audio, video and image resources. Audio is most popular (mentioned by 84% of the respondents) followed by video (mentioned by 52%) and image (mentioned by 28%). If a data resource has also been annotated, this is considered an advantage since value has been added. In many cases, the users of data resources produce the resources they need themselves. Sometimes these resources are also offered to other users.

Authentication: Speech verification (8), Face verification (6), User authentication (5). Other: finger print and signature, biometric authentication (speech, signature).

Recognition: Speech recognition (14), Face recognition (7), Person recognition (3), Expression recognition (3). Other: mimic, music and other sounds, gesture recognition, gestures on a touchscreen.

Analysis: Speech/lips correlation (7), Body movements tracking (lips, hands, head, arms, legs, etc.) (6). Other: co-operation between gesture and speech; acoustics, video, 3D optical, midsagital magnetometry; written language analysis.

Synthesis: Multimedia development (6), Talking heads (5), Humanoid agents (5), Avatars (2). Other: text generation.

Control: Voice control (7), Speech-assisted video (1).

Other: Information retrieval (14), Other: multimodal command languages (speech + gesture), research into cross-modality issues, multimodal dialogue (speech + gesture), linguistic research, information extraction, text summarisation.

Figure 2. Resource application list from the ELRA report in [Knudsen et al. 2002a, chapter 8]. Numbers in parentheses indicate how many respondents gave a particular answer.

2.2.2. What can data resources be used for

The questionnaire mentioned six general task categories for which data resources may be used. For each category, a number of more specific possibilities were listed. Respondents were supposed to indicate the kinds of applications they were interested in. Responses are shown in Figure 2. The primary applications of data resources are information retrieval and speech recognition, each of which were mentioned by 14 respondents. Then follows speech verification mentioned by 8, and face recognition, speech/lips correlation, and voice control, each mentioned by 7 respondents.

2.2.3. Application areas

To get an idea of the overall application or market areas for data resources, the questionnaire listed five possibilities (including "other") among which respondents were asked to choose the ones they found appropriate to their work. The area mentioned most frequently was research (21). Then follows information systems development (e.g. banking, tourism, telecommunication) (14), web applications development (10), education/training (9), and edutainment (6). Other areas proposed include security, control of consumer devices, and media archiving for content providers.

3. Purposes of annotation schemes

This section provides an overview of the purposes for which the reviewed coding schemes [Knudsen et al. 2002b] have been created or used (Section 3.1). Then follows a brief description of practices and best practices as these emerged from the collected material (Section 3.2).

3.1. Annotation schemes

There probably exists a wealth of NIMM annotation schemes most of which are tailored to a particular purpose and used solely by their creators or at the creators' site. Such coding schemes tend not to be very well described. They also tend to be hard to find. The reviewed material includes such coding schemes many of which were created by ISLE participants or people known to ISLE participants, this being the main reason why we were aware of them. Other coding schemes included are fairly general ones, in frequent use, or even considered standards in their field, cf. Section 3.2.

Nearly all the reviewed coding schemes are aimed at markup of video, possibly including audio. A couple of schemes can be used for static image markup.

The collected material comprises schemes for markup of a single modality as well as schemes for markup of modality combinations. Figure 3 provides an overview of the majority of the schemes reviewed, including the annotation purpose for which they were created. The coding scheme descriptions which have not been included below are of a more general nature and do not concern any particular coding scheme and its purpose(s).

3.2. Practices and best practices

In most cases, a coding scheme has been created because a person or site had a particular need, e.g. related to systems development.

In the area of facial expression, MPEG-4 is considered a standard and is being widely used. FACS is also used by many people but is not really well suited for markup of lip movements. ToonFace is good for 2D caricature but not for real (or life-like) facial expression. Other reviewed facial expression schemes seem to have been used by a single person or by a few people only.

In the area of gesture, the picture seems considerably more varied than for facial expression. Where facial expression is often the sole point of focus, gesture often seems to be studied along with other modalities. Only when it comes to the highly specialised area of sign

languages, the schemes we looked at focused solely on gesture. Many other gesture schemes were created to study gesture in combination with one or several other modalities with the purpose of supporting the development of a multimodal system. There are no real standards for gesture markup. HamNoSys seems to be the most frequently used among the schemes we looked at as regards gesture annotation-only. For gesture in combination with other modalities there are many schemes – mostly used by few people - but no standardisation.

The picture, provided by the survey, of a proliferation of home-grown coding schemes is supported by the 28 questionnaires in [Knudsen et al. 2002a], asking people at a multimodal interaction workshop, e.g., which coding scheme(s) they had used or planned to use for data markup. Some people did not answer the question or had not made a decision yet as to which coding scheme to use. However, in no less than 15 cases the answer indicated that a custom-made scheme would be, or was being, used. Only a few respondents also mentioned more frequently used annotation schemes, such as TEI, BAS, or HamNoSys.

Intended for markup of	**Name of coding scheme**	**Purpose of creation**
Gaze	The alphabet of eyes	Analyse any single item of gaze in videotaped data.
Facial expression	FACS (facial action coding system)	Encode facial expressions by breaking them down into component movements of individual facial muscles (Action Units). Suitable for video or image.
	BABYFACS	Based on FACS but tailored to infants.
	MAX (Maximally Discriminative Facial Movement Coding System)	Measure emotion signals in the facial behaviours of infants and young children. Suitable for video or image.
	MPEG-4	Define a set of parameters to define and control facial models.
	ToonFace	Code facial expression with limited detail. Developed for easy creation of 2D synthetic interface agents.
Gesture	HamNoSys	Designed as a transcription scheme for (different) sign languages.
	SWML (SignWriting Markup Language)	Code utterances in sign languages written in the SignWriting System.
	MPI GesturePhone	Transcribe signs and gestures.
	MPI Movement Phase Coding Scheme	Coding of co-speech gestures and signs.
Speech and gesture	DIME (Multimodal extension of DAMSL)	Code multimodal behaviour (speech and mouse) observed in simulated sessions in order to specify a multimodal information system.
	HIAT (Halbinterpretative Arbeitstranskriptionen)	Describe and annotate parallel tracks of verbal and non-verbal (e.g. gestural) communication in a simple way.
	TYCOON	Annotation of available referable objects and references to such objects in each modality.
Text and gesture	TUSNELDA	Annotation of text-and-image-sequences, e.g. from comic strips.
Speech, gesture, gaze	LIMSI Coding Scheme for Multimodal Dialogues between Car Driver and Copilot	Annotation of a resource which contains multimodal dialogues between drivers and copilots during real car driving tasks. Speech, hand gesture, head gesture, gaze.
Speech, gesture and body movement	MPML (A Multimodal Presentation Markup Language with Character Agent Control Functions)	Allow users to encode the voice and animation of an agent guiding a web site visitor through a web site.
Speech, gesture, facial expression	SmartKom Coding scheme	Provide information about the intentional information contained in a gesture.

Figure 3. Reviewed coding schemes and their purposes.

4. Conclusion

Even if we have reviewed a large number of data resources and coding schemes, there probably exist many other NIMM corpora and coding schemes which we did not manage to identify. Many resources are not publicly accessible and their creators do not want to share them with others. Thus, they can be very hard to find. But also, our primary focus has been on resources which are accessible to people other than their creators. We believe that the collected information and resulting reports, although probably far from being exhaustive, reflect quite well the state-of-the-art in the NIMM resources area.

If this is indeed the case, some conclusions are: to a large extent, people still create their own single-purpose data resources and coding schemes without any strong guidance by best practice and standards, and hence without any strong purpose of sharing their resources with others. However, vendors of data resources exist, such as ELRA and LDC, and standards will emerge eventually and become applied. The standardisation process seems to be further advanced for facial expression than for gesture, and for gesture combined with other modalities there is still a long way to go.

In the ISLE project we do not have the resources required for regularly extending the information collected with new data, coding schemes or coding tools. Therefore, a web-based facility will be set up which will enable any interested colleague to upload information about a NIMM resource which has not been included already. We hope that our colleagues in the emerging NIMM community will use the facility to help each other by sharing their information with others and contribute to maintaining an up-to-date and valuable pool of NIMM resource information.

5. Acknowledgements

We gratefully acknowledge the support of the ISLE project by the European Commission's Human Language Technologies (HLT) Programme. We would also like to thank all ISLE NIMM participants for their report contributions which have made the present presentation possible. In particular, we have in this paper drawn on information provided by ELRA, Catherine Pelachaud, Isabella Poggi and Jean-Claude Martin.

6. References

Bernsen, N. O.: Multimodality in language and speech systems - from theory to design support tool. In Granström, B. (Ed.): Multimodality in Language and Speech Systems. Dordrecht: Kluwer Academic Publishers 2002 (to appear).

Bernsen, N. O.: Natural human-human-system interaction. In Earnshaw, Rae, Guedj, Richard, van Dam, Andries, and Vince, John (Eds.): Frontiers of Human-Centred Computing, On-Line Communities and Virtual Environments. Berlin: Springer Verlag 2001, Chapter 24, 347-363.

Dybkjær, L., Berman, S., Bernsen, N. O., Carletta, J., Heid, U. and Llisterri, J.: Requirements Specification for a Tool in Support of Annotation of Natural Interaction and Multimodal Data. ISLE Deliverable D11.2, 2001b.

Dybkjær, L., Berman, S., Kipp, M., Olsen, M. W., Pirrelli, V., Reithinger, N. and Soria, C.: Survey of Existing Tools, Standards and User Needs for Annotation of Natural Interaction and Multimodal Data. ISLE Deliverable D11.1, 2001a.

Knudsen, M. W., Martin, J. C., Dybkjær, L., Ayuso, M. J. M, N., Bernsen, N. O., Carletta, J., Kita, S., Heid, U., Llisterri, J., Pelachaud, C., Poggi, I., Reithinger, N., van ElsWijk, G. and Wittenburg, P.: Survey of Multimodal Annotation Schemes and Best Practice. ISLE Deliverable D9.1, 2002b.

Knudsen, M. W., Martin, J. C., Dybkjær, L., Berman, S., Bernsen, N. O., Choukri, K., Heid, U., Mapelli, V., Pelachaud, C., Poggi, I., van ElsWijk, G. and Wittenburg, P.: Survey of NIMM Data Resources, Current and Future User Profiles, Markets and User Needs for NIMM Resources. ISLE Deliverable D8.1, 2002a.

The reports referenced above are available at the website for the European ISLE NIMM Working Group at isle.nis.sdu.dk

Metadata Set and Tools for Multimedia/Multimodal Language Resources

P. Wittenburg, D. Broeder, F. Offenga, D. Willems

Max-Planck-Institute for Psycholinguistics
Wundtlaan 1, 6525 XD Nijmegen, The Netherlands
peter.wittenburg@mpi.nl

Abstract

Within the ISLE Project about International Standards for Language Engineering the IMDI Metadata Initiative developed a complete environment for creating, maintaining and using metadata descriptions for multimedia/multimodal language resources. This environment includes a proposal for a suitable metadata set, tools to create, browse and search in IMDI metadata domains and suggestions about how to organize centers acting as metadata repositories. By using the IMDI approach a formulation in RDF is intended which enable the IMDI set to be integrated in Semantic Web activities.

1. Introduction

In 1999 the Max-Planck Institute for Psycholinguistics started using metadata to organise its multi-media corpora [1]. This project was called "Browsable Corpus" (BC) because it not only used metadata for resources in order to make them locatable by automatic procedure, but it also used metadata for creating a hierarchical structure that can be browsed for the purpose of corpus exploitation. This was achieved by recursively structuring corpora in ever-smaller sub-corpora structures with each one described by its own metadata description pointing to the metadata descriptions of its sub-corpora. Creating browsable structures this way which creates space to integrate many other types of information such as project notes, also formed a basis for efficient corpus management.

The basic concepts of BC were used as one of the inputs to the ISLE Metadata Initiative (IMDI) [2] founded in early 2000. IMDI aims to reach consensus within a representative part of the linguistic community on a standard for metadata descriptions for multimedia/multimodal language resources. The IMDI metadata set is currently being applied within projects such as DOBES [3], the CGN corpus [4] and, of course the MPI's own corpora. Its relevance was checked for several other multimedia corpora such as the SmartKom [5] corpus. A preliminary showcase combined corpus data from 6 European institutions into one browsable and searchable domain.

2. Using Metadata Descriptions

A key issue in the IMDI approach is that a metadata set should be used for corpus discovery and corpus management as well as corpus exploitation. This implies that the metadata set should be able to describe the resources in sufficient detail to allow the resolution of relevant queries for the domain. It also implies that linked networks of metadata descriptions should be available, generated either automatically or manually and that it should be possible to include human readable texts or files with the metadata descriptions that can assist the user when browsing through a corpus. Corpora organized in this way can be easily integrated into bigger domains and they are an extremely useful facility for corpus managers to group all relevant information and knowledge together to facilitate corpus management. In this domain of linked metadata descriptions the user would be able to browse and search and as a result find a single resource or a sub-corpus to work on. Consequently the user is likely to want to start a suitable tool for analysis, i.e. the metadata must contain information which indicates which operations can be executed on the resources found. Within IMDI it was anticipated that each user has his own view on corpora, therefore it was concluded that the IMDI environment should provide users the possibility of creating their own hierarchies so that several views can co-exist in parallel.

Of course, metadata will always exist as a source of information distributed via Internet, therefore all resources including the metadata descriptions themselves have to be specified as URLs. In this way metadata descriptions and connected resources can be accessed on the Internet by using standard HTTP. This simplifies the connection of different corpus domains to one super-domain. To support global searches via, for example, Dublin Core [6] based service providers, the IMDI domain is available for metadata harvesting in compliance with the Open Archives Initiative protocol [7].

Although the concept of metadata descriptions is still fairly new, the community is becoming aware that metadata descriptions will facilitate re-usage of valuable resources. Currently, most of the many resources are hidden in the storage containers of the various institutions and companies. Only few of them are visible via web-sites each having its own style of description. Since metadata are available to everyone, a domain of unified descriptions form an ideal way of informing others about available data even if the resources themselves are not directly accessible.

3. IMDI Metadata Set

IMDI's guiding principles when defining a metadata set have been that the best way to describe linguistic resources is to be able to describe the events and/or performances that are involved in their creation and usage by the community. The descriptions need to contain as much detail as necessary for a user who needs to easily discover resources, quickly check their usefulness and immediately exploit them. This bottom up approach can be compared with the approach in the media and film community which defined the MPEG7 standard [8]. It can and will lead to a more extensive and structured set than, for instance, the Dublin Core set. In taking such an

approach, the metadata set found can be seen as a first step towards a more complex domain ontology.

Some argue that it is necessary to have a low-overhead metadata set, since users may not want to spend too much time in providing all the information defined by the proposed IMDI element set. For IMDI the solution is that efficient tools are provided and that almost all fields are optional. So the overhead argument in case of more elaborate metadata sets does not hold, if elements are optional as in the IMDI case. Flexibility of the set of elements was one of the recurrent requirements, since we deal with a large number of different projects all recording multimedia material. In IMDI, flexibility was introduced by allowing user definable keyword/value pairs at several levels in the metadata structure.

The IMDI set for sessions[1] contains the necessary elements to describe the project a resource belongs to, the responsible scientists who created it, date and location of the recording, its content, its media files and annotations and if available the its derivative source. In the following a list of all elements is given. It is not the purpose of this paper to explain in detail what all the elements represent. For this we refer to the IMDI web-site: http://www.mpi.nl/ISLE. An attribute specifies whether the element is just a string, constrained (c), associated with a closed vocabulary (ccv) as in the case of "Continents" or with an open vocabulary (ov) which is open for extensions, or refers to a sub-block of information (sub).

Session

Name	str
Title	str
Date	c

Location

Continent	ccv
Country	ccv
Region +	str
Address	str
Description[2]	sub
Keys [3]	sub

Project

Name	str
Title	str
ID	str
Contact	sub
Description +	sub

Collector

Name	str
Contact	sub
Description +	sub

Content

CommunicationContext	
Interactivity	ccv
PlanningType	ccv
Involvement	ccv
Genre	
Interactional	ovl
Discursive	ovl
Performance	ovl
Task	ovl
Modalities	ovl
Languages	
Description	sub
Language+	sub
Description +	sub
Keys	sub

Participants

Description +	sub
Participant+	
Type	ov
Name+	str
FullName	str
Code	str
Role	ov
Language+	sub
EthnicGroup	str
Age	c
Sex	ccv
Education	str
Anonymous	ccv
Description+	sub
Keys	sub

Resources

MediaFile+	
ResourceLink	c
Size	c
Type	ccv
Format	ov
Quality	c
RecordingCondition	str
Position	c
Access	sub
Description	sub
AnnotationUnit+	
ResourceLink	c
MediaID	c
Annotator	str
Date	c
Type	ov
Format	ov
ContentEncoding	str
CharacterEncoding	str
Access	sub
Language	sub
Anonymous	ccv
Description	sub
Source+	
ID	str

[1] Sessions are the leaves in a corpus tree and cover units of linguistic analysis or performance including their media and annotation files. The IMDI initiative has defined a few other very similar metadata sets for corpus nodes, published corpora and lexica. They are not discussed in this paper.

[2] Descriptions are a field which the annotator can use to enter prose text intended for quick inspection by the user.

[3] Keys are those fields which guarantee flexibility. Each project or even user can define extensions in form of key-value pairs.

Format	ov
Quality	ccv
Position	c
Access	sub
Description	sub

References

It is important to mention here how multimedia and multimodality can be described in IMDI. The IMDI set allows the user to describe the *Content* of a session which refers to a unit of analysis in the corpus. Each session is associated with the media and annotation resources belonging together. The IMDI set has elements to describe the *Communication Context*, the *Genre*, the *Task*, the *Modalities*, the *Languages involved*, and to add other useful project specific elements.

In most instances the associated vocabularies clarify what the definition of the element is although IMDI has already provided careful definitions. The element *Task* stands for typical experimental tasks occurring in language engineering and field-linguistics such as *info-kiosk situation, route description, wizard-of-oz experiment, frog-story.* The element *Modalities* has, of course, a vocabulary which includes, amongst others, *speech, gesture, sign, facial expression.*

As can be seen, the IMDI set has elements not only to describe content, but also to describe the *Media Files* (type of data, format of file, quality of material, conditions of recording, etc), the available *Annotations* (type of annotation, format of file, etc), and the *Original Media* (cassette, MD, etc) if available. To give the user immediate feedback ob accessibility, IMDI contains elements to describe the access rights and whom to contact to obtain the resources.

As already indicated, Controlled Vocabularies (CVs) associated with elements are an important component of the IMDI metadata set and its tools, since they will guarantee that elements are used coherently by researchers and that search operations will provide the correct resources.

To achieve interoperability with Dublin Core (a more general set of 15 partially vaguely defined elements used to describe web resources used by the general public) a mapping document was created. Based on DC, another set (OLAC [9]) was created to achieve interoperability in the language resource domain. IMDI repositories will be open to OAI [7] type of metadata harvesting to implement the interoperability with DC and OLAC.

The IMDI set is defined in all respects through an XML Schema which is available at the IMDI web-site. All tools generate and operate on these XML files.

4. IMDI Tools

The tools that support the IMDI metadata set and infrastructure are:

- ? The IMDI BCEditor that is used to create IMDI metadata descriptions.
- ? The IMDI BCBrowser. A viewer for the IMDI metadata descriptions that allows navigating the universe of connected IMDI metadata descriptions.
- ? The IMDI Search tool that allows the user to specify a query for specific resources in the IMDI universe.
- ? A number of scripts allowing to work efficiently

All tools were programmed in Java and Perl for platform independence and are downloadable from the web-site: http://www.mpi.nl/tools.

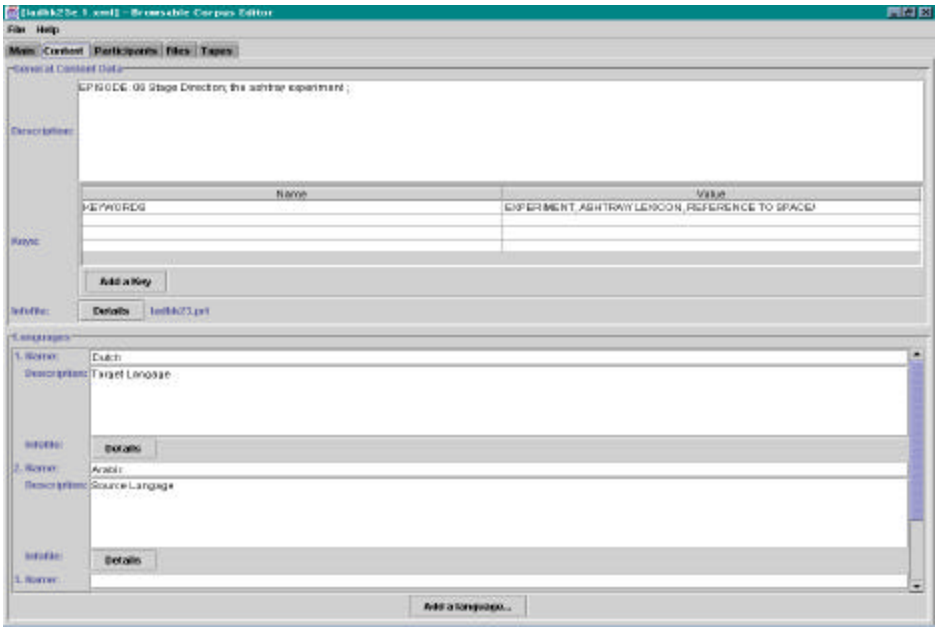


Figure 1 shows a screenshot from the IMDI Editor

The editor presents all the IMDI metadata elements in a structured GUI to the user. It supports the use of Controlled Vocabularies and user definable keyword/value pairs that the IMDI set allows for user or project specific extensions. Also it enforces constraints on the values for some metadata elements where applicable and practical. To aid working efficiency the editor allows the re-usage of a number of element blocks which will recur in many metadata descriptions such as biographical data of the informants and collectors. The editor is programmed to synchronize with repositories providing controlled vocabularies on user command if the computer the editor is running on is connected to the web. This mechanism ensures that the user can download and use the most recent definitions, e.g. of the names of countries. Internationally agreed notation conventions allow differences between different vocabularies. For example, the ISO language lists contain only a few hundred language names and the Ethnologue list [10] contains more than 4000 names. In fact users can add their own

lists but searching would become a problem if there is no mapping definition.

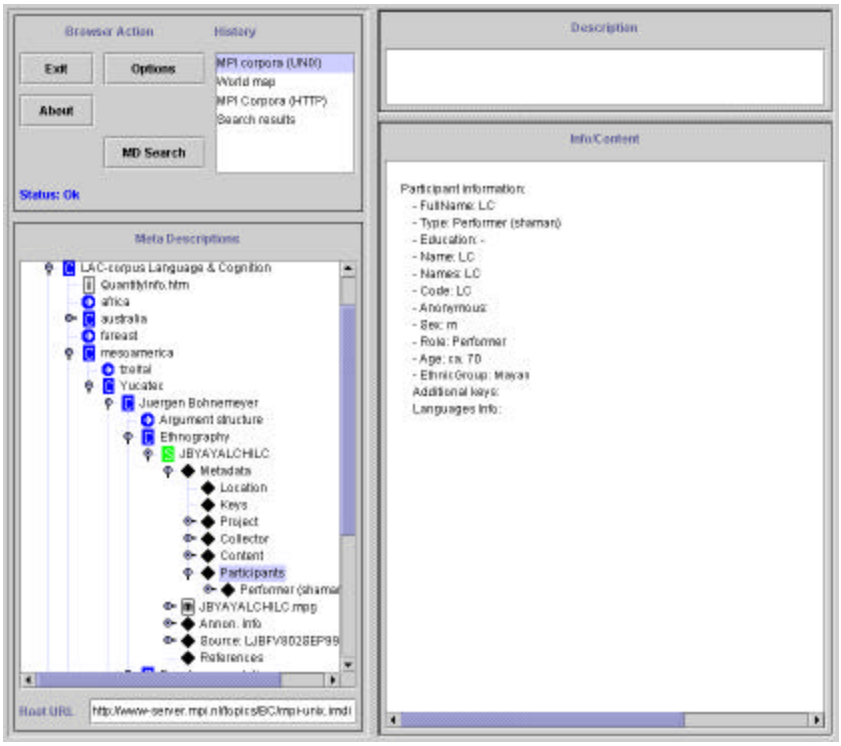


Figure 2 shows a screenshot of the IMDI browser.

The IMDI BCBrowser is the central tool for exploiting the IMDI metadata infrastructure. It allows navigation in the domain of linked IMDI metadata descriptions by clicking on corpus links. The browser keeps track of its position in the browsable corpus structure and displays the metadata and human readable descriptions associated with the sub-corpus in focus. It allows the user to set bookmarks so that easy navigation is facilitated.

The browser is also capable of displaying HTML formatted or PDF files that are often provided as extra documentation for corpora. It is possible to link in such HTML pages or PDF files in the corpus tree. From the HTML pages there may be links back to metadata descriptions making it possible to mix classical HTML browsing with browsing the IMDI corpus universe.

An interesting application of this is a world map that was created as a portal of the MPI corpora. This world map is viewable as an HTML file but has, at the appropriate places, links to metadata descriptions for corpora that correspond to those locations. We are presently engaged in trying to incorporate a professional geographic information system since the HTML world map is not completely satisfactory. The worldmap is just one other alternative view on a corpus since it is organized according to geographical principles.

One of the very important functions of the browser is that It offers the user a set of appropriate tools for further analysing resources once they have been located and it allows for operation in a distributed scenario where all resources are indicated by URLs. Each user or group of users can create a configuration file containing information on how to immediately start a tool and pass over the necessary parameters to start the tool with the discovered resource(s). The browser offers a selection from which the user can choose.

The search tool is the most recent IMDI development. It allows the user to specify a query for sessions whose metadata complies with the specified constraints. The UI offers the user an easy way to specify a query compliant with the IMDI element set, the elements value constraints and CVs used.

Results are presented in the form of URLs for the session metadata description files that comply with the query. The user may make these sessions visible in the IMDI-BCBrowser for further inspection or a special corpus label can be created containing all these sessions that can be saved for future reference and processing. The search tool can, of course, be started from the IMDI-BCBrowser. The search tool has to be extended to support the distributed architecture underlying the IMDI concept and it has to be checked as to how it can support harvesting of other metadata repositories by, for instance, using the OAI protocol. Currently, two teams are working on an improved search tool working in fully distributed scenarios.

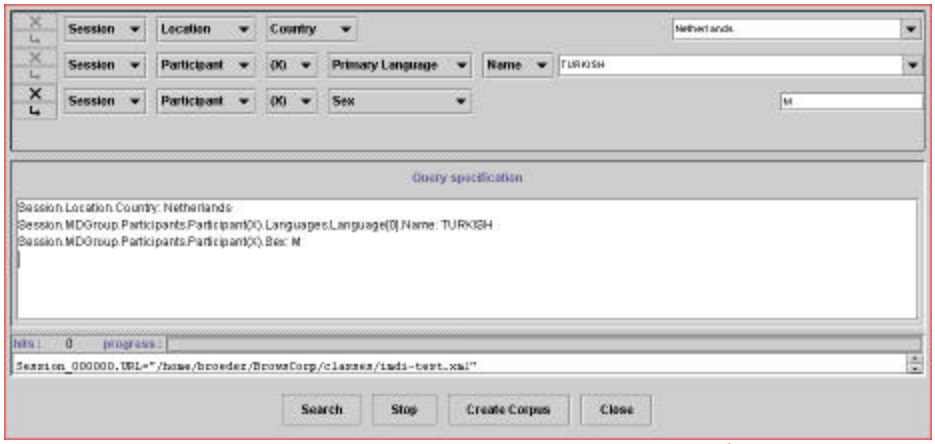


Figure 3 shows a screenshot from the search component.

The IMDI team also created a number of scripts which allow users to efficiently work with IMDI type of metadata descriptions. One such tool is provided to add or change element values in a whole range of MD descriptions by one command. Another allows the user to create metadata descriptions from spreadsheet documents, although this has proved problematic. Spreadsheet entries are not guided by constraints or controlled vocabularies

therefore conformity has to be checked very carefully. There are a few other minor scripts which will hopefully become obsolete when the editor or browser have been extended.

5. IMDI Corpora

At present we have available as IMDI tagged corpora:

- the MPI corpora of the "Acquisition" and "Language and Cognition" group which contains more than 2 TB of media data and more than 7000 multimedia sessions;
- a large second learner language acquisition corpus also containing audio recordings;
- the data of the DOBES project about endangered languages where also audio and video recordings form the basis;
- the data of the CGN (Spoken Dutch Corpus) project.

Furthermore we have been experimenting with converting parts of existing corpora to see if the IMDI set is applicable. These tests range from the well-known "Childes" corpora [11] to language engineering corpora as "TIMIT" [12] and "SmartKom". An interesting project was also the construction of a distributed corpus with examples of (parts of) corpora of six different European institutes. This was demonstrated as a first distributed IMDI scenario during the official opening ceremony of the "European Year of the Language" in Lund in 2001.

6. Future Developments

As a preliminary solution and part of the IMDI showcase, the MPI serves as a focal point maintaining the IMDI web portal as a starting point for the IMDI universe and maintaining the IMDI metadata Schema and CV definitions. However, the MPI does not have ambitions to perform this task in the long run. Such hosting activities are better performed by organisations such as BAS [13], ELRA and LDC. The maintenance of the IMDI set and the related tools by the MPI has been secured for many years by using them in different long-term projects. Besides these organisational problems, there is also a need for further tool development, such as a tool offering users a graphical interface for creating alternative "personal" corpus trees. Maintenance tools are required that allow users to copy parts of corpus trees to other portable media such as CDROM and DVD. In this way they can work under field conditions or make personal archive copies.

A major revision of the IMDI metadata set is expected to occur in 2002, therefore comments on how to improve it are welcome. According to the most recent discussions, it can be concluded that the MD set in general is very mature and stable with the exception of a very few elements such as "Anonymous". But the elements and vocabularies which were defined to describe the content of the resources have to be modified after a year of experience. Here, the elements define the dimensions of descriptions and the vocabularies the values along these dimensions. Although the current definitions are based on linguistic experience, it is obvious that not all contents can be described equally well with them.

Currently, the IMDI definitions are specified with the help of an XML Schema, i.e. the relations between concepts are implicitly defined in the structured IMDI set. To open up the way to the Semantic Web these implicit relations will be explicitly defined with the help of RDF [14]. All RDF Schemas will be put into open RDF repositories so that they can be re-used. It has to be checked whether it will be possible to make use of already existing descriptions within the IMDI set.

7. References

[1] Broeder, D.G., Brugman, H., Russel, A., and Wittenburg, P., (2000), A Browsable Corpus: accessing linguistic resources the easy way. In *Proceedings LREC 2000 Workshop*, Athens.

[2] ISLE/IMDI: http://www.mpi.nl/ISLE &
http://www.mpi.nl/world/ISLE/documents/papers/white_paper_11.pdf &
http://www.mpi.nl/ISLE/documents/draft/ISLE_MetaData_2.5.pdf

[3] DOBES: http://www.mpi.nl/DOBES

[4] CGN: http://www.now.nl/gw/introductie

[5] SmartKom: http://smartkom.dfki.de

[6] DC: http://www.dublincore.org/

[7] OAI: http://www.openarchives.org/

[8] MPEG7: http://mpeg.telecomitalialab.com/standards/mpeg-7/mpeg-7.htm

[9] OLAC: http://www.language-archives.org/OLAC/

[10] Ethnologue Language List: http://www.ethnologue.com

[11] ChilDes: http://childes.psy.cmu.edu

[12] TIMIT: http://www.ldc.upenn.edu/Catalog/LD93S1.html

[13] BAS: http://www.phonetik.uni-muenchen.de/Bas/BasHomeen.html

[14] RDF: http://www.w3.org/RDF & http://www.w3.org/sw

The FORM Gesture Annotation System

Craig Martell[*†], **Chris Osborn**[*†], **Jesse Friedman**[*], **Paul Howard**[*]

[*]Linguistic Data Consortium
University of Pennsylvania
3615 Market Street, Suite 200
Philadelphia, PA 19104
{cmartell, cosborn, jessef2, pch}@unagi.cis.upenn.edu

[†]Department of Computer and Information Sciences
University of Pennsylvania
200 S. 33rd St
Philadelphia, PA 19104

Abstract

The Friedman Osborn Martell (FORM) system for annotating gestures has been created for the purpose of producing a corpus of speech and its corresponding gestures. The corpus is open source and will be available to all researchers who wish to use it in their work. FORM attempts to capture the kinematics of gesture using *quasi*-geometric descriptions of the locations/shapes and movements of the arms and hands. Currently, we have a pilot corpus of 22 minutes of gesture-annotated video of Brian MacWhinney teaching a Research Methods course at Carnegie Mellon University. There are plans to extend the corpus to include not only speech transcriptions and syntactic information, but also body-movement and intonational information as well. We are currently gathering other data of various types and in various settings to add to the corpus. All of these data will be be published under the TalkBank project (http://www.talkbank.org).

1. Introduction

In "An Agenda for Gesture Studies" (Kendon, 1996), Adam Kendon outlines a long-term research agenda for a better understanding of gesture and its relationship to the communicative process. A major aspect of that agenda is the development of what Kendon calls the "Kinetics of Gesture":

> Such a programme of work could be linked to, and would contribute importantly, to research on what might be called the 'kinetics' of gesture (in parallel to 'phonetics'). We really have little explicit knowledge about how gestures are organized as physical actions.... An important part of the 'kinetics' research should include a study of just how gesture phrases are organized in relation to speech phrases.

The FORM project began, in large part, as a response to this challenge[1]. FORM is an annotation scheme designed both to describe the kinematic information in a gesture, as well as to be extensible in order to add speech and other conversational information.

Our plan, then, is to build an extensible corpus of annotated videos in order to allow for general research on the relationship among the many different aspects of conversational interaction. Additionally, further tools and algorithms to add these annotations and evaluate inter-annotator agreement will be developed. The end result of this work will be a corpus of annotated conversational interaction, which can be:

- extended to include new types of information concerning the same conversations; as new tag-sets and coding schemes are developed—discourse-structure or facial-expression, for example—new annotations could easily be added;
- used to test scientific hypotheses concerning the relationship of the paralinguistic aspects of communication to speech and to meaning;
- used to develop statistical algorithms to automatically analyze and generate these paralinguistic aspects of communication (e.g., for Human-Computer Interface research).

2. FORM

2.1. The Annotation Scheme

FORM is designed as a series of tracks representing different aspects of the gestural space. Generally, each independently moved part of the body has two tracks, one track for Location/Shape/Orientation, and one for Movement. When a part of the body is held without movement, a Location object describes its position and spans the amount of time the position is held. When a part of the body is in motion, Location objects with no time period are placed at the beginning and end of the movement to show where the gesture began and ended. Location objects spanning no period of time are also used to indicate the Location information at critical points in certain complex gestures. See Figure 1 for a snapshot of FORM implimented using the Anvil tool (Kipp, 2001).

[1]The authors wish to sincerely thank Adam Kendon for his input on the FORM project. He has provided not only suggestions as to the direction of the project, but also his unpublished work on a kinematically-based gesture annotation scheme was the FORM project's starting point (Kendon, 2000).

An object in a movement track spans the time period in which the body part in question is in motion. It is often the case that one part of the body will remain static while others move. For example, a single hand shape may be held throughout a gesture in which the upper arm moves. FORM's multi-track system allows such disparate parts of single gestures to be recorded separately and efficiently and to be viewed easily once recorded. Once all tracks are filled with the appropriate information, it is easy to see the structure of a gesture broken down into its anatomical components.

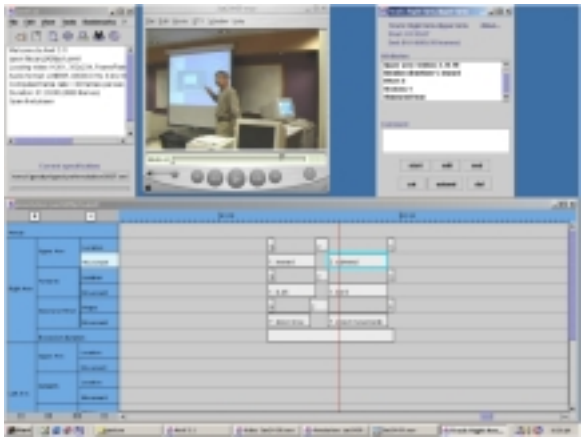


Figure 1: FORM annotation of Jan24-09.mov, using Anvil as the annotation tool

At the highest level of FORM are groups. Groups can contain subgroups. Within each group or subgroup are tracks. Each track contains a list of attributes concerning a particular part of the arm or body. At the lowest level (under each attribute), all possible values are listed. The structure, then, is as follow:

Group

Subgroup

Track

ATTRIBUTE

Value

The following descriptions will follow this structure. The groups described are Right/Left Arm, Gesture Obscured, Excursion Duration, and Two-Handed Gesture. Not described are Head and Torso Movement/Location. These will be implimented in a later version of FORM.

Right/Left Arm

Upper Arm (from the shoulder to the elbow).

Location

UPPER ARM LIFT (from side of the body)

no lift
0-45
approx. 45
45-90
approx. 90
90-135
approx. 135
135-180
approx. 180

RELATIVE ELBOW POSITION: The upper arm lift attribute defines a circle on which the elbow can lie. The relative elbow position attribute indicates where on that circle the elbow lies. Combined, these two attributes provide full information about the location of the elbow and reveal total location information (in relation to the shoulder) of the upper arm.

extremely inward
inward
front
front-outward
outward (in frontal plane)
behind
far behind

The next three attributes individually indicate the direction in which the biceps muscle is pointed in one spatial dimension. Taken together, these three attributes reveal the orientation of the upper arm.

BICEPS: INWARD/OUTWARD

none
inward
outward

BICEPS: UPWARD/DOWNWARD

none
upward
downward

BICEPS: FORWARD/BACKWARD

none
forward
backward

OBSCURED: This is an binary attribute which allows the annotator to indicate if the attributes and values chosen were "guesses" necessitated by visual occlusion. This attribute is present in each of FORM's tracks.

Movement

The next three attributes individually indicate the direction of elbow movement in one spatial direction. When diagonal movement occurs, a non-none (i.e.not *none*) value for more than one of the attributes is chosen. Each attribute has combination values so repeated or back-and-forth motions can be annotated as such.

LINEAR MOVEMENT (HORIZONTAL PLANE: Indicates the direction(s) of inward or outward elbow movement.

none
inward
outward
inward-outward
outward-inward

LINEAR MOVEMENT (MEDIAN PLANE): Indicates the direction(s) of upward or downward elbow movement.

none
up
down
up-down
down-up

LINEAR MOVEMENT (FRONTAL PLANE): Indicates the direction(s) of elbow movement towards or away from the body.

none
towards
away
towards-away
away-towards

UPPER ARM ROTATION: The degree of change of bicep direction. Ranges are exclusive. Direction of change is not included, as it can be inferred from the information in the Location track.

0-45
approx. 45
45-90
approx. 90
90-135
approx. 135
135-180
approx. 180
greater than 180

ARC-LIKE MOVEMENT: This boolean attribute indicates whether or not the elbow movement was arc-like. When checked, Location objects will co-occur to note the location of the elbow at the beginning, apex, and end of the movement.

CIRCULAR MOVEMENT: A non-none value indicates that elbow movement is circular in shape and notes the plane in which the movement is performed as well as its direction (clockwise or counter-clockwise). As was the case for arc-like movements, the Location track will be simultaneously utilized, in this case noting the location of the elbow at the start and halfway mark of the circle. This convention allows the size of the circle to be inferred.

parallel to horizontal plane (c=clockwise)
parallel to horizontal plane (cc=counter-clockwise)
parallel to median plane (c)
parallel to median plane (cc)
parallel to frontal plane (c)
parallel to frontal plane (cc)

EFFORT: Indicates the effort of the movement on a 1 to 5 scale.

1
2
3
4
5

STROKES: Indicates the number of strokes of a movement.

1 . . . 20
More than 20
Indeterminate

OBSCURED

Forearm: the part of the arm extending from the from elbow to wrist)

Location

ELBOW FLEXION: The angle made by the bend in the elbow.

0-45
approx. 45
45-90
approx. 90
90-135
approx. 135
135-180
straight

FOREARM ORIENTATION: Describes the orientation of the forearm if the upper arm were to be by the side and the elbow flexed at 90 degrees.

supine
supine/neutral
neutral
neutral/prone
prone
prone/inverse
inverse

OBSCURED

Movement

ELBOW FLEXION CHANGE: The amount of change in elbow flexion measured in degrees. Direction of flexion change is not indicated, as it can be inferred from information in the Location track.

0-45
approx. 45
45-90
approx. 90
90-135
approx. 135
135-180
approx. 180

FOREARM ROTATION: Direction of change of forearm orientation. Amount of change is not indicated, as it can be inferred from information in the Location track.

none
inward
outward
inward-outward
outward-inward

EFFORT

STROKES

OBSCURED

Hand and Wrist

Shape: Information about the static shape of the hand and orientation of the wrist.

The next two attributes give values to describe the shape of the hand. The values are represented in a catalog of hand-shapes (Figure 2), which is organized as a two-dimensional matrix. This method is employed because the complexity of the hand would make purely physicalistic descriptions too unwieldy.

HAND-SHAPE GROUP: Indicates the group (organized by number of extended fingers with 0 representing fist and 6 referring to miscellaneous shapes) in the hand shape catalog.

0
1
2
3
4
5
6

HAND-SHAPE LETTER: Indicates the appropriate hand-shape within the selected group.

A
B
C
D
E
F
G
H
I
J
K
L
M

TENSION: Describes the amount of tension apparent in the performer's hand. An average amount of tension corresponds to the "slightly tense" variable.

relaxed
slightly tense
very tense

WRIST BEND: UP AND DOWN: How far the wrist is bent towards the upper side or under side of the forearm.

up
up-neutral
neutral
down-neutral
down

Figure 2: Catalog of Handshapes. Based on the HamNoSys catalog (http://www.sign-lang.uni-hamburg.de/Projects/HamNoSys.html)

WRIST BEND: SIDE TO SIDE: How the wrist is bent towards the thumb or little finger.

towards thumb
neutral
towards little finger
extremely towards little finger

PART OF BODY TOUCHED:

none
top of head
eye (same)
eye (opposite)
ear (same)
ear (opposite)
temple (same)
temple (opposite)
nose
cheek (same)
cheek (opposite)
chin
neck (same side)
neck (center)
neck (opposite side)
chest
groin

OBSCURED

Movement

HAND MOVEMENT: Describes type of hand movement (if any). The A joint refers to the knuckle furthest from the fingertip and the B joint refers to the first joint above the A joint. Information about the C joint (the joint closest to the fingertip) is not recorded because C joint movement is usually dependent upon movement of the B joint. The numbering scheme of the first three variables is explained in the Finger Coordination attribute.

none
1) A joint movement
2) B joint movement
3) A and B joint movement
wrist circular
thumb rubbing index finger
thumb rubbing multiple fingers
direct movement between two shapes

WRIST UP-DOWN MOVEMENT: Describes the up-down movement (to the underside or upper side of the arm) of the wrist.

up
down
up-down
down-up

WRIST SIDE-TO-SIDE MOVEMENT

towards little finger
towards thumb
towards little finger-towards thumb
towards thumb-towards little finger

FINGER COORDINATION: Describes the motion of the fingers in relationship to each other. A non-none value is only applicable if one of the choices labeled 1, 2, or 3 was selected from the Hand movement attribute.

parallel movement without thumb
random movement, without thumb
parallel movement, with thumb
random movement, with thumb
movement in sequence

EFFORT

STROKES

OBSCURED

Excursion Duration: Marks the length of the excursion of the arm from a resting position to another resting position. Since there is ambiguity about what constitutes a single gesture, this convention for grouping was adopted.

Gesture Obscured: Similar to above except this attribute refers to the entire gesture duration, rather than just one track.

Two-handed Gestures

RIGHT-HAND CONTACT

none
thumb
index finger
middle finger
ring finger
little finger
palm
back of hand
more than one digit
holding

LEFT-HAND CONTACT: The list of values is identical to that of the Right-hand Contact attribute.

The following seven attributes are all boolean-valued.

MOVING IN PARALLEL

MOVING APART

MOVING TOWARDS

MOVING AROUND ONE ANOTHER

MOVING IN ALTERNATION

CROSSED

OBSCURED

2.2. Ambiguities/Imprecisions in FORM

There are two known ambiguities/imprecisions in the current version of the FORM system.

The first concerns the ***Upper Arm***:*Location* attributes that specify biceps direction. While anatomically it seems more accurate to describe the upper arm rotation by degrees of rotation rather than by using the direction of the biceps in free space, a problem arises when defining the neutral position of the arm rotation. For example, we could define *normal* as the position when the arm is held at the side with palm facing towards the body and the elbow flexed to make a 90-degree angle with the upper arm. If one then lifts the upper arm to the side so it is at 90 degrees with the body and still in the frontal plane, the upper arm has not rotated at all. Let's call this position 1. If, however, one returns to the starting position, raises the upper arm forward so it is at 90 degrees with the body but parallel to the median plane, and then moves the upper arm 90 degrees to the side so that it is in the frontal plane again, it can be seen that this position is also reached without rotating the upper arm. Let's call this position 2. It is clear that position 1 is not the same as position 2, but both were reached by keeping the upper-arm in the *normal* position.

To solve this issue we could define a normal that is rotated 45 degrees when the Upper arm lift is at what we've deemed "approx. 90" and Relative elbow position is "front-outward." This convention, however, is hard to conceptualize by annotators and thus we opted to use the direction of biceps in free space since it is more intuitive. The downside to this approach is that it allows for a large range of positions for each combination of values. Many positions could be called "forward-inward-upward," for example.

The second area of concern is in the ***Upper arm***:*Movement* track. This track describes the movement of the upper arm independent of the forearm, elbow flexion, and hand-and-wrist. This movement can be described either as a combination of linear movement in different planes or as arc-like movement (using Location points to denote points along the curve). Since the upper arm is only

able to move on a partial sphere with the shoulder as the center, it does not make sense anatomically to describe its movement as linear. However, since most movements are small enough not to appear as distinct arcs, linear values sufficiently approximate the movement.

3. The Current FORM Corpus[2]

3.1. Pilot Corpus

We currently have a pilot corpus of about 22 minutes of Brian MacWhinney teaching a Research Methods course at Carnegie Mellon University. These data were chosen since they were freely available via the the TalkBank project (http://www.talkbank.org). They have been very useful for the pilot phase of the project as people often gesture in a clear and exaggerated fashion while teaching. See (Martell, 2002) for a further description of the data format (Annotation Graphs (Bird and Liberman, 1999)), as well as examples of the video and of tool currently being used (Anvil (Kipp, 2001)).

3.2. Annotation Complexity

An experienced annotator can create approximately 3 seconds of annotation per hour. He/she can annotate at most for 6 hours per day, generating 18 seconds/day. Accordingly, it will take an experienced annotator 5 work days to annotate a 90-second video of conversational interaction.

Generating only 90 seconds of annotation per work week makes such an annotation project seem a daunting task. However, the amount of information contained in conversational gesturing is substantial—on the order of 3500 distinct ATTRIBUTE:Value pairs per minute. This underscores the potential value of such a corpus, viz. there is seemingly much more information in 90 seconds of communicative interaction than we are currently capturing by only transcribing speech.

3.3. Preliminary Inter-Annotator Agreement Results

Preliminary results from FORM show that with sufficient training, agreement among the annotators can be very high. Table 2 shows preliminary interannotator agreement results from a FORM pilot study.[3] The results are for two trained annotators for approximately 1.5 minutes of Jan24-09.mov, the video from Figure 1. For this clip, the two annotators agreed that there were at least these 4 gesture excursions. One annotator found 2 additional excursions. Precision refers to the decimal precision of the time stamps given for the beginning and end of gestural components. The *SAME* value means that all time-stamps were given the same value. This was done in order to judge agreement with having to judge the exact beginning and end of an excursion factored out. *Exact* vs. *No-Value* percentage refers to whether both the attributes and values matched exactly or whether just the attributes matched exactly. This distinction is included because a gesture excursion is defined as all movement between two rest positions of the arms and hands. For an excursion, the annotators have to judge both which parts of the arms and hands are salient to the movement (e.g., upper-arm lift and rotation, as well as forearm change in orientation and hand/wrist position) as well as what values to assign (e.g., the upper-arm lifted 15-degrees and rotated 45-degrees). So, the *No-Value%* column captures the degree to which the annotators agree just on the structure of the movement, while *Exact%* measures agreement on both structure and values.

The degree to which inter-annotator agreement varies among these gestures might suggest difficulty in reaching consensus. However, the results on *intra*-annotator agreement studies demonstrate that a single annotator shows similar variance when doing the same video-clip at different times. Table 3 gives the intra-annotator results for one annotator annotating the first 2 gesture excursions of Jan24-09.mov.

Gesture Excursion	Precision	Exact%	No-Value%
1	2	3.41	4.35
	1	10.07	12.8
	0	29.44	41.38
	SAME	56.92	86.15
2	2	37.5	52.5
	1	60	77.5
	0	75.56	94.81
	SAME	73.24	95.77
3	2	0	0
	1	19.25	27.81
	0	62.5	86.11
	SAME	67.61	95.77
4	2	10.2	12.06
	1	25.68	31.72
	0	57.77	77.67
	SAME	68.29	95.12

Table 1: Inter-Annotator Agreement on Jan24-09.mov

Gesture Excursion	Precision	Exact%	No-Value%
1	0	5.98	7.56
	1	20.52	25.21
	0	58.03	74.64
	SAME	85.52	96.55
2	2	0	0
	1	25.81	28.39
	0	89.06	95.31
	SAME	90.91	93.94

Table 2: Intra-Annotator Agreement on Jan24-09.mov

For both sets of data, the pattern is the same:

- the less precise the time-stamps, the better the results;
- *No-Value%* is significantly higher than *Exact%*.

It is also important to note that Gesture Excursion 1 is far more complex than Gesture Excursion 2. And, in both sim-

[2] Most of this section is taken from (Martell, 2002)

[3] Essentially, all the arcs for each annotator are thrown into a bag. Then all the bags are combined and the intersection is extracted. This intersection constitutes the overlap in annotation, i.e., where the annotators agreed. The percentage of the intersection to the whole is then calculated to get the scores presented.

ple and complex gestures, inter-annotator agreement is approaching intra-annotator agreement. Notice, also, that for Excursion 2, inner-annotator agreement is actually better than intra-annotator agreement for the first two rows. This is a result of the difficulty for even the same person over time to precisely pin down the beginning and end of a gesture excursion. Although the preliminary results are very encouraging, all of the above suggests that further research concerning training and how to judge similarity of gestures is necessary. Visual information may need very different similarity criteria.

4. Future Directions and Open Questions

As mentioned in the introduction, the goal of the FORM project is to create a corpus to be used for both scientific and technological research concerning gesture and its relationship to the rest of the communicative process. However, in order to build a corpus suitable for these goals, a number of issues have to be addressed.

Augmentation of the FORM corpus with other aspects of communication is necessary. Over the next 3 years, as we continue to build the FORM corpus, we will also be augmenting it with:

- Speech transcriptions and syntactic information;
- Body movement, in the form of head and torso information; and
- Intonation and pitch contour information.

Much research is needed to discover the best annotation schemes for each of these aspects, as well as to discover which algorithms are best for uncovering the correlations among them.

Better methods of annotation need to be developed. Although we belief it will prove necessary to continue to annotate at a fine-grained level of detail, it is currently too expensive to make using FORM practical. We intend to use the hand-annotated corpus, as it grows, to explore automatic or semi-automatic methods of annotation.

Visualization and Animation Tools which will "play back" an annotation are needed to allow an annotator to better judge the correctness of his/her annotation. Additionally, these visualization tools may be able to help ease the annotation process. If we are able to develop a close enough mapping between the animated character and the annotation scheme, we may be able to use a movable animated character as a means to input the data. Research is need to see if this will indeed speed up the process.

New Metrics for Inner-Annotator Agreement need to be explored. As mentioned in Section 3.3, above, our current numbers are based on the bag-of-arcs technique. However, as the scores there indicate, often annotators agree to a large degree on structure, but differ only on exact beginning or ending timestamp, or on the value of an attribute. Unfortunately, small differences in timestamp and value are judged incorrect to the same degree as large differences. Visual feedback, as just described, will allow us to discover whether small differences in coding actually have little difference visually. If this proves to be the case, then we will need to experiment with more geometrically-based measures of similarity, e.g., distance in n-dimensional space.

Statistical Experiments using FORM are already underway. If FORM is to be successful, it must be shown that our fine-grained analysis sufficiently captures the phenomena in question. To do this we are conducting two sets of experiments.

- We have annotated some of the corpus with Preparation-Stroke-Retraction information. Using standard training-set/test-set methods, we are building Preparation-Stroke-Retraction recognizer system. If the results of these experiments are sufficiently high, we will have demonstrated that FORM captures at least as much information as a more coarse-grained annotation scheme.
- However, only showing that FORM is a as good as coarse-grained annotation scheme is not sufficient justification for using FORM. Accordingly, we are also working on a Statistical Gesture Generation System (SGGS). Given some input set of sentences, the SGGS, if successful, will be able to output those sentences augmented with a FORM description of valid accompanying gestures. This, then, could be used with the above described annimation tool to automatically generate animated gesture excursions.

5. Conclusion

The FORM project has develped a geometrically-based gesture annotation scheme and a 22-minute pilot corpus of gesture-annotated video. Over the next few years, the corpus will be augmented and new tools and algorithms will be developed. The envisioned goal of the project is a large-scale corpus of multi-modal annotations suitable for both scientific and technological research concerning the relationships among different aspects of communicative interaction.

6. References

Steven Bird and Mark Liberman. 1999. A formal framework for linguistic annotation. Technical Report MS-CIS-99-01, Department of Computer and Information Sciences, University of Pennsylvania, Philadelphia, Pennsylvania. http://citeseer.nj.nec.com/article/bird99formal.html.

Adam Kendon. 1996. An agenda for gesture studies. *Semiotic Review of Books*, 7(3):8–12.

Adam Kendon. 2000. Suggestions for a descriptive notation for manual gestures. Unpublished.

Michael Kipp. 2001. Anvil - a generic annotation tool for multimodal dialogue. In *Proceedings of Eurospeech 2001*, pages 1367–1370.

Craig Martell. 2002. Form: An extensible, kinematically-based gesture annotation scheme. In *Proceeding of the International Conference on Language Resources and Evaluation.* European Language Resources Association. http://www.ldc.upenn.edu/Projects/FORM.

Sample Annoted Video Using Anvil and FORM

Craig Martell
Linguistic Data Consortium
University of Pennsylvania
3615 Market Street Suite 200
Philadelphia, PA 19104-2608, USA
cmartell@unagi.cis.upenn.edu
http://www.cis.upenn.edu/~cmartell

Video Sample:

http://www.ldc.upenn.edu/annotation/gesture/Jan24-05.mov

Coding Scheme:

This is a video of Brian MacWhinney lecturing, and we coded his gestures using or annotation scheme FORM. This was described in a separate submission for the workshop -- or you can see the same description at

http://www.ldc.upenn.edu/Projects/FORM

FORM is a kinematic annotation scheme which describes gestures by their physical movements. The idea is to later add speech, and other paralinguistic information, to the data set to better understand the relationship of gesture to speech.

Annotation File:

http://www.ldc.upenn.edu/annotation/gesture/Jan24-05pch.anvil

This is an XML file for use with Anvil, described below

Annotation Tool:

We currently use Michael Kipp's tool Anvil (http://www.dfki.de/~kipp/anvil/). To use this with FORM, you will need the specification file, which contains the entire coding scheme. This can be found at:

http://www.ldc.upenn.edu/annotation/gesture/gestureAnnotation0807.xml

Cross-Linguistic Studies of Multimodal Communication

P. Wittenburg, S. Kita, H. Brugman

Max-Planck-Institute for Psycholinguistics
Wundtlaan 1, 6525 XD Nijmegen, The Netherlands
peter.wittenburg@mpi.nl

Abstract

Gestures are culture specific forms of arm movements which are used in communication to transfer information to the listener, to guide the planning of the speech production process and to disambiguate the incoming speech. To understand the underlying mechanisms gestures have to be analyzed in cross-linguistic processes. Large projects are necessary covering speakers from various cultural background and many recordings. Such projects can only be successfully carried out, when suitable gesture encoding schemes, generic annotation schemes, powerful tools supporting the schemes and efficient methods for easy resource discovery and management are available. At the Max-Planck-Institute all aspects were tackled.

1. Introduction

The MPI for Psycholinguistics has a long history of research on the synchronization between different modalities in human communication. In the 1980s eyetracking signals and signals about pointing gestures produced important information about the mental processes responsible for speech production [1, 2]. Such signals were typically recorded in relation to spoken utterances. The equipment used was designed to make automatic fine grained temporal analysis possible. For gesture registration IR-light based methods were used. Mort recently, ultrasonic equipment was used for this purpose identifying the location of maximally 8 sources. This tradition is still continued in the baby labs where eye tracking is recorded to study, for example, the focus of childrens' attention during linguistic tasks. In recent years brain imaging methods (EEG, MEG, PET, MRI) have often been added to get online information about brain activities during speech production and perception task.

In the last few years, research using multimodality shifted towards observational methods in communicative situations of various sorts. Child-caretaker interaction is studied with the help of extensive video recordings to better understand how childrens' language learning is influenced by input and environmental factors. The use of various types of gestures (pointing, iconic and emblematic) is studied in different situations. The following studies should be mentioned in particular: (1) ethnography of pointing gestures; (2) gestural facilitation of speaking or understanding; (3) gestural expression of motion events; (4) speech dysfluencies and gestures; (5) influence of gestures on recipients' gaze movement; (5) hemispheric specialization of types of gestures [3, 4, 5, 6, 7, 8]. In addition, studies about sign language and their comparison to gestural patterns were carried out. The goal of these recordings is fundamental research about the relation between language and thought and the role of gesture in human communication. Since gestures are very much dependent on language and culture, most of the recordings are cross-linguistic, i.e. various countries and cultures are included.

Nowadays the study of multimodal communication based on video recordings is much easier. Information technology allows science to work with digitized video greatly facilitating the analysis work. For the last two years, all recordings at the MPI have been digitized. yielding an online multimedia corpus consisting of more than 7000 sessions (units of linguistic analysis). Gesture studies form a substantial part of these recordings. Powerful corpus management with the help of metadata descriptions and multimodal annotation tools were developed at the institute to enable the type of research explained. Annotations are stored in well-documented formats well adapted to capturing the complexity of the annotation which are typical of multimodal studies.

2. Multimodality Research

Multi-modal records allow us not only to approach old research problems in new ways, but also open up entirely new avenues of research. An old issue, for example, is just how 'modular' language processing is, that is to what extent non-linguistic processes can intervene in the course of linguistic processing. This can be studied by looking at the interaction between two entirely different behaviour streams, gesture and speech. A large multi-media corpus of natural dialogue shows, for example, that when speakers self-edit speech, gesture inhibition actually occurs earlier, suggesting interaction between the speech and gesture execution systems. Similarly, in the comprehension process it can be shown that gesture content is incorporated into the immediate 'message'. Eye-tracking shows that speakers can manipulate the likelihood of this by looking at their own gestures, which are then more often fixated by listeners. More fundamentally, we can look at the role of the two cerebral hemispheres in the production of the two behaviour streams, speech and gesture. Careful studies of the gestures of split-brain patients show that gesture production is largely driven from the right hemisphere, while language of course is normally processed in the left.

In addition to contributing to such long-standing theoretical issues, annotated multimedia records also make possible entirely new lines of research. For example, we have been interested in whether the semantic character of a specific language leads to a special construal of a scene to be described. The study of gesture during online production shows that the way a language 'packages' information has a demonstrable effect on the depiction of

a scene in gestures. Turkish for example packages movement with direction in a single clause but puts manner of motion into a separate adverbial clause ('The ball descended, rolling') – while English allows manner and direction to occur in the same simple clause ('The ball rolled down'). Turkish speakers tend to produce separate gestures for direction and manner, while English speakers tend to fuse them. In a similar way, we have been able to study spatial thinking as it occurs in non-spatial domains, by examining the gestures of speakers talking about e.g. kinship relations.

Sign languages are another domain which has been opened up by multi-media technology. Sign languages are fully-expressive languages which utilize not only the hands, but also the face, gaze and even body-posture to construct complex utterances with phonology, morphology, syntax and 'prosody'. These different 'articulators' express different distinctions in overlapping time windows, where the offset can indicate e.g. the scope of a question. Even the simplest description of a signed utterance therefore requires a multi-tiered annotation of a video-record, and the development of such annotation tools make possible systematic databases for sign language research for the first time. Fascinating questions can now be pursued about effects of modality on language – for example does the spatial nature of the visual-gestural channel have profound effects on the nature of sign languages, andgive sign languages an underlying commonality? Most deaf signers are exposed to the gestural systems of the surrounding spoken language, and we can also ask to what extent these gestural systems are recruited into the sign language. Preliminary results from the study of a sign language in the process of standardization (Nicaraguan sign language) suggests that there is such an interaction.

These examples should serve to indicate just what a revolution in our understanding of language and its relation to other aspects of cognition is being made possible by the new technologies. There are also fundamental advantages to archiving multi-media records for all branches of the language sciences. For example, studies of the acquisition of language are hugely enriched by having available the very scene available to the infant language user – we now know for example that unexpressed arguments (e.g. subjects and objects) in Inuit care-takers' speech are often recoverable by the child just because they are most likely in the child's field of view at the moment of utterance. Similarly, records of dying or endangered languages are greatly enhanced by having visual information correlate with the language use. In all these cases, richly annotated multi-media records make possible the extraction of systematic information about the correlation of linguistic and non-linguistic events.

3. Gesture Encoding Schemes

General

This variety of studies all based on observational methods (i.e. audio and video, sometimes also gaze) required many different gesture encoding schemes on the different linguistic levels, efficient procedures and powerful tools. Since our researchers are involved in international projects broad agreements on the methods for encoding multimodal behavior are very important. Yet for international standards it seems to be too early, the discipline is too young, although it would facilitate integrating and comparing the data of all the scholarly work.

Most of the studies require careful encoding of the articulator movements[1] and their global timing pattern. Naturally, we are faced with similar problems to those for identifying the articulator movements in the case of speech production. The articulator movements form a continuum, are overlapping and have tolerances dependent on the situation. Therefore, it is not only difficult to make proper time segmentation, but also to classify them.

For gestures which are movements of the arms and its parts accompanying verbal communication acts, it is sufficient to annotate their type and meaning in addition to the articulators. The type of a gesture is a taxonomic classification of its principle purpose and role in communication. It is widely accepted to separate between pointing, iconic and emblematic gestures. Pointing gestures refer to a spatial point or a movement. They appear either as isolated gestures where the meaning is obvious to the listener or mostly in overlap with verbal utterances where the gestures are much more simple to generate and interpret than verbal descriptions. Their meaning is easy to describe by the object they refer to and their intrinsic purpose. Also iconic gestures appear spontaneously as co-speech activities while emblematic gestures stand alone. Iconic gestures have a culturally bound meaning since they are widely accepted within an area.

Gestures often correlate with emotional state, are used to facilitate the planning of speech production and to facilitate speech perception due to their disambiguation capability. Emotional state can be described, although there are no clear conventions yet.

Articulators in Gestures

The basis of all scientific work when studying gestures is an encoding scheme for the articulator movements. It was soon perceived that an exhaustive gesture encoding including all relevant characteristics would be ideal but impossible (except for small segments). On the other hand the recordings were perceived as so valuable that re-usage for various research questions was anticipated. To cope with this contradiction it was realised that only an iterative encoding approach would suffice where the needs of primary research projects do not hinder the addition of gesture encodings dedicated to completely different research interests. To support research, the underlying

[1] For gestures we have as articulators the arms and its parts up to the fingers. Characteristic movements of the head and the eyes in communicative situations are not treated as part of the gesture although they have similar purposes.

scheme should be exhaustive to define a grid allowing easy computational comparison. Therefore, for a number of recordings focused on in the Institute's gesture project, a thorough study was carried out to attain a general gesture encoding scheme that would allow comparative analysis to be made easily.

Based on Kendon's work a more accurate scheme was developed by v. Gijn, vd Hulst and Kita [9] to separate various phases in a gesture. A *MovementUnit* therefore can exist of several *MovementPhrases*. Basically, each of these can be seen as a sequence of a *Preparation* phase, an *ExpressivePhase* and a *Retraction* phase. An *ExpressivePhase* which covers the meaningful nucleus of a gesture is either an *IndependentHold* or a sequence of a *DependentHold*, a *Stroke*, and another *DependentHold*.

*MovementUnit = MovementPhrase**
MovementPhrase = (Preparation) => ExpressivePhase => (Retraction)
ExpressivePhase = IndependentHold
ExpressivePhase = (DependentHold) => Stroke => (DependentHold)
Preparation = (LiberatingMovement) => LocationPreparation >> HandInternalPreparation
Retraction (if subsequent movement) = PartialRetraction

*= consists of, * one or several, => discrete transition, () optional, >> normally blended out, occasionally discrete transition*

The authors developed a set of descriptive criteria to identify the phases and their usefulness was shown in several studies which were successfully annotated by student assistants.

v. Gijn, vd Hulst and Kita also developed an encoding scheme to describe mainly the articulator movements in the *ExpressivePhase* [10]. It is this phase where annotators are confronted with all the about 60 degrees of freedom and where not only the location and shape has to be described but also for example changes in motion and direction. The following aspects are described: *PathMovementShape (straight, circle, round, iconic, 7-form, ?-form, x-form, +-form, z-form), PathMovement Direction ([up|down], [front|back], [ipsilateral| contralateral]), HandOrien-tationChange ([supination| pronation], rotation, [flexion | extension], nodding, [ulnar flexion|radial flexion], lateral flexion), HandShape Change ([opening | closing], [abduction |adduction], [hinging |dehinging], [clawing |declawing], wiggling, opening wave, closing wave, rubbing, cutting), HandOrientation ([up| down], [front | back], [ipsilateral| contralateral]), and HandShape*. For the latter basically the HamNoSys scheme was re-used.

To support the various gesture related research activities simple encoding schemes are most often derived from this exhaustive scheme. The reference back to the unified exhaustive scheme together with the online availability of the annotated multimedia document allows easy re-usage and an enhancement of the annotations. This can either be corrections of the existing or the addition of new tiers.

When encoding gestures it is of great importance to understand the exact time relationships with the verbal utterances. This is not part of the gesture annotation scheme, but the annotation structure scheme has to provide adequate mechanisms.

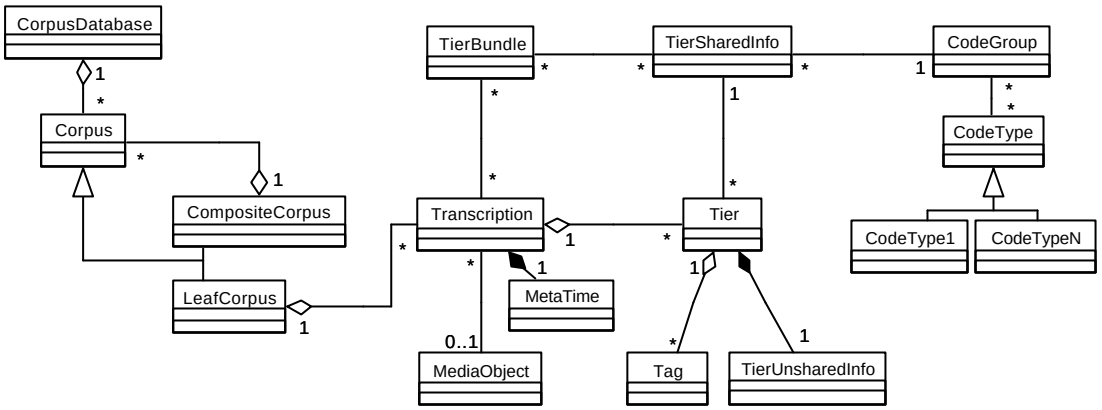
4. Annotation Structures

While the encoding scheme describes how to encode the linguistic phenomena (a close handshape in gestures is encoded as "*close*"), the annotation structure scheme describes the expressive power in structural respects. It has to provide mechanisms for all possible structural phenomena. From our long experience with gesture and sign language studies we know that the annotations can become very complex. There are projects which try to solve this complexity by merging the annotations associated with different linguistic levels into one tier. This method, which is known especially from traditional annotation schemes such as CHAT [11], is also used in new projects. The resulting annotation includes many relations implicitly, i.e. it is the tool which has to include all the knowledge. At the MPI this method was not seen as useful for the future. Different linguistic levels should be separated and all relations such as interruptions, parallelism, semantic correlation should be made explicit. This is the only way to easily modify the coding later.

In many cases different linguistic interpretations of a gesture are possible. The annotation scheme has to take this into account. Essentially, we follow the indicated way: add another tier which can be used by a new annotator. If only adaptations of the existing annotations are intended, a copy action may be useful for bootstrapping the tier.

The structural phenomena which can occur in annotations are described in detail in [12]. We can summarize the main points:

- The number of tiers can become comparatively high and cannot be seen in advance. It will increase due to various annotators and due to new research goals which require additional information.
- There are all kinds of temporal relations between gesture components and especially between annotations associated with different streams like gestures, speech, facial expression, gaze and others. The complexity makes it necessary to link annotations to periods of time and not to encode overlap and other phenomena in the annotations as older formats require.
- In some occasions spatial relations have to be encoded. They can be encoded as other annotations, i.e. individual or group of coordinate pairs can be linked to time periods.
- In many types of annotations hierarchical relationships have to be included to express linguistic phenomena. These can be token or type oriented. Type specific dependencies are defined at the level of



tier type definitions. Token specific dependencies occur randomly and are defined per linguistic unit.

- Cross-references are very relevant in many cases of linguistic annotation. They describe certain relations which the user wants to draw between two different linguistic units which can be on the same tier or on a completely different one. Comments on some annotation can be interpreted as such cross-reference.

5. Abstract Corpus Model

To design the Abstract Corpus Model informal use-case driven method was chosen. In addition a number of existing and well-known annotation formats were analyzed and discussions with linguists about their requirements were carried out. The resulting model defined in UML is more of an operational model than a mere data model.

ACM is realized in first instance as a set of abstract classes that implement common behavior. These abstract classes each have concrete subclasses, one for each of the annotation file formats that ACM currently supports (CHAT, Shoebox [13], relational database [14], Tipster [15], several varieties of XML).

The method calls from ACM's interfaces can be used by a range of annotation related tools. The interfaces are uniform to the tools although the actual objects that implement those interfaces may be instantiated from differently formatted files or even from a relational database. For example, the tools are not aware whether they work on a CHAT file or on a set of database records.

Most ACM objects are implemented as remote objects using Java's RMI facilities (Remote Method Invocation). This means that these objects can exist on a central annotation server while the annotation related tools that use their services run on local clients on the network. Method calls to a set of remote interfaces, with arguments and return value, offer a natural way to organize protocols for an annotation server. This type of support for remote objects is efficient since only data that is asked for is sent over the network, i.e. a tier name instead of a complete tier or annotation document. It also forms the basis for a collaborative annotation environment since remote objects can be simultaneously accessed by multiple users. For a class diagram of the first generation of the ACM see figure 1.

It is not the intention of this paper to discuss the part of the class diagram depicted in figure 1 in detail. For this we refer to [16]. But an example can demonstrate how to read it. In this version of ACM, *Tags* have begin and end times that can be specified or unspecified. To make this possible the order of all unaligned *Tags* (i.e. tags which have no specified time marks yet) in a *Transcription* has to be stored explicitly. The object responsible for this is called *MetaTime* and is associated with *Transcription*.

ACM Revision

Recently, the ACM was revised considerably to include new features. Merging the more elaborated BC (BrowsableCorpus) [17] and EUDICO models of corpora required the introduction of a Session class in ACM. The direct association between Transcriptions and MediaObjects is now administered by a Session object. The composite Corpus structure in ACM is maintained, but as an alternative to BC Corpus hierarchies. There was also a need to introduce Metadata, MetadataContainer and LanguageResource interfaces into ACM as a way to merge in behavior that is needed for BC.

In the first version of ACM, new objects were usually instantiated by their direct ancestors in the corpus tree e.g. Transcription objects were instantiated from LeafCorpus objects. The exact type of the LeafCorpus determined the exact type of the Transcription to be instantiated. In the case of instantiation of a Transcription from a browser over generic corpus trees (like the BC browser) we needed another way to specify the exact type of the Transcription object, and a separate mechanism for creation of this object has to be available.

We were also confronted with a number of related cases where the issue of specifying type and location, and subsequent instantiation of the proper object played a role. For example, in the case of the Spoken Dutch Corpus, currently all digital audio data is delivered on a number of CDROMs. Pointing at and accessing this data, including prompting for the proper CDROM, can be solved by a similar mechanism. For the same corpus, a variation of stand-off annotation is used for annotation documents, where separate annotation tiers are kept in separate XML files in separate directories. Instantiation of an annotation document requires pointing at and combining of these separate files.

To solve this range of problems a design was finished that makes use of the standard mechanisms that Java offers to deal with URLs. Based on a generalization of URL syntax and content type the required access mechanisms (like login prompt, prompt for media carrier) are triggered automatically and the proper type of object is instantiated. In case of ordinary URLs and content types everything automatically falls back on Java's built-in URL handling.
As said, new projects required more complex relations between annotations than the ACM could deal with in its original form. For example, for the Spoken Dutch Corpus both utterances and individual words can be (but don't have to be) time aligned, and each word can have a number of associated codes on different tiers. The Spoken Dutch Corpus also required support for syntactic trees.
For the DoBeS project a wide range of legacy material has to be incorporated in the archive and the EUDICO based archive software has to be able to cope with that. Much of this data is Shoebox or Shoebox-style MS Word data. Therefore interlinear glossing formats have to be supported at the level of ACM. Within the DoBeS community, the maximal format requirements are well described by Lieb and Drude in their Advanced Glossing paper [18].
To support all of these structures two basic types of Annotations were added: AlignableAnnotations and ReferenceAnnotations. While AlignableAnnotations has the necessary characteristics to link annotations to time periods, ReferenceAnnotations provide the necessary mechanisms to draw relations between annotations independent of their tier.

In almost every annotation system or format the concept of a tier exists as a kind of natural extension of the concept of a database field applied to time-based data. It is an old idea to "put different things in different places". A tier is the place to put similar things. A tier is a group of annotations that all describe the same type of phenomenon, that all share the same metadata attribute values and that are all subject to the same constraints on annotation structures, on annotation content and on time alignment characteristics.

Metadata attributes for example can be a participant, coder, coding quality, or reference to a parent tier. Constraints on annotation structures can be aspects such as that annotations on the tier refer to exactly one associated parent annotation on a parent tier (*1-n*) or that Annotations on the tier must be ordered in time.
Also annotation content can be constrained by for example a specific closed vocabulary and by a range of possible characters such as Unicode IPA. Constraints on time alignment can also be of various sort such as: Annotations on this tier may not overlap in time.

Explicitly including these types of constraints in the ACM makes tool support for a wide range of use cases and for user interface optimizations possible. For example, known begin or end times of annotations can be reused for new annotations or as constraints on the time segment of other annotations. Text entry boxes can be set up automatically with the proper input method for IPA, annotation values can be specified using popup menus.
Tier metadata, with attribute values specified or not specified, combined with the tier constraints could be reused as a template for the creation and configuration of new tiers, either in the same document or in another. One step further, a set of tier templates could be part of a document template, making it possible to reuse complete configurations of tiers for other documents.

6. Interchange Format

A direct consequence of the ACM is the definition of a suitable and powerful enough annotation interchange format. It is seen as a framework allowing to make ACM content persistent. Here our intentions are fairly comparable with what is currently worked out especially at NIST - called the ATLAS Interchange Format (AIF) [19]. Since AIF could not yet handle all necessary requirements (AIF did not yet support a tier concept) a EUDICO Interchange Format was defined (EAF, see Appendix). However, we would like to join the AIF train to achieve a high degree of interoperability world-wide. Its main structural components are: (1) Time slot values referring to as many as needed concrete time values; (2) information about the tier types and (3) as many AlignableAnnotations or ReferenceAnnotations as necessary. While the first refer to time slots, the latter refers to annotation IDs.

7. Tools

To provide researchers with an efficient annotation and analysis environment, the Institute began early on to setup digitization lines and to build true multimedia tools. The first was the MAC-based MediaTagger annotation tool [20] built in 1994. Consequently, the Institute decided to fully rely on all-digital techniques, i.e. all video and audio signals were digitized. For video it was decided to rely on MPEG1 (after an initial phase of using MJPEG and CINEPAK). Due to its limited resolution, for example, to identify facial expressions in field recordings, it was now decided to change to MPEG2 as a basis for the multimedia archive which has a factor of about 3 more data and bandwidth.

The development of the Java-based EUDICO Tool Set for annotating and exploiting multimedia signals was begun in 1998 and has now reached a flexibility and functionality which makes it one of the most advanced tools for multimodal work. Its nucleus is based on ACM, i.e. it has a comprehensive internal representational power. It has a flexible and easy-to-use annotation and time linking component which allows the user to define his tier setup, which can work with audio and/or video signals in the same way and which makes it possible to do the annotation in various writing systems. It has input methods, for example, for IPA, Chinese, Cyrillic, Hebrew and Arabic. Annotations can either be linked to moments in time in the media stream or to other annotations. It is possible to include hierarchical annotations which is necessary, for example, for an interlinearized representation of morphology.

The EUDICO tool set also provides various views on the multimedia data which can be sound, video, or annotation tracks or other types of signals such as eye tracking tracks. There are a number of stereotypic views on the annotations scientists prefer, therefore EUDICO supports different views and more views can be added according to individual scientists' needs. An important feature is that researchers can easily select and arrange the data tracks they want to see. All viewers in EUDICO are synchronized, i.e. whenever the cursor in a viewer is set to a certain time or segment, all other viewers will move to that instance. The tool set also has a flexible search interface which allows the user to define patterns and associate them with annotation tiers (including all supported input methods) making it possible to enter complex patterns covering several tiers and distances between the patterns. The EUDICO tool set can work in a fully distributed environment where annotation and media tracks are at different locations and support media streaming of fragments. An XML-based generic interchange format was defined (EUDICO Annotation Format), but other formats such as rDBMS, CHAT and Shoebox are also supported.

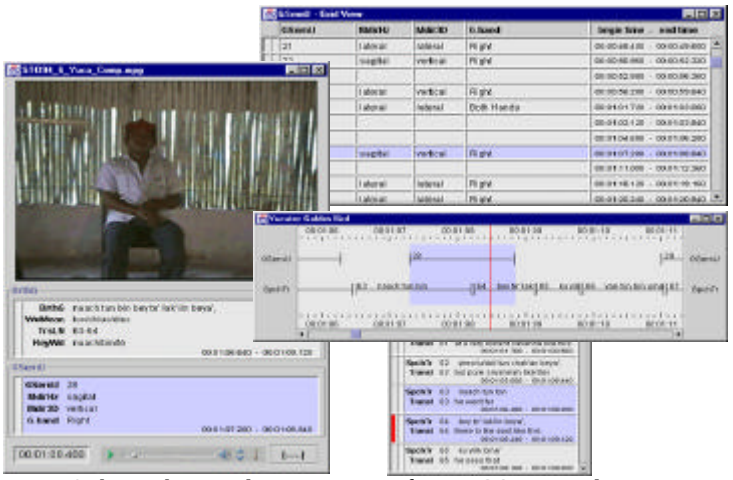


Figure 8 shows the visualization power of EUDICO. Dependent on the project different stereotypic visualizations of the material can be selected. The type of output, the tiers and the order of tiers can be selected by the user. The range of viewers covers dynamic subtitles, a time line view and text viewers with compressed texts.

Tier types can be defined including controlled vocabularies and constraints. Pixel management is very important when dealing with complex tier structures. The user can define the tiers he wants to see and specify the order of presentation. Currently, MPEG1 streaming is supported. MPEG2 is also supported, however downsizing of the video widget is absolutely necessary in order to see the annotations as well.

Further details about the EUDICO Tool Set can be seen on the web-page [21].

8. Conclusions

At the MPI for Psycholinguistic the study of gestures has a long tradition. Gesture recordings are used to better understand the basic mechanisms of the speech production and comprehension processes. Further the usage of gestures in various cultures could help clarifying the relationship between language and thought. Gestures are very much dependent on the culture and the languages spoken in these cultures.

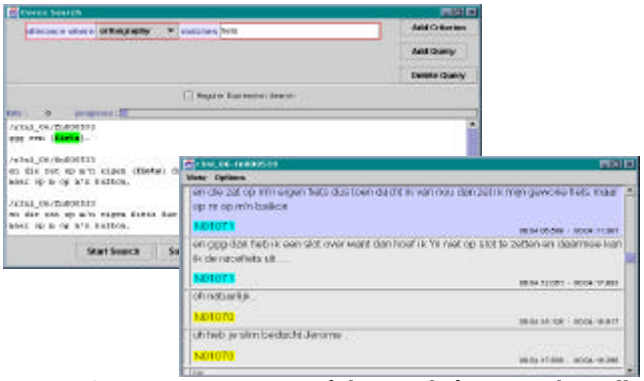


Figure 9 gives an impression of the search feature. It basically allows the user to define search patterns, associate them with tiers and logically combine these patterns to a complete query where also distances can be specified. The result is a list of hits which can be clicked to directly yield the corresponding fragment.

To support this research a large cross-linguistic gesture corpus had to be built including annotations of the speech acts and the gestures. Currently, large international projects have been setup to further investigate the scientific questions raised in this paper.

Such research was only possible by a consequent digitization policy of the institute, by building efficient multimodal annotation and exploitation tools and by powerful mechanisms which help the user to manage large corpora. With the EUDICO and Browsable Corpus technology which was extended within the ISLE project the researchers can rely on tools which will be supported for many years. Since the file formats of both technologies is XML based it can be expected that they will be widely used.

9. References

[1] W.J.M. Levelt (1980). Online processing contraints on the properties of signed and spoken language. In Biological Constraints on linguistic form. U. Bellugi, M. Studdert-Kennedy (eds.). Vgl. Chemie, Weinheim.

[2] G. Richardson (1984). Word recognition under spatial transformation in retarded and normal readers. Journal of Experimental Child Psychology 38, 220-240.

[3] S. Kita, J. Essegbey (to appear). Pointing left in Ghana: How a taboo on the use of the left hand influences gestural practice. Gesture.

[4] S. Kita (1998). Expressing a turn at an invisable location in route direction. In Ernest Hess-Lüttich, J.E. Müller & A. vanZoest (eds.), Signs & SPace. 159-172. Tübingen: Narr.

[5] A. Özyürek, S. Kita (1999). Expressing manner and path in English and Turkish: Differnces in speech, gestures, and conceptualization. In M. Hahn and C. Stones

(eds.), Proceedings of the 21 st Annual Meeting of the Coginitive Science Society. 507-512. Amsterdam.
[6] M. Gullberg, K. Holmqvist (2001). Eye tracking and the perception of gestures in face-to-face interaction vs. on screen. In C. Cave, I. Guaitella, S. Santi (Eds.), Oralite et gesturalite: Interactions et comportemetns multimodaux dans la communication (pp. 381-384). Paris: L'Harmattan.
[7] H. Lausberg, S. Kita (2001). Hemispheric specialization in spontaneous gesticulation investigated in split-brain patients. In C. Cave, I. Guaitella, S. Santi (Eds.), Oralite et gesturalite: Interactions et comportemetns multimodaux dans la communication (pp. 431-434). Paris: L'Harmattan.
[8] M. Seyfeddinipur, S. Kita (2001). Gesture and dysfluency in speech. In C. Cave, I. Guaitella, S. Santi (Eds.), Oralite et gesturalite: Interactions et comportemetns multimodaux dans la communication (pp. 266-270). Paris: L'Harmattan.
[9] S. Kita, I. v. Gijn, H. vd. Hulst (1998). Movement Phases in Signs and Co-speech Gestures, and their Transcription by Human Coders. In I. Wachsmuth and Martin Frühlich (eds.), Gesture and Sign Language in Human-Computer Interaction, Vol. 1371: 23-35. Proceedings of the International Gesture Workshop Bielefeld, Lecture Notes in Artificial Intelligence. Berlin: Springer Verlag.
[10] S. Kita, I. v. Gijn, H. vd. Hulst (2000). Gesture Encoding. MPI Internal Report.
[11] B. MacWhinney (1999). The CHILDES Project: tools for analyzing Talk. Second ed. Hillsdale, NJ: Lawrence Erlbaum.
[12] S. Levinson, S. Kita, P. Wittenburg, H. Brugman (2002). Multimodal Annotations in Gesture and Sign Language Studies. In *Proceedings of the LREC 2002 Conference*, Las Palmas.
[13] www.sil.org/computing/catalog/shoebox.html
[14] www.mpi.nl/world/tg/CAVA/CAVA.html
[15] www.cs.nyu.edu/cs/faculty/grishman/tipster.html
[16] H. Brugman, P. Wittenburg (2001). The application of annotation models for the construction of databases and tools. In Proceed*ings of the Workshop on Linguistic Databases*. Philadelphia.
[17] www.mpi.nl/ISLE
[18] H. Lieb, S. Drude (2000). Advanced Glossing: A language documentation format. Unpublished working paper.
[29] www.nist.gov/speech/atlas
[20] www.mpi.nl/world/tg/lapp/mt/mt.html
[21] www.mpi.nl/world/tg/lapp/eudico/eudico.html
www.mpi.nl/tools

10. Appendix

This appendix contains the DTD for the EUDICO Annotation Format (EAF).

```
<!-- edited with XML Spy v4.1 U (http://www.xmlspy.com) by Hennie
   Brugman (Technical Group) -->
<!--
     Eudico Annotation Format DTD
     version 0.1
     July 5, 2001
-->
<!ELEMENT ANNOTATION_DOCUMENT (HEADER,
   TIME_ORDER, TIER*, LINGUISTIC_TYPE*, LOCALE*)>
<!ATTLIST ANNOTATION_DOCUMENT
     DATE CDATA #REQUIRED
     AUTHOR CDATA #REQUIRED
     VERSION CDATA #REQUIRED
     FORMAT CDATA #FIXED "1.0"
>
<!ELEMENT HEADER EMPTY>
<!ATTLIST HEADER
     MEDIA_FILE CDATA #REQUIRED
     TIME_UNITS (NTSC-frames | PAL-frames | milliseconds)
   "milliseconds"
>
<!ELEMENT TIME_ORDER (TIME_SLOT*)>
<!ELEMENT TIME_SLOT EMPTY>
<!ATTLIST TIME_SLOT
     TIME_SLOT_ID ID #REQUIRED
     TIME_VALUE CDATA #IMPLIED
>
<!ELEMENT TIER (ANNOTATION*)>
<!ATTLIST TIER
     TIER_ID ID #REQUIRED
     PARTICIPANT CDATA #IMPLIED
     LINGUISTIC_TYPE_REF IDREF #REQUIRED
     DEFAULT_LOCALE IDREF #IMPLIED
     PARENT_REF IDREF #IMPLIED
>
<!ELEMENT ANNOTATION (ALIGNABLE_ANNOTATION |
   REF_ANNOTATION)>
<!ELEMENT ALIGNABLE_ANNOTATION
   (ANNOTATION_VALUE)>
<!ATTLIST ALIGNABLE_ANNOTATION
     ANNOTATION_ID ID #REQUIRED
     TIME_SLOT_REF1 IDREF #REQUIRED
     TIME_SLOT_REF2 IDREF #REQUIRED
>
<!ELEMENT REF_ANNOTATION (ANNOTATION_VALUE)>
<!ATTLIST REF_ANNOTATION
     ANNOTATION_ID ID #REQUIRED
     ANNOTATION_REF IDREF #REQUIRED
     PREVIOUS_ANNOTATION IDREF #IMPLIED
>
<!ELEMENT ANNOTATION_VALUE (#PCDATA)>
<!ELEMENT LINGUISTIC_TYPE EMPTY>
<!ATTLIST LINGUISTIC_TYPE
     LINGUISTIC_TYPE_ID ID #REQUIRED
>
<!ELEMENT LOCALE EMPTY>
<!ATTLIST LOCALE
     LANGUAGE_CODE ID #REQUIRED
     COUNTRY_CODE CDATA #IMPLIED
     VARIANT CDATA #IMPLIED
>
```

Development of the User−State Conventions for the Multimodal Corpus in SmartKom

Silke Steininger[†], Susen Rabold[†], Olga Dioubina[†], Florian Schiel[*]

[†]Institute of Phonetics and Speech Communication
[*]Bavarian Archive for Speech Signals (BAS)

Ludwig−Maximilians−University, Schellingstr.3, 80799 Munich, Germany
{kstein, rabold, olga, schiel}@phonetik.uni−muenchen.de

Abstract

This contribution deals with the problem of finding procedures for the labeling of a multimodal data corpus that is created within the SmartKom project. The goal of the SmartKom project is the development of an intelligent computer−user interface that allows almost natural communication with an adaptive and self−explanatory machine. The system does not only accept input in form of natural speech but also in form of gestures. Additionally the facial expression and prosody of speech is analyzed.
To train recognizers and to explore how users interact with the system, data is collected in so−called Wizard−of−Oz experiments. Speech is transliterated and gestures as well as user−states are labeled. In this contribution we will describe the development process of the User−State Labeling Conventions as an example for our strategy of functional labeling.
Key−words: multi−modal, annotation, user−states, human−machine interaction, coding conventions.

1. Introduction

The goal of the SmartKom project is the development of a multimodal dialogue system that allows the user to interact almost naturally with the computer. Among other things the emotions of the user are taken into account by the system. Since not much is known about the role emotions play in a human−machine dialogue, data is collected in Wizard−of−Oz experiments. The analysis of the interaction of the users with the simulated system can reveal which emotions occur in such a situation, in which way the emotions are expressed and in what connection. For such an analysis the data has to be labeled[1].

This contribution deals with the problem of how to define a labeling procedure for emotions, respectively. user−states[2]. We will first describe shortly how the data was collected that was used for the development of the labeling procedure. Then we describe the requirements the procedure had to meet. After that we give an overview over the steps of the development process of the procedure and some open questions.

2. Collection Of Multimodal Data

The data collection is done with the Wizard−of−Oz technique: The subjects think that they interact with an existing system but in reality the system is simulated by two humans from another room.

In each Wizard−of−Oz session spontaneous speech, facial expression and gestures of the subjects are recorded with different microphones, two digital cameras (face and sideview hip to head) and an infrared sensitive camera (from a gesture recognizer: SIVIT/Siemens) which captures the hand gestures (2−dimensional) in the plane of the graphical output. Additionally, the output to the display is logged into a slow frame video stream. Each subject is recorded in two sessions of about 4.5 minutes length each. For more information on technical details of the data collection see Türk (2001).

3. Developing the Labeling Procedures − Starting Point

3.1 Goals

The labeling of user−states in SmartKom serves two main functions:

1. The training of recognizers.
2. The gathering of information how users interact with a multimodal dialogue system and which user−states occur during such an interaction.

These two goals had to be satisfied with the labeling procedures we had to define. For practical and theoretical reasons we decided against a specific system like the "Facial Action Coding System" of Ekman (1978) where the precise morphological shape of facial expressions is coded, but used a simplified, practice−oriented system. The user−states are defined with regard to the subjective impression that a human communication partner would have, if he would be in place of the SmartKom system. This is a functional definition: Not the user−state per se is coded, but the impression the communicated emotion or state generates.

In Steininger, Lindemann & Paetzold (2002a) we already discussed this approach with regard to gestures[3]. The next paragraphs explain our approach relating to user−states.

[1] The development and structure of the gesture labeling is described in detail in Steininger, Lindemann & Paetzold (2002a). The transliteration conventions can be found in Oppermann et al. (2000). The special problem of combining the information of the different labeling steps and the transliteration is discussed in Schiel et al. (2002) at this workshop.

[2] The name “emotion labeling” was changed in “user−state labeling” because the targeted episodes in the data comprise not only emotional, but also cognitive states.

[3] Our gesture coding system also defines hand gestures functionally (not morphologically). A labeled unit is coded with regard to the intention of the user, i.e. with regard to his (assumed) discrete goal.

3.2 Practical Requirements

To satisfy the two goals of the labeling process mentioned above the following requirements had to be met. They apply to transliteration, gesture and user−state labeling.

1. The labels should refer to the functional level[4], not the morphological level. For theoretical reasons we want to use a functional coding system (see below). However, the decision is also made for practical reasons since the structural coding of e.g. facial expressions is exceedingly time consuming.

2. The labels should be selective. Functional codes (as indirect measurements) are not as exact as direct methods, therefore exceptional care has to be taken to find labels that are well−defined, easy to observe and unproblematic to discriminate by means of objective (communicable) criteria. This is even more true for user−states than for gestures because communicable criteria for the discrimination of functional user−state categories are hard to find.

3. The coding system should be fast and easy to use.

4. The resulting label file should facilitate automatic processing (a consistent file structure, consistent coding, non−ambiguous symbols, ASCII, parsability) and preferably should be easy to read.[5]

5. The main categories and most of the modifiers should be realized as codes and not as annotations, in order to heighten consistency. Annotations (free comments and descriptions that don't follow a strict rule) are more flexible, but codes (predefined labels from a fixed set) increase the conformity between labelers.

4. Definition of the User−State Coding System

The questions that have to be solved to detect user−states automatically are: Which features of the face and of the voice contribute to an emotional impression − and in which degree does each feature contribute to the impression? Which of these features can be detected automatically?

If we already knew the answers it would make sense to define coding conventions that mark these features in the data. But since we are far from answering these questions conclusively we decided to use another strategy: The labelers mark beginning and end of a user−state sequence and sort it into one of several subjective categories.

A human in a conversation with another human is able to judge which emotion or user−state his or her communication partner shows. Therefore he or she should be able to discriminate relevant user−states in a video. Of course the labeler does not know which emotion is truly present in his communication partner/a human in a video and he or she will make mistakes. But he or she should be good enough to use his emotion−detection capability to keep the conversation smooth. This goal is the same for the system − it should be able to detect which user−state is present in its communication partner to keep the conversation smooth.

This consideration we used for the definition of the user−state coding system.

4.1 First Step: Pretest − Labeling with some defined subjective categories

First we decided to look for several categories that were deemed interesting for user−state recognition: "anger/irritation", "boredom/lack of interest", "joy/gratifi−cation (being successful)", "surprise/amazement", "neu−tral/anything else". A few sessions were labeled with these categories. Beginning and end were defined by an observable change in the emotional state of the user. It was marked if the user−state seemed "weak" or "strong".

In the first step each session was labeled by at least two different labelers. After the labeling the categories were discussed. "Boredom/lack of interest" was excluded because it could not be distinguished from "neutral". "Neutral" and "anything else" were separated into two different categories because many sequences were found where the users definitely did not show a neutral expression but no meaningful label could be given. Two new categories were included to describe user−states that occurred quite often in the data and are important in the context of human−computer interaction: helplessness and pondering/reflecting.

The label "anything else" comprises three cases:

1. Grimaces with no emotional content, for example playing with the tongue in the cheek, twitching muscles etc. (about 65%).

2. Emotional sequences that have no label in our system, for example disgust (about 5%).

3. States that seem to have an emotional or cognitive meaning, but cannot be decided upon by the labelers (about 30%).

The three cases were put together into one category because they all comprise sequences that are not suited as training material.

Cases like number 2 (disgust etc.) are very uncommon in our context and because of this an extra category was not deemed worthwhile. Cases like number 1 (grimaces for physiological reasons) sometimes look very similar to user−states, but have a different meaning − therefore they have to be distinguished from neutral.

Cases like number 3 would be interesting to analyze further because the comprise complex or difficult to understand user−states. They are sorted into the "anything else" category simply for practical reasons: The other labels should be selective, therefore any label that cannot be categorized for certain has to be sorted into "anything else".

4.2 Second Step: Holistic labeling with the conventions

In a second step the sessions were labeled with the following fixed set of categories:

− joy/gratification (being successful)
− anger/irritation
− helplessness

[4] "Functional code" or "functional unit" is sometimes defined differently by different authors. We use the term in accordance with Faßnacht (1979) for a unit that is defined with regard to its effect or its context.

[5] Many of the practical criteria were adopted from the transliteration conventions for speech in SmartKom, see Oppermann et al. (2000).

− pondering/reflecting
− surprise
− neutral
− unidentifiable episodes

Consistency was achieved by two correction steps. Final correction was done by the same corrector for every session. Difficult episodes were discussed.

Figure 1: Example of the front view that is used for the holistic and the facial expression labeling. The picture was taken from an episode that was labeled as "anger/irritation" in the holistic labeling step.

4.3 Third step: Finding features

The categories are assigned according to the subjective impression of the labelers. Nevertheless the goal is to find detectable features. Additionally the categories have to be describable with observable criteria − otherwise no one else apart from the labelers will be able to understand the content of the labels.

Therefore, for each category some characteristic features were listed. A feature was included in the list if it occurred regularly or if it seemed very distinctive of a category for some subjects.

This step of the development process is still in progress. At the moment the features are simply an aid for labeling. However, the feature list could be studied with objective methods to judge which features are good candidates to be "indicators" for a category.

4.4 Fourth Step: Overcoming some limitations

With the holistic labeling system we were relatively sure to catch all relevant user−state episodes and to sort them into selective categories. However, a serious problem had to be solved: For the recognition of facial expressions the coding system was not well suited. Because of the holistic approach the labels included not only information from the facial expression, but also from the voice and from the context. This is a problem because a facial expression recognizer derives information only from the facial expressions and a prosody recognizer derives information only from the voice.

First, we tried to solve the problem with a special marker of the source for a category: voice or face. But it turned out that it was very difficult to make the judgment with regard to the source. Additionally, only very few episodes with the source "voice" could be found.

We abandoned the source marker and included two different labeling steps: Labeling of the facial expression without audio and prosodic labeling.

For the facial expression labeling a different labeler−group watched the videos without audio. The labelers started with a pre−segmented file (from the holistic labeling) to avoid missing subtle episodes that are hard to perceive without audio and context information. This pre−segmentation was derived from the holistic labeling − the names of the categories (apart form "neutral") were deleted, the borders were retained.

Since it seemed to be difficult to use the functional approach with regard to the voice, we adopted a formal coding system that was used in Verbmobil (Fischer, 1999) and changed it to suit our needs in SmartKom.

For the prosodic labeling the transliteration files are filtered: Only the orthographic transcript remains so that the transliteration labels don't divert the prosodic labelers. For the labeling prosodic features like pauses, irregular length of syllables and other prosodic features which could reveal the emotional state of the particular user are marked. There are nine categories for the prosodic labeling:

1. Pauses between phrases
2. Pauses between words
3. Pauses between syllables
4. Irregular length of syllables
5. Emphasized words
6. Strongly emphasized words
7. Clearly articulated words
8. Hyperarticulated words
9. Words overlapped by laughing

The labels were chosen according to the requirements for the User−State recognition group in SmartKom and are thought to represent prosodic features that are indicative of emotional speech. Hyperarticulated words for example, can be indicative of anger. However, it is still not known very well which prosodic features occur during which emotional states. Nevertheless, by the comparison between the holistic labeling and the prosodic labeling it should be possible to detect relevant user−states in speech. For more information on the usage of prosodic features as indicators of emotional speech please refer to Batliner et al. (2000).

For a detailed description of the labels and concrete examples for the labeling procedure please refer to our paper at the main conference (Steininger, Schiel & Glesner, 2002b).

4.5 Open Questions

We have to state clearly that the user−state labeling procedure is work in progress. The description of the categories, along with some formal criteria to help differentiate categories that can be mixed easily is not complete. After it's completion, the intercoder agreement has to be measured. At the moment, we can only use the extent of corrections that are done in each correction step

as a rough indicator how reliable the labeling procedure probably is:

Holistic labeling: About 20% of all labels are changed with regard to content. About 10% of the segment borders are changed. This is the case for correction step 1 as well as 2.

Facial Expression labeling: Only one correction step exists. Segments borders have to be corrected almost never. Changes of labels with regard to content occur in about 20% of the cases.

Prosodic labeling: Only one correction step exists. Changes of labels with regard to content occur in about 20% of the cases. Changes of time markers occur in about 50% of the cases.

One other problem that remains are mixed emotions. Since there is no category for mixed emotions, all such cases have to be sorted into "anything else". However, the problem is not as big as it seems: Since we use categories that are defined mainly by subjective impression not mainly by formal criteria, it is rare that a labeler has the impression of a mixed emotion[6]. As already mentioned, the labeler take the viewpoint of a communication partner and try to discern which state his opponent is in. On this level, there almost always is an integrated impression of only one emotion at a time. Many emotional states are mixed of course if one analyses them closely. With a formal system like FACS (Ekman, 1978), mixed emotions correspond to mixed expressions: The face may show anger (for example with a frown) and surprise (for example with an open mouth). In a functional system like ours the viewpoint is taken that it is not known if a frown always means anger and an open mouth always means surprise. If the frown and the open mouth leave the observer (labeler) with the impression of reflecting then this label is given. That is to say that a mixed state on the formal level can lead to a new (holistic) impression on the functional level. Actually this is quite often the case. In most instances there is a clear message for a communication partner. We label only this "clear message", not the subtle undercurrents.

Of course the overall impression can also be of a mixed state. In this case the label “anything else” is given since only very few mixed states were found. Since for the voice a formal system is used and in one labeling step the facial expression is judged without the audio information mixed states for speech and facial expression can occur. In some cases they will be real mixed states but in some cases they will occur because of labeling mistakes.

In our view, formal and functional systems can complement each other, but cannot replace each other because they refer to different levels.

A third important open question is the “anything else” category. For practical reasons some of the most interesting cases “disappear” into this category, namely the episodes that cannot be categorized neatly. Of course it would be of great interest to analyze these difficult episodes further. How could this be done? It is no option to ask the subjects what they felt in the case of an unidentifiable user−state, because with the functional approach the emotions are labeled that are transmitted to a communication partner. Introspective evaluation of the emotion by the user will give a different picture because of effects of social conventions (among other things). To include recordings of other modalities could be helpful: Hesitant movements for example could give hints about the user−state “helplessness”. However, we decided against using additional visual context information because we wanted to focus the labelers on the face and on changes in the voice accepting that some episodes remain unidentifiable. Adding such information later can change the impression (which is highly context dependent), therefore the whole labeling process has to be done again. An interesting option would be to have the unidentifiable episodes judged by a group of naive, untrained labelers (without giving them predefined categories). In this way it could be analyzed if the unidentifiable episodes are episodes that are difficult to understand by a communication partner or if at least some of them form a user state not yet identified as important.

[6] With the expeption of “sarcasm”: Cases where the user is smiling and laughing, but it can be suspected that he is also scornful are labeled as “joy/gratification”. Sarcasm is hard to detect reliably, therefore we decided againgst a special label.

5. Conclusion

With the example of the user−state labeling we show a way to handle the problem of finding a labeling system that is consistent, fast and catches the most important episodes in a human−machine dialogue. Since as yet there is not known enough about good indicators for user−state recognition we decided against a formal/morphological system. Instead we define the labels after practical experience with the data, in this way circumventing the danger of missing important aspects by making assumptions about indicators for automatic detection that cannot be justified very well yet.

Additionally, by combining holistic labeling, labeling of the facial expression and a formal system for the speech we can make up for the disadvantages a purely holistic, functional coding system would have. Through comparing the different label files it is possible to analyze and process the data from many different points of view, looking at the whole or at parts at will.

It is also possible to combine the user−state labels with the gesture labels or the speech transliterations. It could be interesting to analyze which kinds of gestures occur during which kinds of user−states. During helplessness there should be less interactional gestures and more searching gestures, for example. The comparison between the gesture labels and the transliterations is especially interesting with regard to reference words that are possibly uttered. A combination of all three modalities could be useful to analyze the question if there are more hesitations and aborts in the speech and gestures during angry and/or helpless episodes.

With the traditional way of annotating input modalities separately such comparisons are not possible. The labeling of data of multimodal systems allows new ways of studying human−machine interaction. However, this will be successful only if the coding conventions allow the combination of the labeling of the different modalities with ease.

6. References

Batliner, A., Fischer, K., Huber, R., Spilker, J., & Nöth, E., 2000. Desperately Seeking Emotions Or: Actors, Wizards, and Human Beings. In R. Cowie, E. Douglas−Cowie, & M. Schröder (Eds.): *Proc. of the ISCA Workshop on Speech And Emotion*. Belfast: Textflow.

Ekman, P., & Friesen, W. V., Facial Action Coding System (FACS), 1978. *A technique for the measurement of facial action*. Palo Alto, Ca.: Consulting Psychologists Press.

Faßnacht, G., 1979. *Systematische Verhaltensbeobach−tung*. München: Reinhardt.

Fischer, K., 1999. Annotating Emotional Language Data. *Verbmobil Report 236*.

Oppermann, D., Burger, S., Rabold, S., & Beringer, N., 2000. Transliteration spontanprachlicher Daten−Lexikon der Transliterationskonventionen−SmartKom. *SmartKom Technisches Dokument Nr. 2*.

Schiel, F., Steininger, S., Beringer, N., Türk, U., & Rabold, S., 2002. Integration of multi−modal data and annotations into a simple extendable form: the extension of the BAS Partitur Format. To appear in the *Proc. of the 3rd Int. conf. on Language Resources and Evaluation, Workshop On Multimodal Resources And Multomodal Systems Evaluation*, Las Palmas, Spain.

Steininger, S., Lindemann, B., and Paetzold, T., 2002a. Labeling of Gestures in SmartKom − The Coding System. To appear in *Proc. of the Gesture Workshop*, London: Springer.

Steininger, S., Schiel, F., & Glesner, A., 2002b. User−State Labeling Procedures For The Multimodal Data Collection Of SmartKom. To appear in the *Proc. of the 3rd Int. conf. on Language Resources and Evaluation*, Las Palmas, Spain.

Türk, U., 2001. The technical processing in the SmartKom data collection: A case study. *Proc. of Eurospeech*, Scandinavia, p. 1541−1544.

Integration of Multi-modal Data and Annotations into a Simple Extendable Form: the Extension of the BAS Partitur Format

Florian Schiel*, Silke Steininger†, Nicole Beringer†, Ulrich Türk†, Susen Rabold†

*Bavarian Archive for Speech Signals (BAS)
†Institut für Phonetik und Sprachliche Kommunikation

University of Munich, Schellingstr. 3, 80799 München, Germany
{schiel,kstein,beringer,tuerk,rabold}@phonetik.uni-muenchen.de

Abstract

Multi-modal resources typically consist of very different data in terms of content and format. This paper discusses a practical solution for the integration of different physical signals as well as associated symbolic data into a common framework. There are ongoing efforts like for instance the ISLE project to develop guidelines and best-of-practice for the standardized representation of such data collections. Since these efforts have not yet converged into a widely accepted concept, we suggest as a starting point to use two different already existing frameworks that can be easily combined for this purpose: The QuickTime format for the handling of synchronized multi-modal signals and the (extended) BAS Partitur Format for the handling of all symbolic data. We can show that with this simple approach it is already possible to integrate the rather complex data streams of the SmartKom Corpus into an easy-to-use format that will be distributed via the Bavarian Archive for Speech Signals (BAS) starting in July 2002.

1. Introduction

The last years have seen quite a number of projects starting to work on the processing / recognition / output of multi-modal data in man-machine-interaction systems. However, a quick survey in the Web sites of LDC[1], ELDA[2], CSLU[3] as well as in general search engines shows that such data are not widely available to the scientific community outside of dedicated project groups[4]. On the other hand projects like ISLE[5] started with the aim to extend the EAGLES initiative with guidelines and standards for multi-modal data, but has not produced any recommendations yet. Although standards and role models do not exist, in most scientific projects people had to get started collecting data for their special needs, in most cases gathering material for training and evaluation of multi-modal input devices. Almost like twenty years ago when the creation of language resources started to get going the concerned scientists nowadays collect and annotate data to their needs and with the tools and standards available.

So did we when we started to collect data for the German SmartKom project[6] beginning of 2000. Unfortunately, this MO will very likely aggravate the future use of these corpora, which is a shame considering the very high efforts (and costs) that are invested into these resources.

Meanwhile the SmartKom group at BAS has collected a vast amount of multi-modal data (about 1500 GByte) and has solved most of the technical problems that come with such a task. As reported elsewhere (Tuerk, 2001) the SmartKom data collection consists of 9 different audio channels, two high resolution video streams, one infrared video stream (black and white) and a screen capture (very low frame rate), a HID input and a pen input. Within the last year we were faced with the problem to integrate all these different modalities (signals) together with the various annotation of data streams into a common framework that may be used for the final distribution of the corpus (starting in July 2002 with the first release of SK Public). The two main problems here are that on the one hand different modalities are recorded by different non-synchronized capture devices, on the other hand annotations to different modalities are produced with the use of different – sometimes even self-written – software tools. All this results in a huge variety of resolutions, time bases, file formats that will hinder the easy usage of the corpus by others.

2. Practical solution

In this contribution we would like to give a proposal (to be precise: two independent proposals) how to handle these problems with existing frameworks. We do not claim that our proposal will be the ultimate and best solution. However, it could act as an intermediate step that allows the immediate work with multi-modal data and might make the conversion of multi-modal resources into a future standard (whatever it might be) less painful.

Let us first list a few basic requirements denoting the intended characteristics of the framework for multi-modal resources (FMMR). Our intended FMMR

- should be extensible and flexible.
  In almost all cases a fixed format for data resources is bad news for the scientist or developer, because he then uses a lot of unnecessary time to solve data format problems. Although this has been true for monomodal resources as well, the problem multiplies when

[1] http://www.ldc.upenn.edu/
[2] http://www.elda.fr/
[3] http://cslu.cse.ogi.edu/corpora/
[4] The only exception being the M2VTS biometrical corpus available at ELDA
[5] http://isle.nis.sdu.dk/
[6] http://smartkom.dfki.de/

it comes to multi-modal data. Therefore the framework should not be a fixed definition for different kinds of modalities and how to treat them but rather an extensible framework that can be easily adapted to upcoming needs.

- should be easy to process.
  The reason for this key point is obvious. The conclusion is that we may use a well developed format for which tools are available (for instance XML) or that we use such a simple format that it may be processed with standard tools on the operation system level.

- should not integrate signals and annotations in one file format.
  According to our experience in many cases users of a data resources do not need to access all signals or all annotations at the same time. To simplify handling and distribution we therefore strongly recommend that signal and annotation data are separated in storage but linked together via the time base (like it was done in the SAM and BAS Partitur File (BPF) standards).

With these basic requirements in mind our proposed method can be summarized as follows:

1. To integrate the raw data we use QuickTime (QT)[7] for all data that are measured signals or events.
2. To integrate annotations we use BPF or a similar flexible framework (e.g. annotation graphs (Bird, 2001)).
3. We link both representations through the physical time base only.
4. We use what ever necessary relational/hierarchical linking only between the annotation layers.

Note that although we use the BPF in the following examples, this is exchangeable to any other equally qualified format. The point we want to stress here is not the format but that the symbolic (annotation) data should be kept seperate from the signals, but be grouped into a single framework for easier analysis.

We will discuss the pro and cons of our approach in the following section using the SmartKom corpus as an example.

3. Example SmartKom

To demonstrate that our proposal does actually work we show as an example the integration of a complex data collection in the SmartKom project where a wide range of signals and annotations are currently used.

3.1. Integration of signals in QT

Let us first look at the integration of signals into a QT frame. QT allows the integration of several kinds of media into a single multi-media file. Theoretically every signal format that describes physical measurements (signals or events) may be incorporated, if you provide the necessary interface to QT. Fortunately, interfaces for most of the common file formats do already exist. Therefore, it is possible to integrate for instance video, audio, images, vector graphic and even text into a QT frame without the need to transform the single modalities from their original format; since they remain in their original files, it is also possible to access to the data via other tools than the QT player, if necessary. The only problem is the synchronisation of different time bases, e.g. the synchronisation of a video stream with 25 frames per sec on one computer with an audiostream captured at 48 kHz on another system. We have not found yet an elegant solution to synchronize automatically. At the moment we use a technique quite similar as in movie productions: we synchronize manually with regard to a significant acoustical and visual event at the beginning of each recording. Even more difficult is the synchronization of 2D spatial data with the video signals. In the Smartkom corpus the output of the gesture analyzer consists of a stream of coordinates in the working area indicating pointing gestures of the user. We solved this problem by converting the two-dimensional data into so-called sprites – that are little bit maps that move in the visual plane – and then overlap both pictures to synchronize the infrared picture of the hand with the sprite. Please refer to (Tuerk, 2001) for a detailed discussion of the synchronization problem.

In Smartkom a typical session file contains the following tracks:

- video of the face, frontal, DV format.
- video of upper body, from left, DV format.
- video of infrared camera directed on display to capture hand gestures, from top, DV format.
- audio in 10 channels (microphone array (4), directed mic, headset (2), backround noise (2), system output) captured by a 10-channel audio card with 48 kHz
- graphical system output captured by a screen capture application at 4fps, AVI format.
- combined video frame with face, upper body, system output and infrared, AVI format.
- coordinate logfiles: output of either the gesture recognition system (finger tip) or the output of the graphic tableau (pen tip)

For performance reasons all streams are captured on different computers. Coordinate logfiles are transformed into a sprite track to make coordinates visible in the video signals. Then all raw signals are synchronised and integrated into a QT frame.

3.2. Pros and Cons of QT

As mentioned above QT is an open format that serves some of our intended purposes: it is quite easy to use, it is extensible to new, yet unknown formats, and data are accessible via the QT standard library. The synchronization is still a problem but solvable. The alternative would be a fully synchronized capturing hardware, but that was far out of our budged range. The original formats of the data are still accessible on the distribution media which makes the

[7] http://developer.apple.com/techpubs/quicktime/quicktime.html

access easy for people that do not want to use QT. Furthermore, parts of the synchronized stream may be used across different data collections.

When the SmartKom project started we also discussed other possible formats than QT. The Java Media Framework (JMF) was already out at that time and would have had the advantange to run completely in JAVA. However, this also caused a very low performance compared to QT which is coded in C++ (encapsulated in a JAVA class library). Also, we could not get necessary drivers in JMF for our intended platforms, for instance no recording drivers for Mac and no DV codec.
The other alternative would have been the Microsoft Media Format (MMF, nowadays mostly replaced by AVI). MMF was only available for MS platforms and – being a mere format definition and no consistent system like JMF or QT – was not flexible enough for our needs.

One major drawback of QT is the still missing QT library and QT player for Linux OS (we managed to get a QT player running in a Win emulation environment, but the performance is very bad). We hope that with the further spreading of QT this will be solved in the near future.
Depending on how many video streams are integrated into the QT frame it is sometimes necessary to spread the frame over more than one DVD-5 which makes working with the data difficult. Also the time deviation between the time bases of the capturing devices is getting significant in longer recording sessions. We avoid this by restricting the length of one recording session to 300 sec.

Figure 1 shows four data streams of a SmartKom recording within a single flattened video frame. In the upper left quadrant the video signal of the face camera is shown; in the upper right quadrant the video signal of the body from the left; in the lower left quadrant the displayed output of the system, in the lower right quadrant the output of the system and as an overlay the video signal of the infrared camera that captures the user's gestures. The shown frame is actually from a video stream that was calculated from the original QT frame; the QT Player Pro is principally capable to show many video streams simultaneously, however the performance on a standard Intel platform is still unsatisfying.

3.3. Integration of Annotations into BPF

During the last 5 years we have shown that the BAS Partitur Format (BPF) developed at the Bavarian Archive for Speech Signals in 1995 is very successful to integrate so called 'symbolic information' (that is in most cases some kind of annotation) of speech recordings into a simple text based format (see for instance (Schiel et al., 1998)). A BPF is a simple text file very similar to the first SAM label file standard, but has no fixed format concerning the syntax and semantics of the contained tier information blocks. Therefore it is quite easy to extend the format to new needs as long as the meta structure is followed to. Based on the UNIX filter concepts it is possible to add new tier information blocks to a BPF without the need to re-write existing application software (as long as this software does not need to access to the new tier information, of course). A simple chaining mechanism within the different tiers allows the integration of annotations without any direct link to the physical time base; by following the chaining to such a tier all remaining tiers are automatically projected to their right position within the signal.

Let us have a closer look at the structure of the BPF[8]:
A BPF file is a simple ASCII file in which each line has a three character key followed by a colon at the beginning that defines the syntax and semantic of this particular line. A BPF consists of a mandatory header structure (compatible to SAM) that must contain a minimum of descriptors, for instance:

```
LHD: Partitur 1.2.11
REP: Muenchen
SNB: 2
SAM: 16000
SBF: 01
SSB: 16
NCH: 1
SPN: ABZ
LBD:
```

Most important entry in this context is 'SAM' which denotes the sampling frequency for all time references in the following annotation tiers.
After this header block an arbitrary number of tier blocks may follow marked by their respective line key. Registered BPF tiers together with their syntax and semantics can be found on the BAS Web pages. For instance the tier block

```
ORT:    0       all
ORT:    1       right
ORT:    2       Mister
ORT:    3       Durante
ORT:    4       <uh>
```

transcribes the pure lexical words of a short utterance. The numbers in the second column are 'links' between different tiers. In principle there may any sort of links units defined (for instance chunks, words, syllables, events etc.). At the moment the BPF standard uses only one type of link that is the word unit counted from the beginning of the recording. Therefore BPF tiers come in only 5 basic types:

1. Events attched to a word, a group of words or the time slot between two words.
2. Events that denote a segment of time without a relation to the word structure.
3. Events that denote a singular time point without a relation to the word structure.
4. Events that denote a segment of time associated with a word, a group of words or the time slot between two words.
5. Events that denote a singular time point associated with a word, a group of words or the time slot between two words.

[8] http://www.bas.uni-muenchen.de/Bas/BasFormatseng.html

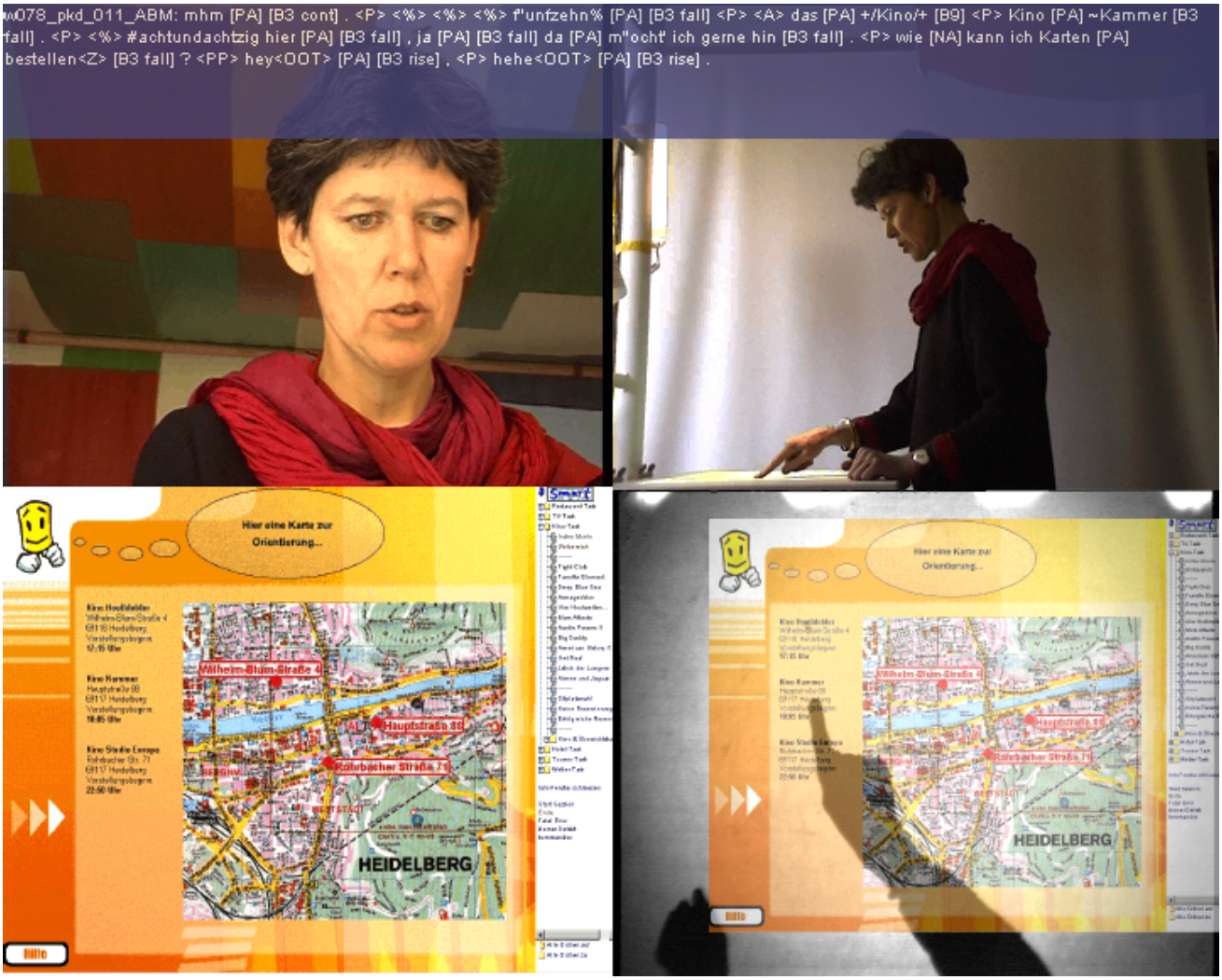


Figure 1: Four synchronized video streams extracted from a SmartKom QT file (see text)

The tier blocks have no preference in order[9] nor hierarchical structure. It is therefore quite easy to cut and paste BPF tiers with standard UNIX tools.

We have shown that the BPF is capable to integrate a variety of symbolic information that was produced within the German Verbmobil project corpus. These data range from simple word alignment over complex syntactic-prosodic tagging up to syntax tree structures. A total of 21 different tiers to the speech signal were used in the Verbmobil corpus (Weilhammer et al., 2002).

Encouraged by this success we started to think about the possibility of integrating symbolic information of multimodal data as well. Surprisingly enough we managed without changing the meta structure of BPF to integrate the following tier information into an BPF (in brackets the corresponding BPF tier keys):

- SmartKom Transliteration of audio channels (TRS,SUP,NOI,ORT,KAN)
- Turnsegmentation (TRN)
- Segmentation and labeling of gestures in the 2D plane (GES)
- Segmentation and labeling of user state (facial and speech) (USH)
- Segmentation and labeling of user state from facial expression only (USM)
- Segmentation and labeling of complex prosodic features to recognize 'emotions' (USP)

Please note that the above annotations are produced with a variety of different software tools (eg. USS, CLAN, Interact). Simple Perl scripts are used to transform the label and segmentation information into the BPF tier information block and add them by concatenation to the existing BPF.

The following example shows an extract from a SmartKom BPF. For better readability the file is abbreviated to the first 12 words of the dialogue and the header block is omitted.

```
TRS:    0       <"ah> [NA] [B2]
TRS:    1       hallo [PA] [B3 fall] . <A> <P>
TRS:    2       kennst [NA]
TRS:    3       du
```

[9] not even within one tier, although the readability is better if the entries follow the time flow

```
TRS:    4       den [B2]
TRS:    5       Wetterbericht [PA]
TRS:    6       f"ur
TRS:    7       heute
TRS:    8       abend [B3 fall] ? <P>
TRS:    9       <:<#> na:> [NA] [B2] ,
TRS:    10      vergi"s [PA]
TRS:    11      es [B3 fall] . <#>
...
SUP:    42,43   w104_mt_SMA.par @1m"ochtest @1du
SUP:    55      w104_mt_SMA.par Pl"atze . <P>2@>
SUP:    56      w104_mt_SMA.par <:<#> hier3@:>
SUP:    61      w104_mt_SMA.par bitte . <P>4@>
ORT:    0       <"ah>
ORT:    1       hallo
ORT:    2       kennst
ORT:    3       du
ORT:    4       den
ORT:    5       Wetterbericht
ORT:    6       f"ur
ORT:    7       heute
ORT:    8       abend
ORT:    9       na
ORT:    10      vergi"s
ORT:    11      es
...
KAN:    0       QE:
KAN:    1       hal'o:
KAN:    2       k'Enst
KAN:    3       d'u:+
KAN:    4       d'e:n+
KAN:    5       v'Et6#b@r"ICt
KAN:    6       f'y:6+
KAN:    7       h'OYt@
KAN:    8       Q'a:b@nt
KAN:    9       n'a+
KAN:    10      f6g'Is
KAN:    11      Q'Es+
...
TRN:    66560   197888  0,1,2,3,4,5,6,7,8,9,10,11 002
TRN:    377984  43776   12,13,14,15 004
...
NOI:    1;2     <A>
NOI:    9       <#>
NOI:    11;12   <#>
...
USH:    0       244480  Neutral
USH:    244480  519040  "Uberlegen/Nachdenken
USH:    517760  25600   Hand im Gesicht
...
USM:    0       515840  Neutral
USM:    515840  216960  "Uberlegen/Nachdenken
USM:    517760  25600   Hand im Gesicht
...
USP:    1364144 3936    27      CLEAR_ART
USP:    1377776 3536    30      CLEAR_ART
USP:    3437728 5856    63      EMPHASIS
USP:    3983392 14992   73      PAUSE_SYLL
...
GES:    265600  32000   U-Geste U - "uberleg - \
        p re Stift nicht erkennbar     640
GES:    376320  30080   I-Geste I - tipp +     \
        re Stift nicht erkennbar
GES:    515200  29440   R-Geste R - emot -     \
        re Hand 393600  8320 "Uberlegung/Nachdenken
...
```

In this example the following tier blocks are contained (see references for details about labeling systems and conventions):

- TRS : SmartKom transliteration (Oppermann et al., 2000)
- SUP : Labeling of cross talk between user and system
- ORT : Lexical entity
- KAN : Citation form in SAM-PA
- TRN : Turn segmentation
- NOI : Noise labeling
- USH : User state labeling using video and audio (Steininger et al., 2002b)
- USM : User state labeling using video only (Steininger et al., 2002a)
- USP : Prosodic labeling of features for user state detection
- GES : Labeling of 2D gestures (Steininger et al., 2001)

3.4. Pros and Cons of BPF

BPFs of Smartkom are fully compatible to BPFs of mono-modal resources. For instance we can easily train a speech recognizer with the data of Smartkom as well as the data of Verbmobil together, since the BPFs tier information blocks for this purpose are identical.
Since the BPF is an open format it is very simple to extend it, for instance by a new tier that contains the time synchronized coordinates of the finger tip delivered by an early stage of the gesture recognizer.
As defined in the BPF format the link to the actual physical signals is solely achieved by reference to the physical time base. It is clear that by doing this the format of the individual signals is arbitrary. It may be the QT format that we use; it may be another format or it may be even just an extraction of a certain modality, as long as the time synchrony is maintained.
Software tools that read only a specific tier information do not need to be adapted when the BPF is extended to a new tier (except of course that the tool needs to process the new tier blocks).
Since the BPF is a simple ASCII file it is usable across platforms.

The BPF does not allow free hierarchical structuring as for instance in the EMU system.
There is no provision in BPF to use UNICODE for special languages or for IPA.
There is no general purpose viewer available for BPF. Up to now we use Praat[10] or SFS[11] to view traditional mono-modal BPFs resources. For the SmartKom corpus we use the QT library that allows to blend in time-aligned text labels as can be seen in figure 1.
There is no dedicated databank system for the BPF. Although we have developed a PROLOG based databank system for the Web that allows simple and complex queries, this is not a general purpose tool. However, it is quite easy to import BPF files into any data bank system.
Last but not least: BPF is not XML. We have started to use parsers that convert BPF tiers into XML. However, it turns out that BPF is easier to read by humans than the XML version.

4. Conclusion

Our approach to use two existing data frameworks, QickTime (QT) and BAS Partitur Format (BPF) for multimodal data collections was borne out of the need to get started without having any role models and/or applicable

[10] http://www.praat.org/
[11] http://www.phon.ucl.ac.uk/resource/sfs/

standards. We recognize that our current mode of operation is a compromise with some drawbacks. On the other hand it is quite surprising that the integration of multi-modal signal data together with their annotations went rather smoothly. We hope that our experiences will help other researchers that face similar logistic problems as well as researchers that are in the process of defining best-of-practice procedures in the field of multi-modal speech resources.
The SmartKom corpus will be made accessible for the public beginning July 2002. Following our policies with monomodal speech resources we will provide a free access to the symbolic data of the corpus via simple FTP download from the BAS server[12]. To obtain the QT files on DVD-5 media please contact bas@bas.uni-muenchen.de or consult the general BAS Web documentation[13].

5. References

St. Bird. 2001. A formal framework for linguistic annotation. *Speech Communication*, 33(1,2):23–60.

D. Oppermann, S. Burger, S. Rabold, and N. Beringer. 2000. Transliteration spontanprachlicher Daten - Lexikon der Transliterationskonventionen. TechDok 02-V4, The SmartKom Project.

F. Schiel, S. Burger, A. Geumann, and K. Weilhammer. 1998. The Partitur Format at BAS. *Proc. of the 1st Int. Conf. on Language Resources and Evaluation, Granada, Spain*, pages 1295–1301.

S. Steininger, B. Lindemann, and T. Paetzold. 2001. Labeling of gestures in SmartKom - The coding system. *Springer "Gesture Workshop 2001", London*, page to appear.

S. Steininger, S. Rabold, O. Dioubina, and F. Schiel. 2002a. Development of the user-state conventions for the multimodal corpus in SmartKom. *LREC Workshop on "Multimodal Resources", Las Palmas, Spain*, page to appear.

S. Steininger, F. Schiel, and A. Glesner. 2002b. Labeling procedures for the multimodal data collection of SmartKom. *Proceedings of the 3rd Int. Conf. on Language Resources and Evaluation, Las Palmas, Spain*, page to appear.

U. Tuerk. 2001. The technical processing in the SmartKom data collection: A case study. *Proceedings of EUROSPEECH Scandinavia*, pages 1541–1544.

K. Weilhammer, F. Schiel, and U. Reichel. 2002. Multitier annotations in the Verbmobil corpus. *Proc. of the 3rd Int. Conf. on Language Resources and Evaluation, Las Palmas, Spain*, page to appear.

[12] ftp://ftp.bas.uni-muenchen.de/pub/BAS

[13] http://www.bas.uni-muenchen.de/Bas

Multimodal Corpus Authoring System:
multimodal corpora, subtitling and phasal analysis

Anthony Baldry* and Chris Taylor†

*Dip. LLSM,University of Pavia,
Strada Nuova 106/c I-27100 Pavia
baldry@gemini.unipv.it

†SSLMIT, University of Trieste,
Via F.Filzi 14 - 34132 Trieste
taylor@sslmit.univ.trieste.it

Abstract

Designed by Baldry and Thibault and constructed by Beltrami and Caglio, MCA is a multimodal concordancer that identifies recurrent patterns in films. As an authoring tool, it enables researchers, however imperfectly, to view short pieces of film and simultaneously to write multimodal descriptions of them. Using MCA's editing tool, researchers can segment film into functional units and, while viewing these units, type out detailed annotations relating both to the semiotic resources they deploy and the functions they perform within the film. The incorporated relational database allows researchers to search the corpora thus created and identify patterns in them, all of which leads to a further round of hypothesis formulation, segmentation, description and comparison of results. As exemplified by the work carried out by Baldry and Thibault on a corpus of TV car ads, MCA was initially conceived as part of research into the applicability of the systemic-functional approach to multimodal description (Halliday, Kress and van Leeuwen) and in particular Gregory's concept of phase and transition. MCA has since been experimented in various projects within *LINGUATEL* (claweb.cla.unipd.it/Linguatel/Pavia/MCA.htm). As the article explains, one such project, namely corpus-driven screen translation, has led the MCA interface to be partly redesigned.

1. Introduction

If you are planning your summer holidays abroad this year, you may well decide to learn the local language, or at least just enough to understand what people around you are saying. If you have no time for evening classes, then you will want to do this in your own home. You could, of course, watch a DVD film in the chosen language, switching on the subtitling in that language so as to be able to identify at least some of the words being spoken. But you will soon realise that a DVD presenting your favourite film will not normally allow you to select *all the cases* of a specific activity or the ways in which that activity 'translates' into grammatical structures, whether those of your own language or the foreign language you have chosen to study. Nor will it allow you to check how a particular word or combination of words is typically used in the film. Thus, while DVD may be a great advance over the VHS cassette, when it comes to language learning it has so far provided only limited forms of access to film and video texts. Without the possibility of confirming your intuitions about the way your chosen foreign language works in relation to your own language, you will soon give up. The end of your language learning plans!

Now consider instead watching, and listening to, film texts in a foreign language using an Internet-based multimodal concordancer that carries out targeted searches in film corpora. By definition, such a tool allows you to carry out multiple 'incursions' into film texts, some of which are likely to correspond to your preferred associative patterns and learning strategies. You might, for example, want to take a strictly grammatical approach, searching, for example, for all the cases in a corpus of utterances that correspond to English "you can" carried out in MCA by a query of the type: *AD contains you can*.

Presented, as in any concordance, as a series of rows, a major difference between multimodal concordancing and the linguistic variety lies in the fact that by selecting the player symbol on the left-hand side of each row, you can see and hear *exactly* that part of the film which contains all (and only) the foreign language expressions that correspond to the concordance query (in this case Italian equivalents for "you can"). Moreover, the returned search also transcribes the words used in the foreign language thus assisting word recognition – so important in the initial stages of language learning. The search also indicates the number the text has in the corpus, so that a further query will allow you to listen to your chosen expression in the context of the entire text. Such a query will be of the type: *AD contains n* (where *n* is the number indexing the specific text).

However, your language learning strategies might be such that you tend to shy away from an overtly grammatical approach. You may well prefer listening to a more extensive piece of text using dual-language subtitling. Given that a multimodal concordancer is likely to be based on a *relational* database (in the case of MCA, *Microsoft Sequel Server*), more complex searches can be made – *de facto* a combination of several searches. A set of dual-language subtitles will returned by a query of the type (see Fig. 1): *AD contains text + English subtitle contains + Italian subtitle contains.* Pursuing this approach, you might well decide to select specific grammatical patterns that illustrate and compare, for example, the way questions are formed in the languages under consideration: queries would be of the type: *AD contains text + Italian subtitle contains questions + English contains questions*. You could, of course, mix the

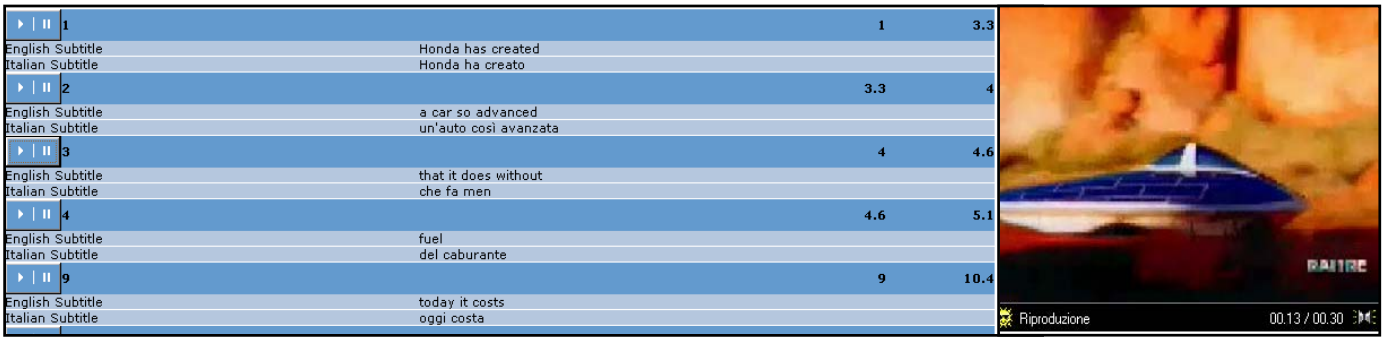


Fig. 1 Dual-language subtitling generated by a relational database

two requirements: *AD contains text* + *Italian subtitle contains* + *English contains questions*. And you could also decide that instead of car adverts you want to listen to (and watch) something else – hence a query of the type *TV News contains…* etc. Flexible as this may seem, as your search skills increase you may well want to integrate the use of subtitling, translation and grammatical patterns with other strategies such as those that explore specific human activities or textual properties. Here a slightly different type of query can be applied. A search of the type: *AD contains text* + *SLOGAN contains YES* (or + *SONG contains YES* or + *HIDING contains YES*) will respectively find all the cases in the corpus exemplifying written and/or spoken slogans, songs and examples of hiding. And as your comprehension of the target language increases, you might also want to go beyond this, associating a linguistic approach to language learning with explorations of meanings made in other semiotic modalities – for example, hand and body movements that couple with language to make multimodal meanings in a way that even the casual observer will recognize as fundamental to a film's overall meaning: e.g. touching something or somebody, possible with a search of the type: *AD contains text* + *touches contains: YES*.

This brief illustration exemplifies how a multimodal concordancer can be used to achieve specific applicative functions (such as language learning) within a multimodal approach to text analysis. Indeed, at the time of writing MCA is still a prototype that is constantly being redesigned – for example, to make it suitable for the learning of minority European languages using the principle of query-generated screen overlays (subtitling, captioning and other more visually-oriented overlays). Other applications include the use of a multimodal concordancer within University courses to help students to understand the multimodal organization of texts, including, as Taylor explains in the following section, efforts undertaken by the Trieste *LINGUATEL* research group to guide students in their learning about screen translation (for a bibliography see Gottlieb).

Indeed, the next section provides a summary of the thinking that led to the original development of MCA and its constant redefinition, a matter discussed in more detail by Baldry in the subsequent section, which also describes MCA's technical specification in relation to research into texts as consisting of phases and transitions between phases, an approach ultimately concerned with defining the typical characteristics of specific multimodal genres.

2. MCA in Trieste

The University of Trieste *LINGUATEL* research unit, as part of a wider national research initiative in Italy sponsored by the Ministry, specialises in multimodal text analysis and the devising of strategies for the translation and subtitling of video text. An example of the work carried out by the unit provides the opportunity to describe how MCA can work in practice. Many types of dynamic text have so far been analysed by the Trieste group (feature films, TV soap operas, cartoons, advertisements, documentaries, news broadcasts, etc.) in particular by using the device of the multimodal transcription, originally devised by Thibault and Baldry (Baldry. 2000, Kress and van Leeuwen, 1996). The multimodal transcription technique consists of breaking a film down into single frames of, say, one second duration and minutely analysing their component parts (visual image, kinesic action, soundtrack, dialogue, etc.) thereby providing an approach that really gets to grips with the *multimodal* side of screen translation (see Fig. 2). It provides an ideal tool for analysing the multimodal text in its entirety and drawing the relevant conclusions in terms of how meaning can be successfully conveyed by the various semiotic modalities in operation, and thus how dispensable or indispensable the verbal element is in different sets of circumstances. From this premise it is possible to make informed choices regarding the translation strategies to adopt in subtitling a film.

One type of video text subjected to this multimodal approach was the television comedy series 'Blackadder'. Humour is notoriously difficult to translate, especially apparently British humour, as it involves a large number of interweaving factors: word play, register shifts, timing, characterisation – and creating the humorous effect through subtitles is doubly difficult. The episode examined here, from the Elizabethan era series, features a highly implausible plot involving Lord Blackadder and his scatter-brained assistants, the dreadful Baldrick and Lord Percy, who have inadvertently executed the wrong man, while temporarily in charge of the royal prison. The wife of the unfortunate victim, Lady Farrow, insists on seeing her husband, who she believes is still awaiting trial in the prison. Blackadder's scheme to extricate himself from this situation is to impersonate Lord Farrow at the meeting with his wife by wearing a bag over his head**.** Lord Percy has the job of explaining this to the unsuspecting lady**.**

1		Shot 1 CP: stationary/ HP: frontal/ VP: median/ D: MLS; VC: interior of the jail; Percy; Blackadder; Baldrick; Mr. Ploppy/ VS: the bag, exactly in the middle of the scene/ CO: artificial set; VF: distance: median; orientation: Blackadder's and Baldrick's gaze towards Percy **Kinesic Action**: Blackadder orders Percy out by shouting to him/ Tempo: M	{RG} [ ] Blackadder: (**)*Go on,* (NA)*go on*// Pause/ Volume: f/ Tempo: F	Sbrigati! Sbrigati!
5		↓ VF: orientation: Percy with closed eyes, avoiding Lady Farrow's gaze; Lady Farrow staring at him **Kinesic Action**: Percy turns his head and closes the door, keeping his left hand on the handle/ Tempo: M	{RG} [ ] Lord Percy: *Em* (#) (*)*sorry about the delay* (NA)*madam*// Pause/ Volume: n/ Tempo: M {RG} [ ] *eh* (#) *as you know* (#) *you're about to meet your* (NA)*husband, whom you'll recognise on account of the fact that* (#) *he has got a* (*)*bag over his head*// Pause/ Volume: n/ Tempo: M	Ehm,scusate il ritardo.Tra qualche istante potrete vedere vostro marito. Lo riconoscerete dal sacco in testa

Fig. 2: An example of a multimodal transcription

For reasons of space only two of the rows in the transcription of the subphase (the first and last 1-second frames) have been included. In the first row, Blackadder can be seen among his henchmen preparing to put the bag on his head. Lord Percy is nervously getting ready to meet Lady Farrow who is waiting outside (Row 5). Blending an interpersonal interpretation onto this ideational description, the viewer sees the participants from the same conspiratorial level. Indeed, the viewer has a better perspective than any of the characters in that he/she has an unhindered view of all of them as they are arrayed on the screen. Blackadder is recognised as the boss – he has central position and the others occupy the margin, to use Kress and van Leeuwen's terminology (1996: 206). From a textual point of view, the scene is the thematic element for the whole phase (Gregory, *in press*) covering the fateful meeting, and marks a cohesive element with the third sub-phase when Percy re-enters the room. The set is an obvious mock-up of a prison, the costumes instantly recognisable as Elizabethan period, from Percy's fancy ruff to the rags that Baldrick wears as a member of the lowest social order, the colours and lack of colour playing an important role. This already prepares the audience for the incongruous actions that are to follow, which are at the heart of all humour. The audience subconsciously knows that humour is based on this premise and generally makes every effort to make sense of the text somehow, however bizarre it may be. They are helped by their intertextual knowledge of similar texts they have previously been exposed to. Even a patchy and scholastic knowledge of Elizabethan England prepares the viewer for the setting, knowledge of past BBC comedies, the style of Rowan Atkinson, and indeed past Blackadder series, prepare him or her for the kind of parody that will take place. The foreign audience, however, may need a little more priming, especially pre-planning in the form of prior publicity, articles in other media, etc., but the basic mechanisms come into play just the same. Otherwise, how could we account for the massive popularity of Brazilian 'telenovelas' on Russian television?

To turn now to the question of the translation, the only thing that is said in this brief scene is Blackadder's impatient injunction to Percy – **Go on! Go on!** – and would thus not seem to tax the powers of the translator unduly. But conflicting pressures come to bear. In the interests of condensation, the obvious first step would be to remove the repetition, but this would overlook the importance of interpersonal elements; here the repeated order is designed to express Blackadder's contempt for Percy and intense irritation at Percy's constant incompetence. He almost snarls the words. So do we keep the repetition? At this point, the question of the audience arises. A minimum knowledge of the source language would equip any viewer with the necessary resources to interpret the text. And it is true that even those with no knowledge of the source language would still catch the aggressive intonation and the head movements expressing the feelings of the speaker. However, repetition of a word or short expression puts less pressure on the receptive capacities of a viewer than new material, and repeating the order would probably be the best option. This to-ing and fro-ing between competing solutions reflects the thought processes of the translator as various options are considered, a process well illustrated by Krings 'thinking aloud protocols' (Krings, 1987). But the problem still remains of what actual words to use. A literal translation into Italian would provide something like – **Avanti! Avanti!** – but if the interpersonal elements are to be integrated, namely the contempt and the irritation, then a version incorporating a fairly colloquial verb plus the second person singular intimate pronoun (expressing the superior to inferior relationship), might be preferred: **Sbrigati! Sbrigati!**

The time taken to discuss this first minimum utterance is an indication of how much thought is required to translate a film for subtitles, but also shows how the multimodal transcription enables the translator to focus his efforts. Proceeding in this vein, the analyst/translator/ adapter/subtitler (who may or may not be the same person) gets a very clear picture of how meaning is being

expressed and therefore to what extent s/he can intervene on the purely verbal element.

Analysing and subtitling large numbers of different kinds of texts, some of which purely for research purposes, others for student thesis preparation, others for language teaching modules, others still for genuine practical use, puts a large burden on computer storage facilities and databank management. Access to MCA allows the researcher/student to plug into a large selection of on-line filmed material which can be experimented with, in Trieste, without downloading material unless and until necessary. Secondly, in a reciprocal light, Trieste users can add to the stock of material on the MCA corpus which then becomes available to other members of the research team, the research community at large, and other selected participants.

This kind of symbiosis is already a reality within the *LINGUATEL* structure. In this way research into limitless genres and subgenres of video text can continue apace and at the same time feed back into the system material already analysed (even tagged) which can be used for other purposes. It is, of course, hoped to extend this service to all interested parties. The potentialities of the system have already far exceeded original expectations and are destined to produce ever more interesting avenues of use.

3. MCA in Pavia

3.1 Why we decided to build MCA

The multimodal transcription illustrated above and originally developed by Baldry in relation to the comparison of scenes from different medical texts (Baldry, 2000) and by Thibault in terms of a complete system for multimodal annotation of a bank advert (Thibault, 2000) has many limitations. Essentially a multimodal transcription is a static representation of something that is quintessentially dynamic, providing an *in vitro* frame-by-frame analysis of the component parts of a film. This is fine as far as it goes. But if we want to understand how films make meaning we need instead to develop instruments that examine texts in terms of an *in vivo* analysis treating them as if they were living objects which, as they unfold in time, present constantly changing patterns of semiotic selections. Dynamic texts need to be seen for what they are: a constant weaving and foregrounding of different constellations and integrations of meaning-making resources such as space, gesture, language, ambient sounds, music and gaze.

As Thibault (2000:320-321) points out, multimodal text analysis does not accept either in theory or in practice the notion that the meaning of the text can be divided into a number of separate semiotic 'channels' or 'codes'. The meaning of the text is the composite product/process of the ways in which different resources are co-deployed. A text can be segmented into a series of phases and transitions between phases. This will tell us how the selections of resources from different semiotic systems achieve a consistency of co-patterning. Phases, according to Thibault, are the enactment of the locally foregrounded selections of options which realise the meaning which is specific to a given phase of the text. Phases and subphases refer to salient local moments in the global development of the text as it unfolds in time. A given phase will be marked by a high level of metafunctional consistency or homogeneity among the selections from the various semiotic systems that comprise that particular phase in the text. Thibault also observes that the points of transition between phases have their own special features that play an important role in the ways in which observers or viewers recognise the shift from one phase to the next. Generally speaking, transition points are perceptually more salient in relation to the phases themselves. Thus, Thibault concludes, viewers of texts have no difficulty in perceiving particular textual phases thanks to their ability to recognise the transition points or boundaries between phases.

Of course, the multimodal transcription can be a useful starting point for an understanding of the ways in which resources such as gaze, gesture and language combine in typical phasal patterns. In the early stages of this work Baldry and Thibault developed a dynamic version of the static multimodal transcription, a forerunner of MCA, which allowed the user to generate the individual rows of a transcription through a query mechanism, and which facilitated understanding of how visual objects and their movements could be analysed in terms of Halliday's metafunctions (Halliday, 1994:38-144).

In an extension to their original conception Baldry and Thibault also devised a form of multimodal transcription that incorporated a multimodal tagging system based on Halliday's description of transitivity (Halliday, 1994: 101-144) but which also included the gestural semiotic (Baldry and Thibault: 2001:94-98). This kind of work can be particularly useful in understanding how (for example in a lecture) a speaker will typically use combinations of language, voice prosodics and gesture to express the point of view and/or circumstances of another person. All this helps us to understand how gesture and language combine to instantiate projection (Baldry and Thibault, 2000:96-98). This approach fits in with the notion of a specialised corpus highlighting specific kinds of textual phenomena such as projection, visual collocation, visual metaphor and so on within in multimodal approach to textuality.

But if we are to pursue our understanding of the codeployment of semiotic resources any further we need to understand how dynamic texts typically unfold in time and to ensure that this unfolding in time can be captured by *in vivo* rather than by *in vitro* multimodal analysis. In order to be able to identify typical patterns, the research process requires us to build corpora that can be analysed in terms of various textual phenomena, including in particular a study of the typical phasal organisation of a specific genre.

These then are the premises that led Baldry and Thibault to construct a corpus TV of car ads (currently 100) that a multimodal concordancer could help analyse. While sketching out some of the very preliminary results of this analysis (the corpus is still being constructed), we may describe the current organisation of MCA (the Beta test version released on 24.03.2002).

3.2 How it works

MCA is an XML-based multimodal concordancer whose user interface presents a series of rectangular green buttons (Fig. 3) which make up the *Projects Menu* and which represent the complete set of multimodal texts in the database.

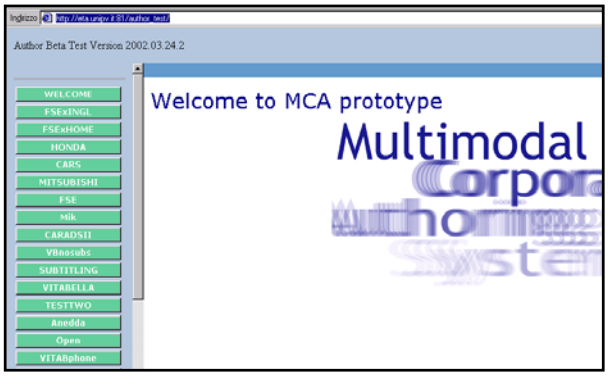


Fig. 3 Home page

When one is selected a specific project will open containing a video text. Using the scroll bar to browse through the *Projects Menu,* two light-blue buttons appear at the bottom of the list which respectively allow the user to create a *new* project in MCA or define *new* query-and-analysis parameters for use within the existing projects. Clearly, the system can be interpreted in one of two ways. While, on the one hand, it may be seen as an archive that can be consulted permitting detailed study of texts, on the other hand, MCA also represents a new area of distance work, given that users can enrich the relational database (which is the heart of MCA) with *their own* projects that are created and modified on-line and which the Server will immediately update and make available in Internet. When a particular MCA project is opened, a web page is loaded with a series of six light-blue buttons (grouped together on the left-hand side of the page) designed to 'handle' multimodal texts in ways described below.

Fig. 4: Analysis Inquiry and the Query Page

A user merely interested in viewing a film (such as the language learner posited in the *Introduction*) need only click on the last button, *Analysis Inquiry*, which will open up the *Query Page* and which when searched, using the *Parameters tool,* will return a film. In the example shown in Fig. 4, the query takes the form: *AD contains text 7*. This will automatically be opened and shown in *Windows Media Player* in the upper right-hand side of the web page. Clicking on the two leftmost *Media Player* buttons (Fig. 4), will, of course, stop and start the film. When we observe the open document, however, we quickly realise that construing MCA merely as a system for the reproduction of film texts would be reductive.

MCA in fact merges a relational database with streaming video technology, that allows specific sequences in a much longer film to be identified (and viewed) and to associate a description to each sequence. The basic idea is that the user can, in this way, consult a series of film sequences which share common characteristics. For example, a scholar concerned with an analysis of soap operas might want to find all the cases in which there is a dialogue between three, as opposed to two, speakers. Although in its current stage of development MCA is not able to show more than one film sequence at a time, the user can, as Fig. 5 indicates, nevertheless identify, with complete accuracy and certainty, all the cases sharing a particular feature, in the case in point all the Audi ads in the corpus.

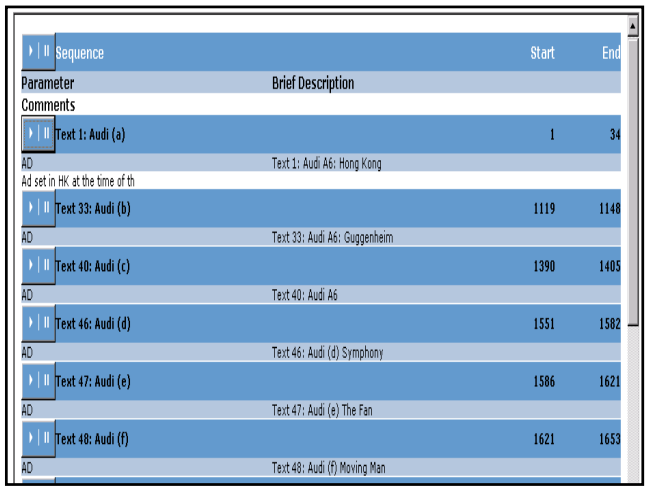


Fig. 5 Finding a subcorpus within the corpus

In this way, for example, the various sequences can be compared in such a way as to understand how body movements and gestures accompany these linguistic acts in characteristic patterns. To give one example (not shown here), we may care to analyse the way hand-and-arm movements are (as is always the case with gesture) crucially co-deployed with space in the construction of meaning. In a car ad for the New Mini, zombies appear from under the earth and prepare to lay their hands on a couple quietly kissing in a remote spot in their new car. The ad is constructed around the notion of space below and above ground (the zombies pop up from the world below) and inside and outside the car. A series of hand-arm movements are correlated to these notions creating the meaning, recurrent in contemporary European car ads, that the car is a place of safety. Indeed, the transition points in the advert coincide with the camera selecting parts of the car which divide the space in the Mini from the outside world (thereby reinforcing the message of safety and protection): the doors centrally locking from

inside, the front and rear windscreens. They also focus on a cohesive chain of hand movements: zombies outstretched hands (ready for attack), couple's hands raised upwards in fear and defence; driver's hands yawning; one zombie's hand replacing the windscreen wipers and patting the windscreen in a gesture of reassurance and leave-taking. All this is created in tandem with language which takes the form of an off-screen narrator (a voiceover) commenting on the zombies failed attack on the car.

In order for the user to be able to carry out such analyses and to make comparisons with other texts, preliminary operations (segmentation, indexing and tagging) need to be carried out. In particular, the user-author must:

- define a new project associating a film to MCA's descriptive tools (*Project Definition*)
- select the parameters used to tag and describe individual sequences (*Parameters Selection*)
- break the film up virtually into various sequences (*Video Indexing*)
- describe the characteristics of the individual sequences (*Sequence Analysis*) with a view to obtaining finely detailed information when queries are made.

When this has be done the final tool – *Analysis Inquiry* – can be used to produce the results shown in the various figures.

The user who wishes to create a new project can do so by clicking on the *New Project* button but can also modify an existing one (having selected it from MCA's home page). Creation and modification require completion (or redefinition) of the *Project Definition* menu. It is in this phase that the researcher associates a film (previously converted to the **.wmv* format) with the project. Input and output film are the same in MCA. In fact the film remains in its original form. All the work of segmentation, description tagging and retrieval is carried out within the relational database and associated tools.

When this first phase has been completed the parameters relevant to the research project need to be selected through the *Parameters Selection* menu. In the case that appropriate parameters are not available they can be added via a page accessible through the *Parameters Definition* menu. The list of parameters selected can be seen in *Sequence Analysis* page, through which a detailed description of the video text can be made. But before tagging and describing data, the film needs to be split up *virtually* into sequences. This operation is carried out in *Video Indexing*. The research and development cycle is completed with the use of *Analysis Inquiry,* from whose menu various queries and comparisons can be made.

4. Discussion

Although the system is relatively simple to use, nevertheless like any other software program, the MCA system is the result of specific design work which allows a limited degree of flexibility, on the one hand, but, on the other, allows the user to carry out investigations at a speed which would be hard to achieve by other means or which would be so demanding as not to be worth the candle. Thus while the system requires the use of Microsoft Internet Explorer browser and preferably release 5.5 or higher and a suitably updated release of Windows Media Player, on the other hand, the user is spared the constant need to wind backwards and forwards as is the case with video-cassettes or the wasteful dead-times associated with transferring, reproducing and downloading film.

Moreover, the user/author is able not only to play movie samples but also to add further annotations, providing he/she is authorised to do so (a system of authorisations and passwords is currently being added). Thus two or more researchers working in different locations can work on the description and/or tagging of the same corpus.

In one year's use many initial difficulties have been overcome, the system functioning reliably and responding to requirements for which it was not originally designed

Originally, conceived as an instrument to support research it has proved to be a useful means for teaching and on-line thesis preparation, the user base now including the following categories:

- the researcher who wishes to carry out his/her own work using MCA
- the teacher who needs to hold a language lesson supported by multimedia files
- the student who wishes to follow a self-access language learning course from his/her own home
- the thesis student who must carry out multimodal descriptions of texts
- the user who wishes to show the results of his/her research inserting them into a database.

For each of these user categories, specific needs have emerged which have required further work on the system so as to update it periodically to meet new demands. Feedback from users, who have required help in overcoming problems relating to minimum system requirements, has enabled us to perfect the film-coding technique, in such a way as to optimise the connection via modem and via LAN, avoiding, for example, lack of synchronisation between audio and image in the streaming video. A particular note needs to be made relating to the creation of new materials with subtitles, given the difficulties that users have encountered. For this reason the suggestion has been made that it would be appropriate to introduce a "text box" below the *Media Player* area in future MCA prototypes in which to introduce the subtitle corresponding to the sequence shown, which would be memorised in the database by means of a procedure that would extend the virtual approach adopted in MCA (i.e. subtitles would appear to be printed on the film, whereas in fact they are generated separately from the film).

Like any prototype MCA needs to be improved on and a fully-fledged second prototype is under production which, in addition to what has been outlined above, will introduce account-based security and privacy features respecting different user needs and user typologies more fully and introducing appropriate customisations. On the basis of the experience so far acquired, which has indicated a wider user base than at first expected, we can assume that other user categories, including institutional

users such as Language Centres and Libraries, will make use of MCA for the conversion/distribution of analogical or paper-based data, which may lead to the identification of new criteria for use of MCA. To date, most users have been closely associated only with Italian Universities, and mostly with the University of Pavia. Nevertheless, it has been exciting to follow the progress of graduating students who have used the system as an integral part of their graduation theses. We can therefore expect a growth in the number of graduating and postgraduate students who will use this system and are thus actively seeking inter-University ties and inter-University development projects that will help stimulate this goal.

5. Conclusion

Born in the text linguistics sector, MCA is an instrument for analysing dynamic multimodal texts, i.e. film and video texts which, as they unfold in time, display different and constantly varying constellations of sound, image, gesture, text and language (Baldry, 2000, Thibault 2000). Much of this work has already been reported elsewhere but this paper has described a new version of MCA as well as some of the results of one year's use of the tool. The growth in MCA's user base is evidence, apart from the growing interest in the description of multimodal texts, of the desire to learn about the potential and characteristics of this instrument, (including, of course, the need to understand how it works). Designed initially as a support for researchers dealing with the multimodal text analyses of texts, and specifically to provide them with the possibility of examining and comparing multiple contexts and texts in real time, it has proved a useful self-access distance language-learning and text analysis tool, since it provides students with the possibility of listening to, and watching, film clips, that are played and stopped at will. But the system has not yet benefited from critical comparison, one reason why we have decided to present it in various congresses. MCA has been built in virtual isolation vis-à-vis other systems and, in this respect, needs to grow considerably.

6. References

Baldry, A. P. (2000) 'Introduction', *Multimodality and multimediality in the distance learning age*, ed. A. P. Baldry, Campobasso: Palladino Editore, pp. 11-39.

Baldry, A.P. and Thibault, P.J. (2001) 'Towards multimodal corpora' in *Corpora in the description and teaching of English*, G. Aston and L. Burnard, eds. Bologna: CLUEB, pp. 87-102

Gottlieb, H (1997) *Subtitles, Translation and Idioms,* Copenhagen: Department of English, University of Copenhagen

Gregory, M (*in press*) 'Phasal analysis within communication linguistics: two contrastive discourses'. Relations and functions in language and discourse. P. Fries, M. Cummings, D. Lockwood and W. Sprueill, eds New York & London: Continuum

Halliday, M.A.K. (1994[1985] *Introduction to Functional Grammar*. Second Edition. London and Melbourne: Arnold.

Kress, G. and van Leeuwen, Th. (1996*) Reading Images. The Grammar of Visual Design*. London and New York: Routledge.

Krings H.P., 1987. The Use of Introspective Data in Translation, in Faerch & Kasper 1987: 159-176.

Taylor, C. (2000) 'The subtitling of film; reaching another community' in *Discourse and community; doing functional linguistics,* E. Ventola, ed. (Tübingen: Gunter Narr Verlag, , pp. 309-327.

Taylor, C. and Baldry, A. P. (2001) 'Computer assisted text analysis and translation: a functional approach in the analysis and translation of advertising texts' in *Exploring translation and multilingual text production: beyond content*, ed. E. Steiner and C. Yallop, (Berlin: Mouton de Gruyter, pp. 277-305).

Thibault, P.J. (2000) The multimodal transcription of a television advertisement: theory and practice. In *Multimodality and multimediality in the distance learning age*, ed. A. P. Baldry (pp. 311-38). Campobasso: Palladino Editore.

The Observer® Video-Pro: a Versatile Tool for the Collection and Analysis of Multimodal Behavioral Data

Niels Cadée, Erik Meyer, Hans Theuws & Lucas Noldus

Noldus Information Technology bv
P.O. Box 268, 6700 AG Wageningen, The Netherlands
n.cadee@noldus.nl, http://www.noldus.com

Abstract

The Observer is a professional tool for the collection, management, analysis and presentation of observational data. The user can record activities, postures, movements, positions, social interactions or any other aspect of behavior. The Observer can be used either for live scoring, or for scoring from analog or digital video material. With The Observer's generic configuration utility, detailed coding schemes can be designed for observing hand gestures, body postures, and facial expression. In the current version of the software, speech transcription is supported through free-format comments with time stamps. Improvements in the area of speech annotation are currently under way in the framework of the EU-funded NITE project. This includes more advanced coding schemes for speech annotation, as well as interfacing with other linguistic data collection and analysis tools via XML.

1. Introduction

The Observer (Noldus et al., 2000) has originally been developed as a tool to support observational studies in ethology. However, over the years it has become clear that the generic nature of the data collection and analysis functions of The Observer make it suitable for almost any observational study. The Observer is currently in use at thousands of universities, research institutes and industrial laboratories worldwide. Applications are found in a wide range of disciplines, including psychology, psychiatry, human factors and ergonomics, usability testing, industrial engineering, labor and time studies, sports research, consumer behavior and market research. Recently, we have noticed an increasing interest from researchers in the area of multimodality and speech annotation. We are in the process of extending our software to cater for the specific demands of researchers in this field, through our active collaboration in the EU-funded NITE project (Natural Interactivity Tools Engineering, http://nite.nis.sdu.dk/). Some research groups are already using The Observer to study, for example, turn taking in dialogues between two persons, verbal and non-verbal communication between mothers and toddlers, and for making a dictionary of everyday gestures.

2. Program features

The Observer can be used either for live scoring (Basic version), or scoring from analog or digital video material (Video-Pro version). The Observer can control a video recorder from the pc, and by use of a video overlay board, display the video image of a tape on the computer screen, within the program (Fig. 1). Digital video files can be played directly within The Observer. There is a direct coupling between time code in the data files with scored annotations and the video material. This allows for accurate scoring, even when playing the video at slow or fast speeds. Advanced search functions allow the user to find particular events or time stamps on the videotape or media file. A search for events is always based on elements from the coding scheme. A special highlights function allows the user to select specific episodes based on the scored events, and make analog or digital video clips for presentation purposes.

3. Program demonstration

During the workshop, The Observer 4.0, the latest release, will be demonstrated. Compared to previous editions, version 4 features improved usability, especially for design of the coding scheme. Data selection has been completely redesigned, and allows for the most complex filtering of annotation results. For example, one can define time intervals of variable length based on actual scored events, to answer questions like 'How often did Peter grin between the time when John entered the room, and the time when John left the room again?' Finally, The Observer 4.0 has an intuitive new layout that shows projects and their content in a tree view (Fig. 2).

4. Designing coding schemes

With The Observer's generic configuration utility, detailed coding schemes can be designed for observing hand gestures, body postures, and facial expression. Gaze can be scored manually, or recorded with additional eye-tracking equipment. The coding scheme is based on behavioral classes, which each contain a set of mutually exclusive behaviors. In a simple example, you can have one behavioral class with hand gestures, and another class with types of speech. Aggressive and normal speech could be two mutually exclusive behaviors in the speech class, while pointing and waving could both be in the hand gesture class. A pointing gesture in the hand gesture class can be scored at the same time as aggressive speech in the speech class, but it could also be scored during normal speech. The user can further detail the coding scheme by attaching one or two modifiers to the behaviors. These can indicate for example the intensity of a behavior, or the person or object the behavior is aimed at. For example, for a pointing gesture, you can also score the object that the person is pointing at.

5. Speech annotation

Speech and other audio signals can also be annotated with The Observer. The current form of speech transcription is as free-format comments with time stamps. We are working on improvement in the speech area through participation in the NITE project. We are currently looking into options for XML export, and for allowing more structure for speech annotation. The

Observer and prototypes of other software developed in this project will be shown in a demonstration session during the LREC conference.

6. Data analysis

The Observer has extensive features for data analysis. The user can filter the data with the data selection function, and for example select variable time intervals based on scored events. Data can be visually examined in tables and plots of events against time (Fig. 2). A range of elementary statistics can be calculated. Reliability analysis is another important feature, where users can check the consistency of several different people's annotations of the same video material. With lag sequential analysis, temporal relations and patterns can be discerned.

7. References

Noldus, L.P.J.J., Trienes, R.J.H., Hendriksen, A.H.M., Jansen H., & Jansen, R.G. (2000). The Observer Video-Pro: new software for the collection, management, and presentation of time-structured data from videotapes and digital media files. *Behavior Research Methods, Instruments & Computers* **32**, 197-206.

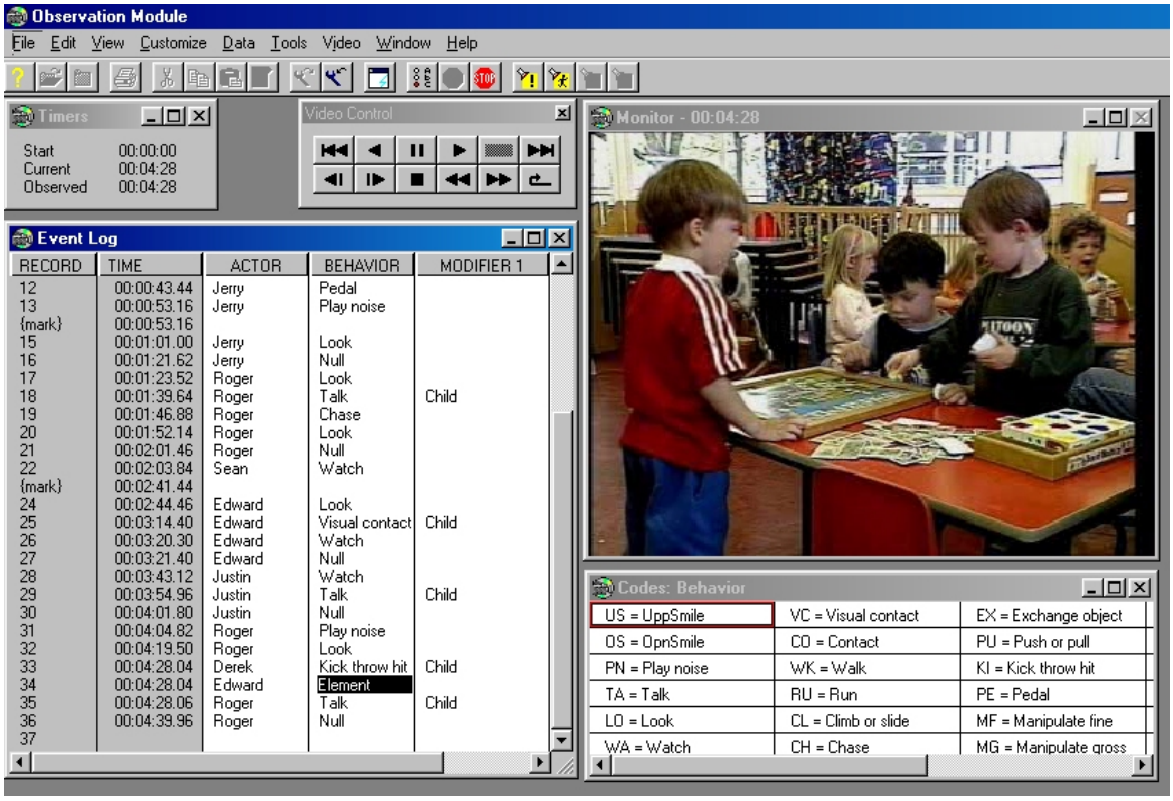


Figure 1: The Observation Module of The Observer. This example shows a project on children's playing behavior. The user can customize the size and position of the windows, and select which ones to display. In this case, the screen shows the codes for annotation, the event log with the data file with times and scored events, the video image, video controls, and timers.

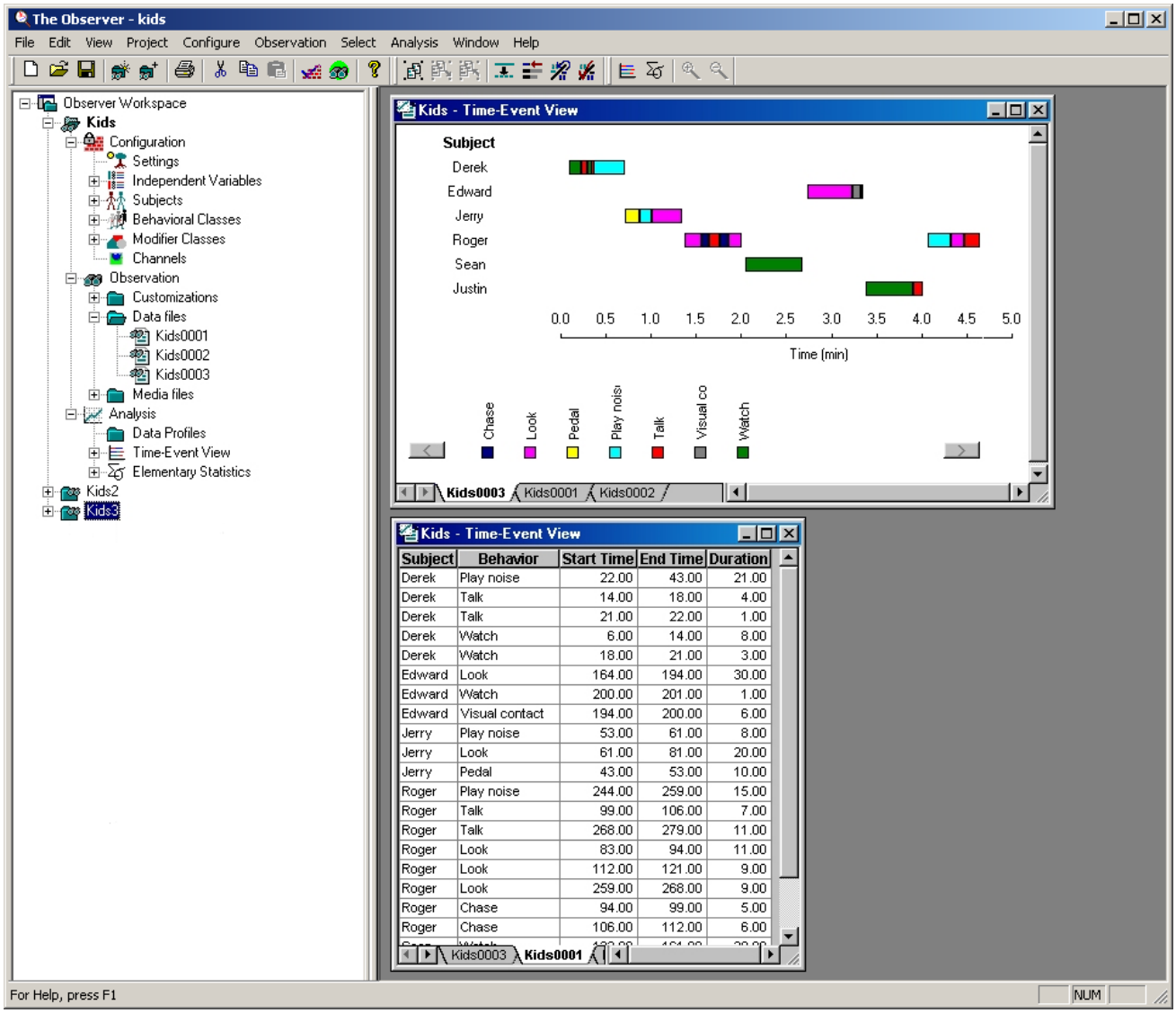


Figure 2: The Main Module of The Observer. In the Explorer view on the left side, a workspace with three projects is shown. One project (on children's playing behavior) is expanded to show its contents. The Configuration contains all settings of the coding scheme. On the right side of the screen, two types of analysis results are shown: a time-event plot and a time-event table.

Prosody Based Co-analysis of Deictic Gestures and Speech in Weather Narration Broadcast

Sanshzar Kettebekov, Mohammed Yeasin, Nils Krahnstoever, Rajeev Sharma

Department of Computer Science and Engineering
Pennsylvania State University
220 Pond Laboratory
University Park, PA 16802,USA
[kettebek; yeasin; krahnsto; rsharma]@cse.psu.edu

Abstract

Although speech and gesture recognition has been studied extensively all the successful attempts of combining them in the unified framework were semantically motivated, e.g., keyword co-occurrence. Such formulations inherited the complexity of natural language processing. This paper presents a statistical approach that uses physiological phenomenon of gesture and speech production process for improving accuracy of automatic segmentation of continuous deictic gestures. The prosodic features from the speech signal were co-analyzed with the visual signal to create a statistical model of co-occurrence with particular kinematical phases of gestures. Results indicated that the above co-analysis improves continuous gesture recognition. The efficacy of the proposed approach was demonstrated on a large database collected from the weather channel broadcast. This formulation opens new avenues for bottom-up frameworks of multimodal integration.

1. Introduction

In combination, gesture and speech constitute the most important modalities in human-to-human communication. People use large variety of gestures either to convey what cannot always be expressed using speech only or to add expressiveness to the communication. Motivated by this, there has been a considerable interest in incorporating both gestures and speech as the means for Human-Computer Interaction (HCI).

To date, speech and gesture recognition have been studied extensively but most of the attempts at combining them in an interface were in the form of a predefined signs and controlled syntax such as "*put <point> that <point> there*", e.g., (Bolt, 1980). Part of the reason for the slow progress in multimodal HCI is the lack of available sensing technology that would allow non-invasive acquisition of natural behavior. However, the availability of abundant processing power has contributed to making computer vision based continuous gesture recognition in real time to allow the inclusion of natural gesticulation in a multimodal interface (Kettebekov and Sharma, 2001, Pavlovic et al., 1997, Sharma et al., 2000).

State of the art in *continuous gesture recognition* is far from meeting the requirements of a multimodal HCI due to poor recognition rates. Co-analysis of visual gesture and speech signals provide an attractive prospect of improving continuous gesture recognition. However, lack of fundamental understanding of speech/gesture production mechanism restricted implementation of the multimodal integration at the semantic level, e.g. (Kettebekov and Sharma, 2001, Oviatt, 1996, Sharma et al., 2000). Previously, we showed somewhat significant improvement in co-verbal gesture recognition when those were co-analyzed with keywords (Sharma et al., 2000). However, the implications of using a top-down approach has augmented challenges with those of natural language and gesture interpretation and made automatic processing challenging.

The goal of the present work is to investigate co-occurrence of speech and gesture as applied to continuous gesture recognition from a bottom-up perspective. Instead of keywords, we employ a set of prosodic features from speech that correlate with deictic gestures. We address the general problem in multimodal HCI research, e.g., availability of valid data, by using narration sequences from the weather channel TV broadcast. The paper is organized as follows. First, a brief overview of the types of gestures that occur in the analysis domain is presented. The synchronization hierarchy of gestures and speech is also reviewed. In section 3 we discuss a computational framework for continuous gesture acquisition using a segmental approach. Section 4 presents a statistical method for correlating visual and speech signals. There, acoustically prominent segments are detected and aligned with segmented gesture phases. Finally, results are discussed within the framework for continuous gesture recognition.

2. Co-verbal Gesticulation for HCI

McNeill (1992) distinguishes four major types of gestures by their relationship to the speech. *Deictic* gestures are used to direct a listener's attention to a physical reference in course of a conversation. These gestures, mostly limited to the pointing, were found to be co-verbal, cf. (McNeill, 1992). From our previous studies, in the computerized map domain (***i*MAP,** see Figure 1) (Kettebekov and Sharma, 2000), over 93% of deictic gestures were observed to co-occur with spoken nouns, pronouns, and spatial adverbials.

Iconic and *metaphoric* gestures are associated with abstract ideas, mostly peculiar to subjective notions of an individual. *Beats* serve as gestural marks of speech pace. In the weather channel broadcast the last three categories roughly constitute 20% of all the gestures exhibited by the narrators. We limit our current study to the *deictic* gestures for a couple of reasons. First, they are they are more suitable for manipulation of a large display, which becomes more common for HCI applications. Second, this

type of gestures exhibits relatively close coupling with speech.

2.1. Gesture and Speech Production

The issue of how gestures and speech relate in time is critical for understanding the system that includes gesture and speech as part of a multimodal expression. McNeill (1992) distinguishes three levels of speech and gesture synchronization: semantic, phonological, and pragmatic. The pragmatic level synchrony is common for metaphoric and iconic gestures and therefore is beyond the scope of the present work.

Semantic synchrony rule states that speech and gestures cover the same idea unit supplying complementary information when they occur synchronously. The current state of HCI research provides partial evidence to this proposition. Previous co-occurrence analysis of weather narration (Sharma et al., 2000) revealed that approximately 85% of the time when any meaningful gestures are made, it is accompanied by a spoken keyword mostly temporally aligned during and after the gesture. Similar findings were shown in the pen-voice studies (Oviatt et al., 1997). The implication of the semantic level synchronization rule was successfully applied at the keyword level co-occurrence in the previous weather narration study (Sharma et al., 2000).

At the phonological level, Kendon (1990) found that different levels of movement hierarchy are functionally distinct in that they synchronize with different levels of prosodic structuring of the discourse in speech. For example, the peaking effort in a gesture was found to precede or end at the phonological peak syllable (Kendon, 1980). These findings imply a necessity for viewing a continuous hand movement as a sequence of kinematically different segments of gestures. This approach is reflected in the next section. Issue of using the phonological peak syllables is associated with the complexity of the nature of the tonal correlates, e.g., pitch of the voice. Pitch accent, which can be specified as low or high, is thought to reflect a phonological structure in addition to the tonal discourse, cf. (Beckman et al., 1992). We address this issue by proposing a set of correlate point features in the pitch contour that can be associated with the points on the velocity and acceleration contours of the moving hand (section 4).

3. Gesture Acquisition

Building human computer interfaces that can use gestures involves challenges that range from low-level signal processing to high-level interpretation. A wide variety of methods had been introduced to create gesture driven interfaces. With the advances in technology there has been a growing interest in using vision-based methods (Pavlovic et al., 1997). The advantage of these is in their non-invasive nature. The idea of a natural interface comes from striving to make HCI as close as communicating in ways we are accustomed to. Vision-based implementation therefore can be very useful for a natural interface.

One could expect that the meaning encoded in multimodal communication is somehow distributed across speech and gesture modalities. A number of recent implementations used predefined gesture syntax, e.g., (Oviatt, 1996). A user is confined to the predefined gestures for spatial browsing and information querying. As a result, a rigid syntax is artificially imposed. Therefore the intent of making interaction natural is defeated. However, with imprecise recognition of non-predefined gestures, it may be harder to argue for replacing more precise HCI devices, e.g., electronic pen with fixed predefined functions.

The key problem in building such interface, e.g., using statistical techniques, is the lack of existing *natural* multimodal data. Studies from human-to-human communication do not automatically transfer over to HCI due to artificially imposed paradigms. This controversy leads to a "chicken-and-egg" problem.

While the use of the weather narration domain as a bootstrapping analysis offers virtually unlimited bimodal data it can be assumed as a reasonable simplification of an HCI domain. In the series of the previous studies we employed the weather narration broadcast analysis (Sharma et al., 2000) to bootstrap ***i*MAP** framework (Figure 1) (Kettebekov and Sharma, 2001). It showed that the gesticulative acts used in both domain have similar kinematical structure as well as gesture and keyword co-occurrence patterns. However, the key aspect for choosing the weather domain for the current study is in a possibility of applying simple processing techniques for extraction of prosodic information from uninterrupted narration.

Figure 1. ***i*MAP** testbed in the context of a computerized map. The cursor is shown within the circle.

Over 60 minutes of the selected weather narration data was used in the analysis. The video sequences contained uninterrupted monologue of 1-2 minutes in length. The subject pool was presented by 5 men and 3 women.

3.1. Kinematics of Continuous Gestures

A continuous hand gesture consists of a series of qualitatively different kinematical phases such as movement to a position, hold, and transitional movement. We adopt Kendon's framework (Kendon, 1990) by organizing these into a hierarchical structure. He proposed a notion of gestural unit (*phrase*) that starts at the moment when a limb is lifted away from the body and ends when the limb moves back to the resting position. The *stroke* is distinguished by a peaking effort and it is thought to constitute the meaning of a gesture (Kendon, 1990). After extensive analysis of gestures in weather narration and *i*MAP (Kettebekov and Sharma, 2001, Sharma et al., 2000) we consider following strokes: *contour*, *point*, and *circle*.

Kita (1997) suggested that a *post-stroke hold* was a way to temporally extend a single movement stroke so that the *stroke* and *post-stroke hold* together will

synchronize with the co-expressive portion of the speech. It is thought that a *pre-stroke hold* is a period in which gesture waits for speech to establish cohesion so that the stroke co-occurs with the co-expressive portion of the speech. Therefore, in addition to our previous definitions we also include *hold* as a functional primitive.

3.2. Continuous Gesture Segmentation

Sixty minutes of weather domain gesture data for training and testing was collected from broadcast video using a semi-automatic gesture analysis tool (GAT) (see Figure 2). The tool provides a convenient user interface for rapid and consistent collection of positional data and a easily configurable set of pattern classification tools. GAT is integrated with PRAAT software for phonetics research (Boersma and Weenink, 2002) for speech processing and visualization.

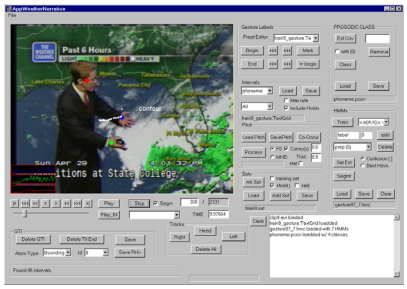


Figure 2. Gesture analysis tool (GAT) interface

The task of positional data ground truthing involves initialization the head and hand tracking algorithms (described in 2.3.1) at the beginning of each video sequence and in the events of self-occlusions of the hands.

3.2.1. Motion Tracking

The algorithm for visual tracking of the head and hands is based on motion and skin-color cues that are fused in a probabilistic framework. For each frame and each tracked body part, a number of candidate body part locations are generated within a window defined by the location of the body part in the previous frame and the current estimate of the predicted motion. The true trajectories of the body parts are defined as the most probable paths through time connecting candidate body part locations. The Viterbi algorithm is used to efficiently determine this path over time. This approach effectively models the hand and head regions as skin-colored moving blobs (Figure 3).

3.2.2. Kinematical Analysis

To model the gestures, both spatial and temporal characteristics of the hand gestures (phonemes) were considered. The time series patterns of gesture phases can be viewed as a combination of ballistic and guided motion of the hand reflected on the skewedness of the velocity profile. In the current study, a gesture phoneme is defined as a stochastic process of 2D positional and time differential parameters of the hand and head over a suitably defined time interval.

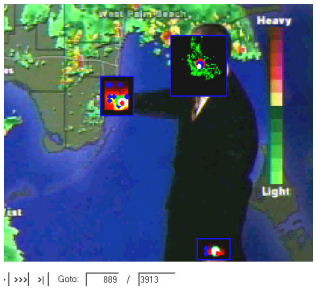


Figure 3. Semi-automatic ground truthing process employing a tracking algorithm;

A Hidden Markov Model (HMM) framework was employed for continuous gesture recognition, as described in (Sharma et al., 2000). The total of 446 phoneme examples extracted from the segmented training video footage were used for HMM training. The results of the continuous gesture recognition showed that only 74.2 % of 1876 were classified correctly. Further analysis indicated that phoneme pairs of preparation-pointing and contour-retraction constitute most of the substitution errors. This type of error, which can be attributed to the similarity of the velocity profiles, was accounted for the total of 33% of all the errors. The deletion[1] errors were mostly due a relatively small displacement of the hand during a pointing gesture. Those constituted approximately 58% of all the errors.

Although purpose of this work was not to introduce a robust algorithm with a high recognition rate there is an inherent limitation with the current acquisition method. I.e., 2D projected motion data can potentially introduce spurious variabilities that can have a detrimental effect on the recognition rate. The gesture model is based on the observed end-effector motion of the hands and the motion of the head projected into the camera plane and is only and indirect measurement of the true body

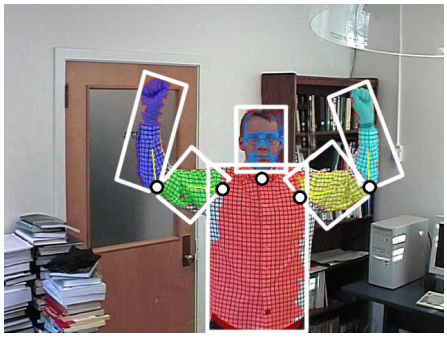


Figure 4. Model based tracking for future extraction of direct kinematical gesture parameters.

[1] Deletion type of errors occur when a gesture phoneme is recognized as part of another adjacent gesture.

kinematics. This observation model can hence introduces distortions and additional spurious variabilities that complicate the differentiation between gestures. Current work in progress, cf. (Krahnstoever et al., 2002), has the goal of visually extracting the true 3D kinematical parameters such as body pose and angles of the shoulder and arm joints (see Figure 4).

4. Prosody Based Co-analysis

Both psycholinguistic, e.g., (McNeill, 1992), and HCI, e.g., ***i*MAP** (Kettebekov and Sharma, 2000), studies suggest that deictic gestures do not exhibit one-to-one mapping of form to meaning. Previously, we showed that the semantic categories of strokes (derived through the set of keywords), not the gesture phonemes, correlate with the temporal alignment of keywords, cf. (Kettebekov and Sharma, 2000). This work distinguishes two types of gestures: referring to a static point on the map and to a moving object (i.e., moving precipitation front). Due to the homogeneity of the context and trained narrators in the weather domain we can statistically assume (mismatch <2%) that pointing gesture is the most likely to refer to the static and contour stroke to the moving objects. Therefore, for simplicity we will use *contour* and *point* definitions.

The purpose of the current analysis is to establish a framework by identifying correlate features in visual and acoustic signals. First we will separate acoustically prominent segments. A segment is defined as a voiced interval on the pitch contour that phonologically can vary from a single phone/foot[2] to intonational phrase units, see (Beckman, 1996) for details. Then we will analyze alignment of the prominent segment with the gesture phonemes. This framework was implemented in GAT.

4.1. Detecting Prosodically Prominent Segments

Pitch accent association in English underlines the discourse-related notion of focus of information. Fundamental frequency (F_0) is the correlate of pitch defined as the time between two successive glottis closures (Hess, 1983). We employed PRAAT software to extract F_0 contour, as described in (Boersma, 1993).

Prominent segments were defined as segments which were relatively accentuated (or perceived as such) from the rest of the monologue. We considered combination of the pitch accent and the pause before each voiced segment to detect abnormalities in spoken discourse. Maximum and minimum of F_0 contour represent features for high pitch and low pitch accents. Maximum gradient of the pitch slope was also considered. A statistical model of prosodic discourse for each narration sequence was created (Figure 5), see (Kettebekov et al., 2002) for details.

To find an appropriate level of threshold to detect prominent segments we employed a bootstrapping technique involving a perceptual study. A control sample set for every narrator was labeled by 3 naïve coders for auditory prominence. The coders had access only to the wave form of speech signal. The task was to identify at least one acoustically prominent sound within the window of 3 seconds. The moving window approach was considered to account for abnormally elongated pauses in the spoken discourse. Allowing 2% of misses, the threshold was experimentally set for each narrator (Figure 5). If a segment appeared to pass the threshold value it was considered for co-occurrence analysis with the associated gesture.

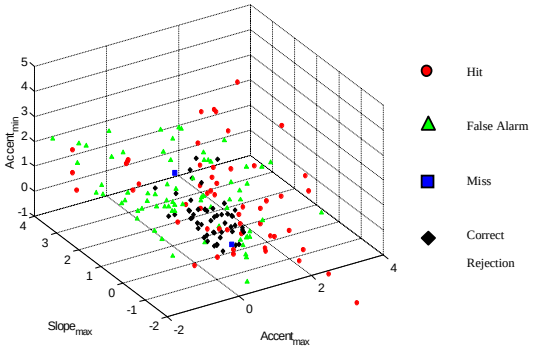


Figure 5. A sample distribution of auditory prominence for a female narrator with the decision boundary from the perceptual study.

4.2. Co-occurrence Models

A statistical model of the temporal alignment of active hand velocity and a set of features of the prominent pitch segments was created for every gesture phoneme class (Figure 6). The features on the pitch profile included max, min, beginning, and max of derivative of F_0, see (Kettebekov et al., 2002) for details. Present formulation

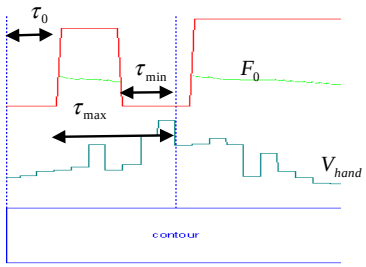


Figure 6. A set of features used for co-occurrence modeling of the hand velocity (V_{hand}) and a pitch (F_0) segment. Red contour represents prominence level of corresponding segments

accounts for the two levels of possible prosodic co-occurrence: discourse and phonological. The onset between a gesture and the beginning of a prominent segment is to model discourse cohesion (pauses). The onset of the peaks in the F_0 and peaks in the velocity profile of the hand addresses phonological level synchronization. All of 446 phonemes that have been used for training gesture phonemes were utilized for training of the co-occurrence models. Analysis of the resulted models indicated that there was no significant difference between *retraction* and *preparation* phases. Peaks of *contour* strokes tend closely to coincide with the peaks of the pitch

[2] Foot is a phonological unit that has a "heavy" syllable followed by a "light" syllable(s).

segments. *Pointing* appeared to be quite silent, however, most of the segments were aligned with the beginning of the *post-stroke hold* interval.

Figure 7 summarizes findings of the co-analysis framework. At the first level we separate co-verbally meaningful gestures (*strokes*) from *auxiliary* phonemes that included *preparation* and *retraction* phases. Also, we exclude strokes that are re-articulate previous gestures such as a stroke can be followed by the identical stroke where the second movement does not have associated speech segment. At the second level co-verbal strokes can be further classified according to their deixis, cf. (Kettebekov and Sharma, 2001). As it was noted before, in the context of the weather narration we can statistically consider those to be represented by *point* and *contour* phonemes without further definitions. *Preparation* and *retraction* phases were eventually collapsed into the same category and were not differentiated.

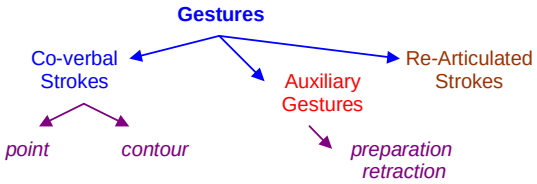


Figure 7. Prosodic co-analysis framework

The co-analysis models for co-verbal strokes were merged with the beginning of the post stroke-*hold* phases for classification purposes. Such redefinition of the co-verbal strokes for the purpose of co-analysis was motivated by the results associated with the *pointing* strokes and it was included into the computational framework.

4.3. Continuous Gesture Recognition with Co-occurrence Models

We employed Bayesian formulation to fuse the gesture framework and the co-occurrence models at the decision level, see (Kettebekov et al., 2002). The resulted segmentation showed significant improvement in the overall performance with the correct recognition of 81.8% (versus 72.4%). Subsequently, there was a significant reduction of deletion (8.6% versus 16.1%) and substitution errors (5.8% versus 9.2%). The deletion type of errors were minimized due to the inclusion of small point gestures, which are quite salient when correlated with prominent acoustic features. Figure 8 shows example of elimination of a deletion error after applying co-analysis. White trace on the figure illustrates visually negligible hand movement trajectory. Improvement of substitution errors can be attributed to the differentiation between the auxiliary gesture phases and the strokes in the co-occurrence analysis.

5. Conclusions

We presented an alternative approach for combining gesture and speech signals from the bottom-up perspective. Unlike commonly controlled gesture

a) b)

Figure 8. Example of deletion error using: a) visual-only signal resulted in *hold* gesture; b) with co-occurrence model *point* was recognized as a part of preceding *hold* (case a.);

recognition domains, we address this problem in the weather broadcast domain, which can be characterized by relatively unrestricted narration. Such formulation is more favorable for automated recognition of continuous deictic gestures then the semantic based (keyword co-occurrence). The current results demonstrate the concept of improving recognition of co-verbal gestures when combined with the prosodic features in speech. This is a first attempt which requires further improvement. The issues of portability to an HCI setting, e.g., ***i*MAP** framework, are currently under investigation.

Applicability of the current formulation for the other types of gestures is probably possible if the segmental approach is considered for the gesture acquisition. In a domain with more spontaneous behavior, e.g., in a dialogue (e.g., ***i*MAP**) (versus monologue as presented in the present work) the methodology of prosodically prominent feature extraction is more complex. It would require acquisition of an improved kinematical model (see section 3.2.2.) that considers additional visual cues such as turn of head (direction of the gaze), and etc.

6. Acknowledgements

The financial support of this work in part by the National Science Foundation CAREER Grant IIS-97-33644 and NSF IIS-0081935 is gratefully acknowledged. We thank Ryan Poore for his help with the data processing and implementation.

7. References

Beckman, M. E., Dejong, K., Jun, S. A., and Lee, S. H. 1992. The Interaction of Coarticulation and Prosody in Sound Change [JAN-JUN]. *Language and Speech* 35:45-58.

Beckman, M. E. 1996. The parsing of prosody [FEB-APR]. *Language and Cognitive Processes* 11:17-67.

Boersma, P. 1993. Accurate short-term analysis of the fundamental frequency and the harmonics-to-noise ratio of a sampled sound. Paper presented at *Institute of Phonetic Sciences of the University of Amsterdam*.

Boersma, P., and Weenink, D. 2002. PRAAT. Amsterdam, NL: Institute of Phonetic Sciences. University of Amsterdam, NL.

Bolt, R.A. 1980. Put-that-there: Voice and gesture at the graphic interface. *In SIGGRAPH-Computer Graphics*.

Hess, W. 1983. Pitch Determination of Speech Signals. In *Springer Series of Information Sciences*. Berlin: Springer-Verlag.

Kendon, A. 1980. Gesticulation and speech: Two aspects of the process of the utterance. In *The relation between verbal and non-verbal communication*, ed. M.R. Key, 207-227. Hague: Mouton.

Kendon, A. 1990. *Conducting Interaction*: Cambridge: Cambridge University Press.

Kettebekov, S., and Sharma, R. 2000. Understanding gestures in multimodal human computer interaction. *International Journal on Artificial Intelligence Tools* 9:205-224.

Kettebekov, S., and Sharma, R. 2001. Toward Natural Gesture/Speech Control of a Large Display. In *Engineering for Human Computer Interaction*, eds. M.R. Little and L. Nigay, 133-146. Berlin Heidelberg New York: Springer Verlag.

Kettebekov, S., Yeasin, M., and Sharma, R. 2002. Prosody based co-analysis for continuous recognition of co-verbal gestures, submitted to ICME'02.

Kita, S., Gijn, I.V., and Hulst, H.V. 1997. Movement phases in signs and co-speech gestures, and their transcription by human coders. Paper presented at *Intl. Gesture Workshop.*

Krahnstoever, N., Yeasin, M., and Sharma, R. 2002. Automatic Acquisition and Initialization of Articulated Models. Paper presented at *To appear in Machine Vision and Applications*.

McNeill, D. 1992. *Hand and Mind*: The University of Chicago Press, Chicago IL.

Oviatt, S. 1996. Multimodal interfaces for dynamic interactive maps. Paper presented at *Conference on Human Factors in Computing Systems (CHI'96).*

Oviatt, S., Angeli, A. De, and Kuhn, K. 1997. Integration and synchronization of input modes during multimodal human-computer interaction. Paper presented at *Conference on Human Factors in Computing Systems (CHI'97).*

Pavlovic, V. I., Sharma, R., and Huang, T. S. 1997. Visual interpretation of hand gestures for human- computer interaction: A review. *IEEE Trans on Pattern Analysis and Machine Intelligence* 19:677-695.

Sharma, R., Cai, J., Chakravarthy, S., Poddar, I., and Sethi, Y. 2000. Exploiting Speech/Gesture Co-occurrence for Improving Continuous Gesture Recognition in Weather Narration. Paper presented at *International Conference on Face and Gesture Recognition (FG'2000)*, Grenoble, France.

A Generic Formal Description Technique for Fusion Mechanisms of Multimodal Interactive Systems

Philippe Palanque, Amélie Schyn

LIIHS - IRIT, Université Toulouse III, 118, route de Narbonne, 31062 Toulouse Cedex 4
Tel : +33 (0)561 556 359 - Fax : +33 561 556 258
palanque@irit.fr, schyn@irit.fr
http://lihs.univ-tlse1.fr/palanque, http://lihs.univ-tlse1.fr/schyn

Abstract

Representing the behaviour of multimodal interactive systems in a complete, concise and non-ambiguous way is still a challenge for formal description techniques. Indeed, multimodal interactive systems embed specific constraints that are either cumbersome or impossible to capture with classical formal description techniques. This is due to both the multiple facets of a multimodal system (in terms of supported modes) and the strong temporal constraints usually encountered in this kind of systems. This position paper presents a formal description technique dedicated to the engineering of interactive multimodal systems. The formal description technique is then used for the modelling and analysis of two fusion mechanisms. Lastly, benefits and limitations of the approach are discussed.

1. Introduction

Despite some efforts for providing toolkits for the construction of multimodal interactive systems (Bederson et al. 2000, Chatty 94), the actual engineering of multimodal interactive systems remains a cumbersome task usually carried out in a rather crafty process. Indeed, while the design (Coutaz & Nigay 1993, Nigay & Vernier 2000) and the evaluation (Coutaz et al. 1996) of multimodal interactive systems have been thoroughly studied, the process of going from a given design to an actual functional system has been the focus of very little research work.

An important aspect of this development process is the reuse of work done from a previous design to another application. Some work on toolkits (Bederson & al. 2000) and architectures (Nigay & Coutaz 95) address this problem at a very low level of abstraction thus making the solution bounded either to modalities or to development platforms.

We believe that the use of an adequate formal description technique can provide support for a more systematic development of multimodal interactive systems. Indeed, formal description techniques allows for describing a system in a complete and non-ambiguous way thus allowing for an easier understanding of problems between the various persons participating in the development process. Besides, formal description techniques allow designers to reason about the models by using analysis techniques. Classical results can be the detection of deadlock or presence or absence of terminating state. A set of properties for multimodal systems have been identified (Martin 1999, Coutaz et al. 1995) but their verification over an existing multimodal system is usually impossible to achieve. For instance it is impossible to guarantee that two modalities are redundant whatever state the system is in.

The paper is structured as follows. Next section is dedicated to related work dealing with specification of multimodal interactive systems. Section 3 is dedicated to the informal presentation of the ICO (Interactive Cooperative Objects) formalism. Section 4 presents extensions to ICO called the MICO formalism (Multimodal Interactive Cooperative Objects) which is dedicated to the formal description of multimodal interactive systems. This formalism is applied to two fusion mechanisms. The first one integrates voice and gesture while the second one features two handed interaction. Last section (section 5) presents the advantages and limitations of the approach as well as future and ongoing work.

2. Related work

Work in the field of multimodal can be sorted in four main categories. Of course the aim of this categorization is not to be exhaustive but to propose an organisation of previous work in this field.

- Understanding multimodal systems
  - (Coutaz & Nigay 1993) typology of multimodal systems, refined in (Bellik 1995)
  - (Coutaz et al. 1995), (Martin 1999) presenting properties of multimodal systems
- Software construction of multimodal systems
  - (Chatty 94, Bederson et al. 2000) propose toolkits for the construction of multimodal systems
  - (Nigay & Coutaz 95) proposes a generic software architecture for multimodal systems
- Analysis and use of novel modalities

- ✍ (Bolt 1980) presents the first use of voice and gesture as combined modalities.
- ✍ (Buxton & Myers, 1986) introduces two handed interaction.
- ✍ (Bolt & Herranz 92) introduces the use of the two handed interaction in virtual reality applications.
- ✍ (Vo & Wood 1996) presents Jeanie, a multimodal application, to test the use of eyes tracking and lips movements' recognition.

o Multimodal systems description.
- ✍ (Nigay & Coutaz 95) presents a Multimodal Air Traffic Information System (MATIS) using both voice and direct manipulation interaction.
- ✍ (Cohen et al. 1997) presents QuickSet a cooperative user interface to military systems using both voice and gesture.
- ✍ (Bier et al. 1993) presents a drawing systems featuring two handed interaction through a trackball and a mouse.

While formal description techniques have been defined and used for interactive systems since the early work from Parnas (Parnas 69), their extension and use for multimodal systems is still relatively rare. We can quote for instance work from (Duke & Harrison 1997) or (MacColl & Carrington 98) were they present how software engineering techniques such as Z and CSP can be used for the modelling of MATIS the multimodal air traffic information system developed byNigay (Nigay & Coutaz 1995).

We believe that multimodal interactive systems feature intrinsic characteristics that make formal description techniques used in software engineering not directly suitable for multimodal systems. First, multimodal interactive systems are, by definition, interactive and thus behave in an event-driven way, usually hard to capture and represent in state based descriptions such as Z. Second, the temporal constraints are at the core of these systems which are more often than not real time and highly concurrent. Indeed, users' actions may occur simultaneously on several input devices and the fusion mechanism must process those input in real-time. Formal description techniques with an interleaving semantics (such as CSP, CCS or LOTOS) are not capable of representing such truly concurrent behaviours. Lastly, the use of temporal windows in fusion mechanisms requires, from a formal description technique, the possibility to represent time in a quantitative way by expressing for instance that an event must be received within 100 milliseconds.

Petri nets is one of the few formal description techniques that allows for representing the behaviour of such systems. Indeed, they feature true-concurrency semantics, they are able to deal both with events and states and they provide several ways to represent quantitative time (Bastide & Palanque 1994).

For space reasons we do not present in detail the notation here but next section shows how these characteristics are used while modelling two fusion mechanisms.

3. Informal Description of ICOs

The Interactive Cooperative Objects (ICOs) formalism is a formal description technique dedicated to the specification of interactive systems (Bastide et al. 1998). It uses concepts borrowed from the object-oriented approach (dynamic instantiation, classification, encapsulaton, inheritance, client/server relationship) to describe the structural or static aspects of systems, and uses high-level Petri nets (Genrich 1991) to describe their dynamic or behavioural aspects.

ICOs are dedicated to the modelling and the implementation of event-driven interfaces, using several communicating objects to model the system, where both behaviour of objects and communication protocol between objects are described by Petri nets. The formalism made up with both the description technique for the communicating objects and the communication protocol is called the Cooperative Objects formalism (CO and its extension to CORBA COCE (Bastide et al. 2000)).

In the ICO formalism, an object is an entity featuring four components: a cooperative object with user services, a presentation part, and two functions (the activation function and the rendering function) that make the link between the cooperative object and the presentation part.

Cooperative Object (CO): a cooperative object models the behaviour of an ICO. It states how the object reacts to external stimuli according to its inner state. This behaviour, called the Object Control Structure (ObCS) is described by means of high-level Petri net. A CO offers two kinds of services to its environment. The first one, described with CORBA-IDL (OMG 1998), concerns the services (in the programming language terminology) offered to other objects in the environment. The second one, called user services, provides a description of the elementary actions offered to a user, but for which availability depends on the internal state of the cooperative object (this state is represented by the distribution and the value of the tokens (called marking) in the places of the ObCS).

Presentation part: the Presentation of an object states its external appearance. This Presentation is a structured set of widgets organized in a set of windows. Each widget may be a way to interact with the interactive system (user → system interaction) and/or a way to display information from this interactive system (system → user interaction).

Activation function: the user → system interaction (inputs) only takes place through widgets. Each user action on a widget may trigger one of the ICO's user services. The relation between user services and widgets is fully stated by

the activation function that associates to each couple (widget, user action) the user service to be triggered.

Rendering function: the system → user interaction (outputs) aims at presenting to the user the state changes that occurs in the system. The rendering function maintains the consistency between the internal state of the system and its external appearance by reflecting system states changes.

ICO are used to provide a formal description of the dynamic behaviour of an interactive application. An ICO specification fully describes the potential interactions that users may have with the application. The specification encompasses both the "input" aspects of the interaction (i.e. how user actions impact on the inner state of the application, and which actions are enabled at any given time) and its "output" aspects (i.e. when and how the application displays information relevant to the user).

An ICO specification is fully executable, which gives the possibility to prototype and test an application before it is fully implemented (Navarre et al. 2000). The specification can also be validated using analysis and proof tools developed within the Petri nets community and extended in order to take into account the specificities of the Petri net dialect used in the ICO formal description technique.

4. Fusion Mechanisms Modelling

This section presents how MICO formalism can be used for the modelling of two fusion mechanisms. As explained in previous section this formalism is able to capture all the elements that are embedded in fusion mechanisms.

4.1. Voice and Gesture Interaction

Our first example is Bolt's system (Bolt 80). Bolt was the first to have the idea to use voice and gesture recognition synergistically for multimodal input. This idea had been implemented in a drawing application, in which user can specify a command orally and give its arguments with either a precise oral description or with a deictic word (this, here, there, ...) and a designation gesture

In this system, five different commands are allowed: create, name, delete, make and move. Each command features a given number of arguments. As long as the command is incomplete, the system waits for the missing argument(s). When a deictic is uttered, user's gesture is taken in account.

As an input for the modelling of this system, we have taken the informal description that can be found in Bolt's papers. Of course, as this application has been presented in natural language and implemented, but not formally described, it is difficult to perfectly understand the functioning of the integration between deictic and gesture. We have supposed that the analysing of a deictic word is at the origin of the triggering of the gesture recognition. Similarly, fusion criteria between command and its potential arguments are not detailed. In the model, this has been represented by the use of typing constraints. Figure 2 and Figure 2 present the formal description of the system according to the assumption presented above. This model describes in a non-ambiguous way the behaviour of the fusion mechanism. In the model rectangles (called transitions) represent actions the system can perform while ellipses (called places) represent state variable of the system. Places can hold tokens and the distribution of tokens in the places represent the current state of the system. The Petri model used in the MICO formalism is called a high-level Petri net model as token can hold values.

.

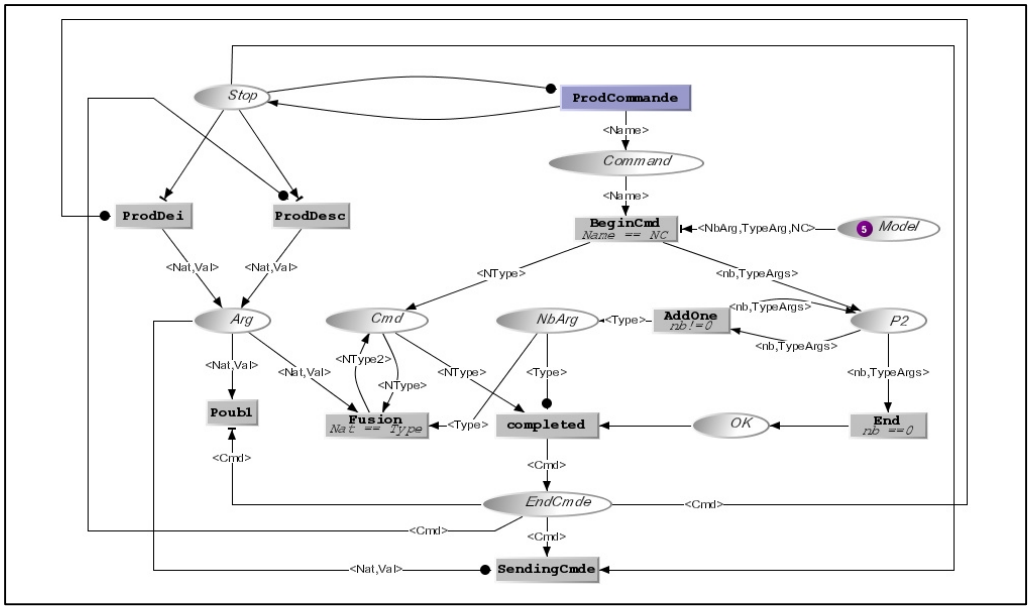


Figure 1. A formal description of the fusion mechanism in Bolt's system (behavioural part)

Type definition of Places in the Petri net

```
Place Stop < >
Place Command < String >
Place Model < Command  > =
        {       [2, [Object, Object], “Make”],
                [3, [Object, Position], “Move”],
                [1, [Object], “Delete”],
                [2, [Object, String], “Name”],
                [2, [Object, Position], “Copy”]          }
Place Arg < Argument >
Place P2 < Int, Sort >
Place NbArg < Sort >
Place Cmd < Sort [ ] >
Place OK < >
Place EndCmde < Sort [ ] >
```

Definition of types used in the Petri net model

```
Class Position      {       Int X;
                            Int Y;              }
Class Object        {       String Name;
                            String  Size;
                            String Color        }
Class Sort  { Object, Position, String }
Class Command       {       Int NbArg ;
                            Sort [ ] TypeArg ;
                            String NC;          }

Class Argument      {       Sort Nat;
```

Avail function:

Avail (ProdCommand) = { ProdCommand } ;
Avail (ProdDei) = { ProdDei } ;
Avail (ProdDesc) = { ProdDesc } ;

Activation function:

Media	Interaction object	Event	Service	Rendering method
Microphone	No	Utterance of “Make”	ProdCommand	No
Microphone	No	Utterance of “Move”	ProdCommand	No
Microphone	No	Utterance of “Delete	ProdCommand	No
Microphone	No	Utterance of “Name”	ProdCommand	No
Microphone	No	Utterance of “Copy”	ProdCommand	No
Microphone	No	Utterance of a deictic word	ProdDei	No
Microphone	No	Utterance of a description	ProdDesc	No

Event production function:

Element name	Event produced
EndCmde	Event Comand_completed

Figure 2. A formal description of the fusion mechanism in Bolt's system(interaction part)

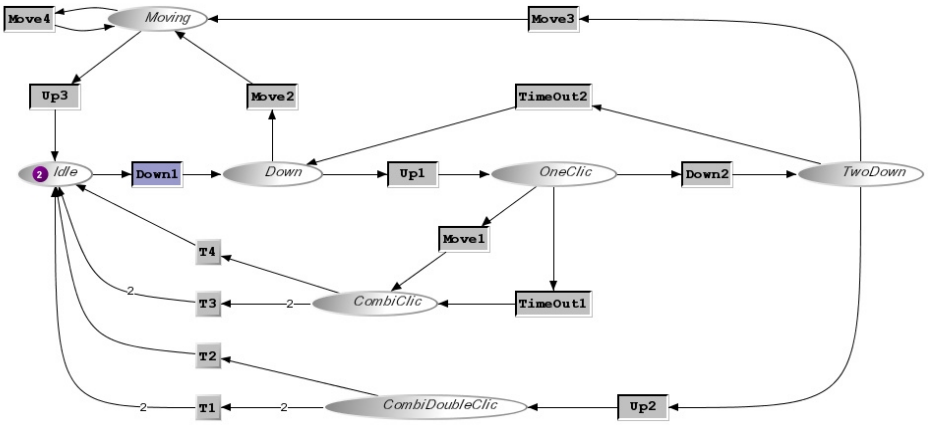
The initial state of the system presented in Figure 2 is thus the presence of 5 tokens in place *Model* and no token in the other places of the net. The values of these token are the description of the commands the system can interpret i.e. their number of arguments, the type of this arguments (object, position or name) and their name (create, name, delete, make or move). Places and transitions are related by arcs. A transition can be fired (i.e. an action performed) if and only if each input place of the transition holds at least one token. When a transition is fired, the token are remove from the input places and one token is deposited in each output place. The model in Figure 2 features two specific kinds of arcs. Tests arcs model the fact that a token is tested i.e. it is not removed or changed by the firing of the transition but its existence is necessary for the actual firing of the transition. Such an arc is represented between transition ProdDesc and place *Stop* meaning that in order for the system to process a description (transition ProdDesc) the system must be in the Stop state i.e. place *Stop* holds at least one token. Inhibitor arc model the zero test in a Petri net. For instance the arc between place *Stop* and transition ProdCommand is an inhibitor arc (the end of the arc is a black dot) meaning that this transition can only be fired if there is no token in place *Stop*. Relationship between transitions and events is done by means of dedicated transitions called synchronised transitions. A synchronized transition can only be fired if it is fireable (according to the current marking of the net) and the associated event is triggered (for instance after a corresponding user action on a dedicated input device). In Figure 2, “ProdCommand”, “ProdDei” and “ProdDesc” are synchronized with user events (utterance of a

command, deictic word or univocal description). As voice modality is dominant in this system, gestures are taken in account only if a deictic word is uttered. So there is no gesture event. Formal analysis of the Petri net of Figure 2 guarantees that whatever state the system is in, there is always at least one transition in the model that is fireable which means that the model is live. For space reasons we don't explain in details other properties that can be proven on the model and how the formal analysis is performed.

4.2. Two handed interaction

The models in Figure 4 and Figure 3 describe the interaction level events policy for two handed interaction. There is no assumption about the type of the devices except that they are graphical and produce the same set of low level events. It tells when and how those events are produced according to the user's actions on the devices. In this Petri net, the policy works like a transducer: each time a physical event is accepted, the Petri net fires a transition and creates higher level events.

For example, in the policy represented in Figure 4, the physical mouse-move (m) is transformed into a higher level mouse-move (M), e.g. the transition between places *One_Click* and *Idle* reacts to the event m by generating another event M plus an event click (C). However; all physical (low level) events are not immediately translated into interaction level events. For example, each event d (down) received while the system is in the initial state, is consumed without any production.



Type of places

Place Ilde < Int > = { 0, 1 }
Place Down < Int >
Place OneClic < Int >
Place TwoDown < Int >
Place CombiClic < Int >
Place CombiDoubleClic < Int >
Place Moving < Int >

Avail function :

Avail (MouseDown) = { Down1, Down2 } ;
Avail (MouseUp) = { Up1, Up2, Up3 } ;
Avail (MouseMove) = { Move1, Move2, Move3, Move4 } ;
Avail (Time) = { TimeOut1, TimeOut2 } ;

Figure 3. Formal model of a fusion mechanism for two handed interaction (behavioural part)

Activation function:

Media	Interaction object	Event	Service	Rendering method
Mouse button	All	Down	MouseDown	Internal
Mouse button	All	Up	MouseUp	Internal
Mouse	All	Move	MouseMove	Move_Mouse
		TimeOut	Time	Internal

Event production function:

Element name	Event produced
T1	Event CDC
T3	Event CC
Up2	Event DC
Move1	Events C and M
TimeOut1	Event C
TimeOut2	Event C
Move2	Event B
Move4	Event D
Up3	Event E
Move3	Events C and B

Figure 4. Formal model of a fusion mechanism for two handed interaction (interaction part)

We have already presented this example in (Accot et al. 1996) while presenting how transducer can be modelled using Petri nets. The model in Figure 4 presents a way to integrate information provided by two different mice. The system can react to the following set of events produced through users' actions on the physical devices: Button Down (d), Button Up (u), Mouse Move (m) and Time Out (t). If a precise sequence of event is performed on the mice, multimodal events are produced by the model. Such events are: "CombiDoubleClick" and "CombiClick" corresponding to the arrival, in the model, of tokens in place CombiClick and CombiDoubleClick.

5. Conclusion and future work

The MICO formalism is an extension of ICO formalism that is formalism dedicated to the design specification, verification and prototyping of interactive systems. Its formal underpinnings make it especially suitable for safety critical interactive systems. ICO formalism has been applied to various kinds of systems including business, Air Traffic Management and command and control applications. The continuously increasing complexity of the information manipulated by such systems calls for new interaction techniques increasing the bandwidth between the system and the user. Multimodal interaction techniques are considered as a promising way for tackling this problem. However, the lack of engineering techniques and processes for such systems makes them hard to design and to build and thus jeopardises their actual exploitation in the area of safety critical application.

This position paper has presented a formal description technique that can be used for the modelling and the analysis of multimodal interactive systems. This work is part of a new project on the evaluation and use of multimodal interaction techniques in the field of command and control real time systems.

6. References

(Accot et al. 1996) Accot, J. Chatty, S. and Palanque, P. (1996). A Formal Description of Low Level Interaction and its Application to Multimodal Interactive Systems. In *3rd EUROGRAPHICS workshop on "design, specification and verification of Interactive systems"* (pp. 92-104). Springer Verlag,.

(Bastide & Palanque 1994) Bastide, Rémi and Palanque, Philippe. Petri Net based design of user-driven interfaces using the Interactive Cooperative Objects formalism. in: Paternò, Fabio (Ed.). Interactive systems: design, specification, and verification, DSV-IS'94. Springer-Verlag; 1994; pp. 383-400.

(Bastide et al. 1998) Rémi Bastide, Philippe Palanque, Duc-Hoa Le, Jaime Muñoz.Integrating Rendering Specifications into a Formalism for the Design of Interactive Systems. 5th Eurographics workshop on "design, specification and verification of Interactive

systems", DSV-IS'98, U.K., 3-5 june 1998, Springer Verlag,

(Bastide et al. 2000) Bastide, Rémi; Sy, Ousmane; Palanque, Philippe, and Navarre, David Formal specification of CORBA services: experience and lessons learned. ACM Conference on Object-Oriented Programming, Systems, Languages, and Applications (OOPSLA'2000); Minneapolis, USA. ACM Press; 2000: 105-117. ACM SIGPLAN Notices. v. 35 (10)).

(Bederson et al. 2000) Bederson, B. B. Meyer, J. and Good L. (2000). Jazz An Extensible Zoomable User Interface Graphics Toolkit in Java. In *UIST'2000, ACM Symposium on User Interface Software and Technology, CHI Letters* (2(2), p. 171-180).

(Bellik 1995) Bellik, Y. (1995). Interfaces Multimodales: Concepts, Modèles et Architectures. Thèse de Doctorat en Informatique. Université de Paris-XI.

(Bier et al. 1993) Bier, E. Stone, M. Pier, K. Buxton, W. and DeRose, T. (1993). Toolglass and Magic Lenses: the See-Through Interface. In *Proceedings of ACM SIGGRAPH'93* (pp. 73-80). Anaheim, ACM Press.

(Bolt & Herranz 1992) Bolt, R and Herranz, E. (1992). Two-Handed Gesture in Multi-Modal Natural Dialog. In *Proceedings of the fifth annual ACM symposium on User interface software and technology* (p 7-14). Monteray, California. ACM Press.

(Bolt 1980) Bolt, R. (1980). Put That There: Voice and Gesture at the Graphics Interface. In *SIGGRAPH'80* (Vol 14, p262-270).

(Buxton & Myers 1986) Buxton, W. and Myers, B. (1986). A Study in Two-Handed Input. In *Proceeding of the ACM CHI* (p 321-326) Addison-Wesley.

(Chatty 1994) Chatty, S. (1994). Extending a Graphical Toolkit for Two-Handed Interaction. In *Proceedings of the ACM symposium on User Interface Software and Technology,* (p195-204) Marina del Rey, California. ACM Press.

(Cohen et al. 1997) Cohen, P. Johnston, M. McGee, D. Oviatt, S. Pittman, J. Smith, I. Chen, L. and Clow, J. (1997). QuickSet : Multimodal Interaction for Distributed Applications. In *Proceedings of the fifth ACM international conference on Multimedia* (p 31-40) Seattle, Washington. ACM Press.

(Coutaz & Nigay 1993) Coutaz, J. and Nigay L. (1993). A Design Space for Multimodal Systems Concurrent Processing and Data Fusion. In *Human Factors in Computing Systems, INTERCHI'93 Conference proceedings* (p 172-178) Amsterdam, The Netherlands.

(Coutaz et al. 1995) Coutaz, J. Nigay, L. Salber, D. Blandford, A. May, J. and Young, R. (1995). Four Easy Pieces for Assessing the Usability of Multimodal in Interaction the CARE Properties. In *Human Computer Interaction, Interact' 95* (p115-120). Lillehammer, Norway.

(Coutaz et al. 1996) Coutaz, J. Salber, D. Carraux, E. and Portolan, N. (1996). Neimo, a Multiworkstation Usability Lab for Observing and Analysing Multimodal Interaction. In *Human Factors In Computing Systems CHI'96 Conference Companion* (p 402-403) Vancouver, British Columbia, Canada. ACM Press.

(Duke & Harrison 1997) Duke, D. and Harrison, M. D. (1997). Mapping User Requirements to Implementations. In *Software Engineering Journal* (Vol 10(1), p 54-75).

(Genrich 1991) Genrich, H.J. *Predicate/Transition Nets*, in K. Jensen and G. Rozenberg (Eds.), High-Level Petri Nets: Theory and Application. Springer Verlag, Berlin, pp. 3-43.

(MacColl & Carrington 1998) MacColl and Carrington, D. (1998). Testing MATIS: a Case Study On Specification-Based Testing of Interactive Systems. In *FAHCI'98* (p57-69). ISBN 0-86339-7948.

(Martin 1999) Martin, J.C. (1999). TYCOON six Primitive Types of Cooperation for Observing, Evaluating and Specifying Cooperations. In *Working notes of the AAAI Fall 1999 Symposium on Psychological Models of Communication in Collaborative Systems.* Sea Crest Conference Center on Cape Cod, North Falmouth, Massachusetts. http//www.limsi.fr/Individu/martin/aaai99/html/martin-final-v7.html

(Nigay & Coutaz 1995) Nigay, L. and Coutaz, J. (1995). A Generic Platform for Addressing the Multimodal Challenge. In *Conference proceedings on Human factors in computing systems* (p 98-105). Denver, Colorado.

(Nigay & Vernier 2000) Nigay, L. and Vernier, F. A Framework for the Combination and Characterization of Output Modalities. In *Proceeding of the DSV-IS '2000 Conference* (p 35-50). P. Palanque, F. Paterno (Eds) Spinger.

(OMG 1998) OMG. The Common Object Request Broker: Architecture and Specification. CORBA IIOP 2.2 /98-02-01, Framingham, MA (1998).

(Parnas 69) Parnas, D. L. (1969). On the Use of Transition Diagram in the Design of a User Interface for Interactive Computer System. *In Proceedings of the 24th ACM Conference,* (p. 379-385).

(Vo & Wood 1996) Vo, M.T. and Wood, C. (1996). Building an Application Framework for Speech and Pen Input Integration in Multimodal Learging Interface. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP) IEEE*, (Vol 6, p 3545-3548)

Eye-Bed

Ted Selker, Winslow Burleson, Jessica Scott, Mike Li

MIT Media Lab
20 Ames St. Cambridge, MA 02139
{selker, win}@media.mit.edu; {jbscott, mli}@mit.edu

Abstract

The Eye-bed prototype introduces new ergonomic language scenarios. This paper focuses on developing a demonstration eye gesture language for intelligent user interface and computer control. Controls are integrated with a user model containing a history of previous interactions with the interface. Context recognition enables the Eye-Bed environment to continually adapt to suit the user's inferred and demonstrated preferences. Staring, gazing, glancing, blinking, etc… are part of person-to-person visual communication. By creating a computer interface language composed of exaggerated eye gestures, we create a multi-modal interface that is easy to learn, use, and remember.

1. Introduction

Computer human interfaces have long been applied to everyday situations. These interfaces are often trapped in a user-directed model, relying on the user to know and use a language to directly specify what she wants from the computer. More recently computers are finding their way into everyday things. These days our appliances seem to need to have their computers booted before they work. Cars, phones, record players even house locks come to a grinding halt when their computers don't work. We have been programming our thermostats, watches, and videotape recorders for so long that it seems reasonable to spend hours learning to use an MP3 music player that fits in your hand. They are about to get simpler.

At the very least hospital beds have some control for comfort, height, angle, and temperature. The bed area has another control to call an assistant. Usually each bed has a control for a television. How should we position ourselves optimally to watch a movie from bed? Automation of communication and media in bed could be very useful. The Media-Bed scenario adds new rich command and control integration opportunities to the computer human interface (Shelley, R., 2001).

If one integrated environmental controls, educational materials, and entertainment media into a bed interface how would a user communicate with it? Imagine imagery projected on the ceiling over a bed. (Figure 2.) Consider functions presented spatially as an integrated ecological user interface. This would require a person to select things on the spatial interface. Spatial selection has many dexterity problems. Historically the control theory issues and other obstacles of the components of Fitts and Steering laws (Hinckley, K. et al, 1994) have paled in comparison to the difficulty of learning command languages. We consider new languages that are trivial to learn and require low cognitive overhead to use. Through mimicry and extension of the social communication people employ nonverbally, we explore the realm of reduced consciousness communication.

Graphical interfaces are wonderful in that they allow a user to recognise something they might not have been able to remember otherwise, like a place in a file hierarchy. If they want an item they simply point at it to select it. Still, graphical interfaces have long been cumbersome and frustrating. People control 3D interfaces with analogue devices that change the rate and angle of motion as though the ultimate way to interact with a computer would be some sort of hovering gravity-independent helicopter. It is hard to learn to fly a helicopter. Many novice users of 3D interfaces have the constant feeling of listing dangerously as they walk into walls and can't stop the scenes from rotating.

Figure 1: Multimedia Bed with Ceiling Projection

A provocative area of user input design has been eye control. It has seemed like one of the ultimate interface

approaches since the 1960s when the first rudimentary mechanical Perkinji trackers were demonstrated. Indeed psychologists and marketing people have used eye position to understand people's interests. (Yarbus, L. 1967). Unfortunately eye tracking utility has been stymied: the head moves; the eyes don't want to look at one thing; the tracking devices work in lab settings better than in an office; etc…. For the past 40 years, people have been improving eye-tracking technology using Electro-occulogram eye trackers, contact lenses, tracking infrared cameras and dual cameras to get to lightweight camera based systems. Dual camera systems like the Autostereoscopic user-computer interface (Pastoor, S., Skerjanc, R., 1997) and multiple multiplexed structured light source camera systems like the Blue Eyes™ system at IBM have become excellent tools (Flickner, 2001). Unfortunately, these "better eye tracking systems" have made it even more obvious that the eye is not simply looking for interesting things that it wants to effect.

The eye is not a cursor control device. The eye notices movement in the periphery and has to attend to it vigilantly searching for danger. The eye is a guard dog; it has a job to do. Using eyes and trackers to move a cursor precisely is like using a security officer in a bank to show people the bathroom. The officer could do it but not as well as a concierge; at the same time the officer would risk being remiss in the primary security duties.

2. Eye motion as language

Large areas of the brain are devoted to interpreting visual input and controlling the eye (Carpenter, M., 1976). The sensitivity of the eye itself makes it a strange choice for a pointing device. The eye, after all, seeks to understand anything in its view. The area centralis is some 3 degrees wide; anything in this visual area is well known to the mind. One of the most difficult issues with eye-tracking scenarios is that the eye-tracking computer demands "eye contact". This is the very thing people are most used to devoting to scanning for safety, acknowledging other people, and to expressing their feelings non-verbally (Clark, H., Wilkes-Gibbs, D., 1986).

Interest Tracker (Maglio et al, 2000) lets people use generalized directional gaze to select information content by demonstrated interest, much as a person does when meeting a new acquaintance. This stands in contrast to the standard eye-tracking interface in which a user is asked to stare at a specific thing until it is selected: the physical difficulty of doing so; the social inappropriateness; and the uncomfortable feeling of the interface is significant. In contrast if the user is asked to look at the general area of an item to be selected these interface obstacles are diminished.

More recent work, demonstrated "Magic Pointing" (Shumin, Z. 1999), an approach that uses eye gaze to make a non-linear jump or "warp" a cursor to where the eye is looking on a screen. Subsequent GUI control is done through the standard cursor control device. It is quite easy to use eye tracking to identify areas of interest. Of value to interface design is the fact that the eye is a course output device and a fine input device. The most important notion however is that Interest Tracker and Magic Pointing take advantage of the fact that the eye wants to look in the area of interest. The syntax that the action of looking at a work area changes the spatial position of the cursor is a powerful one. Using a dwell time of just 0.3 seconds was more than adequate to allow a user to distinguish things they wanted to select. It was also found to be much faster than a mouse can select the same area (Maglio et al., 2000) Interest Tracker, introduced above, is a system that shows another simple and productive use of gaze interpretation. It augments a person's natural gaze at an area of interest with additional information or content of a similar nature.

Invision (Li, M.; Selker, T., 2001), takes this one step further, based on evidence that shows that the paths that people's eyes follow demonstrate what they are thinking (Yarbus, 1967). When people rapidly transit from one place to another they are more likely to be making a selection of a familiar item. When people's eyes move slowly around the field-of-view they are taking in information, and making decisions, but not selections.

The pattern based Invision interface made two contributions to eye tracking. It demonstrated that eye tracking accuracy could in many cases be improved by interpreting eye movement as the endpoints of the trajectory (i.e. knowing where the eye had moved from and too helps to understand user intent and focus). In the second, and more interesting case, the relationships between objects that the eye gazes at and the order that they are gazed at become the language that drives the computer. The system showed a set of objects representing the various sponsors of the Media Lab. As a person traversed them with their eyes, it used the path to notice their interests. As a person's eye went back and forth, between two things, the objects they were looking at moved closer to one another. In this way, as a user shows interest in a group of items the interface literally brings these items together. This has been explored as an interface for a kitchen as well (opening the refrigerator, oven, cabinets and dishwasher). These pieces of research all focus our attention on the information that comes out of an eye.

2.1.1. Gaze vs. Stare Detection:

The Eye-ARe Project took this further. Eye-aRe is a simple system that consists of a glasses mounted infrared LED and photodiode that detect reflected infrared light from the eye's cornea and sclarea. (Selker et al., 2001) A small PIC can detect when a person is staring and when their eyes stay relatively fixed. It is not hard to separate simple eye gaze intent. This approach can separate out intended versus unintended selection events. Even without a camera, Eye-aRe has successfully been used to send business card information when a user stares at (or is engaged in conversation with) another person, to bring up information about a display when a person looks at it, and detect closed and opened eyes and individual blink signatures.

If the actions used to interact with a computer mimic the normal use of human eye gesture language, this synergy could assist user's learning and memory. Can such an eye gesture based language be the basis of an ecological interface? Can such a natural control language

be integrated without being difficult to learn or generating confusion? Can reasoning, learning, and representation of intelligence be employed to give users more control?

Complex social dynamics are traceable to eye motion (Clark, 1986). These can be used to enhance human computer communication. Eye motion demonstrates a social gesture language. These are significantly easier to record than eye position. With this thesis we will describe the ways that eye gestures and task modeling have been experimented with in the Eye-Bed to reduce reliance on direct manipulation in the interface.

Figure 2: Ecological display projected on Ceiling

3. Media-Bed & Eye-Bed

The bed is a place where the average person will spend approximately one third of their life. Once made of plant fiber and then synthetic materials, we have now made the bed digital. The Media-Bed and Eye-Bed (Figure 1.) are a response to the challenges of integrating environmental, educational, and entertainment controls in a universal interface. The Media-Bed and Eye-Bed could simplify the controls of a hospital bed while adding new features that integrate these domains [good morning america].

The Media-Bed and Eye-Bed are part of a growing body of language based interface development. (Selker, T.; Burleson, W., 2000) The thesis is that replacing explicit spatial selection with a language-based interaction may provide interfaces that are easy to learn, use, and remember. One novel control approach in this direction has been the use of eye tracking. The social language of the eye (i.e. "wink, wink…. Know what I mean" as said again and again in … Monty Python's Holy Grail) can be used as a natural easily understood language. In the Bed projects we overlay and map expected characterized ocular responses such as stare, gaze, wink, etc… with a language to communicate interface intentions between the user and the computer.

The Media-Bed and Eye-Bed are a computer systems that recognizes and remembers what a person is doing in bed to provide useful information and environmental modifications. They "listens" to many information channels to enhance the semantics of a language. The Eye-Bed extends language recognition of the Media-Bed to include eye-tracking semantics: blinking, winking, staring, and gazing. Both create a user model which includes time stamps, interface states, knowledge of the position and sound of the user, in additon to the traditional direct user input channel.

The Media-Bed and Eye-Bed are a place for us to experiment with new scenarios for using a computer in our live. They are also a place to experiment with new multi-modal input devices. For example, eye tracking in a bed has advantages. The person's head is supported and can be stabilized. This naturally reduces the difficulty of finding and tracking the eye position. The bed consists of an integrated multimedia personal computer and video projector. It runs a Macromedia Director movie projected onto the ceiling above a standard bed. This projection creates a virtual world that provides the user with a space for interaction and reactive input.

3.1. Prototype Scenario

A person is lying in bed. Many simple activities can be computer-facilitated making lying in bed more pleasurable and productive. A scene appears, projected on the ceiling above the user's head (Figure 2). It is a scene of rolling hills dotted with icons: an e-mail kiosk, a TV satellite dish, a juke box, a person reading in a lawn chair, a newspaper stand, the moon and stars, and the sun. Each of these icons can move the user into another part of the world depending on his needs and wishes at the time. We have experimented with different renditions of physical world imagery or so-called "ecological interfaces". Ecological interfaces have been shown to improve speed and accuracy of selection over two-dimensional interfaces when users are familiar with them (Ark et al. 1998).

Pointing and selecting it, the kiosk enlarges to fill the screen, bringing the user into another space. A smaller rendition of the rolling hills at the top of the screen points to the original main screen where the user came from. The user can similarly watch TV, read the newspaper or read an online book while lying on their back in bed. The display is projected upward to cover the ceiling above the bed. When reading something or watching TV or a movie, the user no longer has to prop themselves up with their arms or find a comfortable position to sit in. If the user has back or neck problems, this is especially important.

Once the user has finished reading e-mail selecting the hills at the top of the screen returns them to the initial selection screen. It's time to go to sleep, so the user moves to the moon and stars, a soothing song begins to play and a sunset that gradually darkens to reveal the night sky is projected. The bed can subtly and playfully encourage or persuade a person to go to sleep at an hour that they should by shifting to this mode as well. (Fogg, B., 1998) Selecting the moon presents the outlines of constellations. As the user explores the night sky, the names of the constellations and planets appear. Selecting a planet brings up its path and other information. This is an example of how the system can function in an educational and informational role as well. As the user falls a sleep (their eyes close and they move less), the bed recognizes the hour, and sets sunrise wake up music to accommodate the user's sleep patterns. The bed has learned how long it's occupant likes to sleep by

monitoring the use patterns of the alarm clock. Since the bed has access to the user's calendar, it knows the user will not miss any appointments by waking up at eleven o'clock. In the morning, the sun rises on the ceiling, accompanied by morning music. The room is gradually lit up by the sunlight, and the day's schedule is presented for review along with e-mail and newspaper customized to the user's interests and preferences. In this scenarios the user is able to enjoy the activities that they normally enjoy with the media selection assistance from the computer.

Selection of functions on the Media-Bed selection of items on the ceiling was originally accomplished with a Polhemous 6-degree of freedom system in a ball. The position of the ball controlled a ball-shaped cursor on the landscape imagery of the ceiling interface. The ball used a bed based coordinate system to control a cursor on the screen. It was tiresome to hold it in exactly the right position on the bed to activate the functions . The Gyromouse™ did not require the person's hand to go to a specific place in the air or on the bed to use spatial control. The TrackPoint™ in a custom built handle and a TrackPoint keyboard were much easier to use allowing hands to rest on the device. The next step in evaluating the Media Bed interface was to add an eye tracker. The newer Eye-Bed system uses the eye-tracker, positioned in a lamp mounted to the headboard, to control the system.

Through the construction of user model profiles, the Media-Bed and Eye-Bed can learn to suit the user's wishes by understanding what they are interested in seeing, doing, and listening to. The boom box and media presenting applications in the bed do this explicitly. A hiking boot icon when selected kicks the juke box or media player indicating to the user that the system will try to change what media to present. The system changes the current media and updates the model of what to try in the future. It uses artificial intelligence to record actions and reactions of the user to build a model of what kind of information and media will be useful in which situations.

The Eye-Bed version augments the positional syntax of a cursor on a GUI with a language of few simple eye gestures to make an even more interesting interaction scenario. This is done through a paradigm of *relaxed eye tracking*. The Eye-Bed version develops a contextual knowledge of the situation. It uses the "eyes shut" condition to know when a person is asleep or not wanting to see imagery anymore. "Eyes open" to tell the bed that a person need not hear the loud version of the alarm clock, "excessive blinking" or "nervous eyes" to change the station of the radio or TV, and "gazing" into a sparse ecological interface to select interface icons. The eye position itself and the way that a person is looking at something can determine what should be done. If the eye isn't wandering and there is only one nearby object of interest the selection is obvious. Using this multi-modal and contextually aware approach we have enhanced the user interface in the Media bed.

3.1.1. Nervous Eyes Want Change

Work with Eye-aRe and the work of many other researchers have shown that it is easy to recognize rapid blinking as a sign of dissatisfaction. In the Eye-Bed we integrate rapid blinking as the syntactic way to say you are not satisfied with the current interactions. For example, we used rapid blinking to change the channel on the radio and video, in a similar manner to the boot kicking the player. Since this action is similar to the natural way of communicating dissatisfaction, people are able to remember the action and accomplish it with ease.

3.1.2. Open Eyes

It is extremely easy to know when an eye is open or closed. Eyes open presumes the person is not asleep and is thus the syntax for telling the bed to activate wake-up imagery of a sunrise and turning off the loud alarm if the time is morning or if the user generally wakes up at that time of day. Likewise if a person is not in bed the wake up alarm is not needed. An eye projected on the ceiling shows the eye open and labels the status "open". This projected eye is part of the feedback to the user that the eye tracking is on and working.

3.1.3. Eyes Closed

Missing pupils is the syntax for putting the system into a sleep mode. Of course, a person need not watch TV or other things when they are asleep so it can fadeout these media. The Eye-Bed system puts up a black screen with "Zzzzzzz…" written across it when a person closes their eyes for several seconds.

3.1.4. Stare

Attention is a fundamental communication act. When a person looks at something intently we call it staring. In the Eye-Bed we use dwell time to activate a spatial icon. Eye-aRe demonstrates that staring at a toy dog is an obvious way to make it respond with a bark; staring at a TV is an obvious way to demonstrate interest in the TV show. Therefore staring in the Eye-Bed is used to select and activate media.

3.1.5. Gaze

When a person looks around we could say they are gazing. In the Eye-Bed the eye moving around without staying anywhere is interpreted as lack of focus on the bed interface. The system shows the interpretation on the ceiling display eye indicator.

The eye gesture syntax described in this section is small. The simple language of eye states has been enough to drive the entire Eye-Bed demonstration.

3.2. Discussion

Typical spatial interfaces use a spatial inclusion syntax. (Selker, T.; Appel, A, 1991). The control moves an indicator or cursor to within the boundaries of a spatial object or icon to associate syntax to it. The eye gesture language is an augmented visual language in which some eye gestures have global consequence while others act as parameters of a selection device just as mouse buttons on a mouse are parameters to the graphical object that the cursor associates it with. The Eye-Bed eye gesture language has made it possible for people to control the entire Media-Bed interface using only their eye gestures.

In using a gesture-based interface it usually becomes difficult to teach and use the gestures. This system's use

of natural eye gestures, which people do anyway, makes using the bed almost as natural as a social interaction. One goal of creating "natural interfaces" is to create interfaces that use the actions that people are familiar with and relate them to actions the system might expect of users. This can be achieved by copying the actions of people. Studying perceptual and physiological actions and capabilities of people is important as well. It has been shown that in many situations people treat computers as they do people (Reeves, B.; Nass, C., 1986). This paper and these uses of eye input demonstrate how the higher order behavioural and social psychological areas can be used as a motivating approach for interface design. By carefully studying these fields exciting taxonomies of natural behaviour can be found. Once found these can become a basis for more natural, social, and gesture-based interaction languages with the computer. Our goal is developing interaction languages that are amalgamations of typical human actions with appropriate computer augmentation to assistance people in what they want to do.

3.3. Status

3.3.1. Media Bed

The Media-Bed is a Macromedia Director program running on a computer. The Media-Bed with physical inputs has been demonstrated to hundreds of people at the MIT Media Lab; the opening of Media Lab Europe in Dublin, Ireland; and at the AAAI Fall 2000 workshop in Falmouth, MA. We are surprised at how relaxing it is to lie down to demonstrate the night time and wake up scenarios. Within days of it working people were approaching us to form marketing alliances. We have used the media Bed and its display as a place to work and find that it is quite relaxing.

3.3.2. User Model

All of the selection scenarios are enhanced by the creation of the user model. The simplest user model is that a person whose eyes are closed need not be shown imagery. Currently we consider a person whose eyes are closed to be asleep.

The user models in the radio and TV are the most sophisticated. These models notice what time of day it is, what has been playing and how long a user listens or watches it as a basis for appreciation. If a user likes the music then similar music continues to play. Of course we have found that some people don't like to hear the same music over and over again. Refining the heuristics for this is a current goal. The eye tracking approach has allowed us to simulate nervousness or detect actual nervousness as the way to tell the media generator that it should attempt to find other media to play. If a person is not paying any attention to anything near the media player and has not recently turned it on, these analyses of nervousness most likely are not about the media

3.3.3. Eye-Bed

An early version of the Eye-Bed was demonstrated on Good Morning April 10 2001 (Shelley, R. 2001). The Eye-Bed is the Media-Bed with another computer running the eye gesture recognition software. Mike Li wrote a Java version of the Eye-Bed software. It was replaced with a C version written by Jessica Scott that requires much less of the Ethernet communication for its interpretations. The New version has a much better ability to interpret eye gestures. Further, the new version includes the eye indicator on the ceiling bed display.

The Eye-Bed eye gesture based interface has been demonstrated dozens of times at the MIT Media Lab. The ability to control it with less than a minute of instruction amazes everyone. The impressive thing about Eye-Bed is that people enjoy using it and don't need much instruction. The system is so easy to use that we often have visitors demonstrate the eye-gesture based interface to one another. The real value of this interface is the ease with which we can recognize the gestures of eyes closed, open, gaze, stare, blink, and nervous blinking.

The current system has limitations. Text entry has not been satisfactorily resolved. There are good and bad times to use the system. So far the system is designed for a single user and does quite well at integrating the many controls of the previously discussed hospital bed. However the system does not make any accommodation for the social or sexual activities that take place in bed in fact at this point many users think that the current features are too intrusive. They are appalled at the thought of email intruding into their bedroom and literally "hanging over their heads".

3.4. Future Work

The interface is effective enough for us to sleep with it on and beneficial enough for us to enjoy it when we are awake. The goal of demonstrating the limits of time and fidelity of eye gesture are central to our future work. The integration and evaluation of new eye gestures and other physiologically natural gestures is central to the context aware stance of the research group that this work takes place in. Understanding what social cues are for and how to make them reliable within a graphical interface system continues to be exciting. We will extend the language that we have developed to include other forms of implicit communications such as facial gestures. The question as to whether a serious formal theory will aid in this endeavour stands before us.

Discussions in bed, on the phone, or in person will be augmented by pervasive access to information. The nature of this information will also rely upon user models. For example, a four-year old who wants to know what bears eat, is looking for a different answer than what a college biology major with the same question is looking for. We will continue to explore the integration of health monitoring and feedback systems. Sound sensing and acoustical feedback will be used to monitor sleep apnoea and snoring. The Media-Bed and Eye-Bed has moved into educational areas, starting with astronomy. We will soon move on to other contextually appropriate topics. Especially interesting is the context of looking up such as in auto mechanics, marine biology, meteorology, ornithology, and rainforest canopy sciences. This work will also be extended into the realms of fun, play, and creativity by implementing games and motivational activities.

4. Conclusion

The appropriate use of interface techniques should be the focus of the Computer Human Interface field. Unfortunately as industry develops new interface techniques and scenarios designers bring untested ideas into the market. In this paper we attempt to show that a well-understood language of a few eye gestures can simplify the use of the eyes as a control for user interfaces. We further use an ecological interface to simplify teaching control of the user interface. In doing so we create a system that is natural and ease for people to learn, use, and remember. The goal of developing improved user interactions will continue to require us to invent new scenarios and test where and how they can be applied.

5. Acknowledgements

We thank the MIT Media Lab for supporting this work; Jesse Pavel for the Polhemus to Director Interface; Kim May and Ian May for the hand held track point.

6. References

Ark, W.; D. Christopher Dryer, Ted Selker, Shumin Zhai.1998 Representation Matters: The Effect of 3D Objects and a Spatial Metaphor in a Graphical User Interface. People and Computers XIII, Proceedings of HCI'98, (Eds) H. Johnson, N. Lawrence, C. Roast. Springer. 209 –219.

Carpenter, M., 1976. Human Neuroanatomy. Waverly Press. Baltimore, MD.

Clark, H. H., & Wilkes-Gibbs, D. 1986. Referring as a collaborative process. *Cognition*, 22:1-39. *Reprinted in* P. R. Cohen, J. L. Morgan, & M. E. Pollack (Eds.), 1990. *Intentions in Communication.* Cambridge: MIT Press.

Flickner, M. 2001. Blue eyes: Suitor [WWW Document].URLhttp://www.almaden.ibm.com/cs/blueeyes/suitor.html (visited 2001, February 2).

Fogg, B.J. 1998. Persuasive Computers: Perspectives and Research Directions. Conference on Human Factors in Computing Systems: CHI 1998 Conference Proceedings. 225-233Lieberman, H.; Selker, T. 2000. Out Of Context: Computer Systems That Adapt To, and Learn From, Context. IBM Systems Journal 39, Nos. 3&4: 617-632.

Hinckley, K., Pausch, R., Goble, J. C., Kassell, N. F. April 1994. A Survey of Design Issues in Spatial Input**,** Proc. ACM UIST'94 Symposium on User Interface Software & Technology. 213-222.

Li, M.; Selker, T. 9-2001. Eye Pattern Analysis Occular Computer Input, IVA.

Maglio, P.P., Barrett, R. Campbell, C.S., and Selker, T. SUITOR: An attentive information system. In *Proceedings of IUI2000* (New Orleans, LA, Jan. 2000), ACM Press.

Pastoor, S.; Skerjanc, R. 1997. Autostereoscopic user-computer interface with visually controlled interactions. Digest of Technical Papers, SID'97 International Symposium. 277-280.

Reeves, B.; Nass, C. 1996. The Media Equation. How People Treat Computers, Television, and New Media Like Real People and Places. Stanford, CA: CSLI Publications.

Selker, T.; Burleson, W. 6-2000. Context-Aware Design and Interaction in Computer Systems. IBM Systems Journal 39, Nos. 3&4.p. 880-891

Selker, T.; Lockerd, A.; Martinez, J.; Burleson, W. 2001. Eye-aRe: A Glasses-Mounted Eye Motion Deteciton Interface. Conference Proceedings Conference on Human Factors in Computing Systems, CHI2001. Extended Abstracts. 179-180.

Selker, T.; Appel, A, 1991. Graphics as Visual Language.. Handbook of Statistics, Vol. 9, Elsevier Science Publishers BV.

Shelley, R. April 10, 2001. Multimedia Bed. ABC News "Good Morning America."

Yarbus, A. L. (1967). Eye movements during perception of complex objects, in L. A. Riggs, ed., `Eye Movements and Vision'. New York: Plenum Press, chapter 7, 171-196.

Zhai, S., Morimoto, C. & Ihde, S. (1999). Manual and gaze input cascaded (MAGIC) pointing. Proc. ACM CHI ’99 Human Factors in Computing Systems Conference, Addison-Wesley/ACM Press, 1999. 15-20.

PROMISE - A Procedure for Multimodal Interactive System Evaluation

Nicole Beringer[1], **Ute Kartal**[1], **Katerina Louka**[1], **Florian Schiel**[2], **Uli Türk**[1]

[1] Institut für Phonetik und Sprachliche Kommunikation,
[2] Bavarian Archive for Speech Signals (BAS)
Schellingstr. 3, D-80799 München, Germany
{beringer,ukartal,kalo,schiel,tuerk}@phonetik.uni-muenchen.de

Abstract

This paper describes a general framework for evaluating and comparing the performance of multimodal dialogue systems: PROMISE (**Pro**cedure for **M**ultimodal **I**nteractive **S**ystem **E**valuation). PROMISE is a possible extention to multimodality of the PARADISE framework ((Walker and al., 1998; Walker et al., 1997) used for the evaluation of spoken dialogue systems), where we aimed to solve the problems of scoring multimodal inputs and outputs, weighting the different recognition modalities and of how to deal with non-directed task definitions and the resulting, potentially uncompleted tasks by the users.
PROMISE is used in the end-to-end-evaluation of the SmartKom project - in which an intelligent computer-user interface that deals with various kinds of oral and physical input is being developed. The aim of SmartKom is to allow a natural form of communication within man-machine interaction.

1. Introduction

The aim of this paper is to give an extended framework on dialogue system evaluation for multimodal systems in the end-to-end evaluation of SmartKom.

In the SmartKom project, an intelligent computer-user interface is being developed which deals with various kinds of oral and physical input. The system allows and reacts to gestures as well as mimic and spoken dialogue. Potential benefits of SmartKom include the ease of use and the naturalness of the man-machine interaction which are due to multimodal input and output. However, a very critical obstacle to progress in this area is the lack of a general methodology for evaluating and comparing the performance of the three possible scenarios provided by SmartKom:

- SmartKom Home/Office allows to communicate and to operate machines at home (e.g. TV, workstation, radio),
- SmartKom Public provides public access to public services, and
- SmartKom Mobile

Because of the innovative character of the project, new methods for end-to-end evaluation had to be developed partly through transferring established criteria from the evaluation of spoken dialogue systems (PARADISE), and partly through the definition of new multimodal measures. These criteria have to deal with a fundamental property of multimodal dialogue systems, namely the high variability of the input and output modalities with which the system has to cope. As an example, the system can accept a task solution not only via three different input devices (mimic-camera, voice-recording, gesture-camera), but also many different long-term solution strategies. For example, in order to plan an evening watching TV, a subject using the system may start with a sender (or a time, or an actress, etc.), progress to give a time (or sender, or actress, etc.) and end choosing a single programme (or a series, or many programmes, or none, etc.). This inherent complexity and the tasks to be completed make it necessary to find an evaluation strategy which measures up to the possibilities of such a flexible system. Earlier evaluation strategies (PARADISE) had to deal with systems with only one input modality, used given solution strategies and thus were easily able to measure the success of a task. The advancements of SmartKom, though, cannot be adequately measured nor evaluated using an evaluation-strategy based on a monomodal system with pre-given solution strategies.
The following section gives an overview of standard problems of dialogue system evaluation which principly can be solved by the PARADISE framework (Walker and al., 1998; Walker et al., 1997). Section three describes how to define a task and extract the attribute value keys out of the description - solving a problem not uniquely belonging to multimodal dialogue evaluation. How we deal with incomplete tasks or tasks that get a very low task success measure due to incooperativity of the user, is described in section four. The scoring of multimodal inputs and outputs can be found in section four. Sections five to six give a detailed description of the status of multiple-to-one input facilities, i.e. the possibility to express the same user intention via multiple input as well as via different input modalities. Section seven defines the approach of PROMISE as a multimodal dialogue evaluation strategy which normalizes over dialogue strategy and dialogue systems. In the last section we sum up some ideas to be implemented in our framework.

2. Standard problems of dialogue system evaluation

Of course, multimodal dialogue evaluation has to deal with the same problems spoken dialogue system evaluation has to deal with, namely

- How can we abstract from the system itself, i.e. the different hardware and software components, in order to evaluate across dialogues and scenarios (see above)?

- How can we abstract from different dialogue strategies?

The PARADISE framework (for detailed description please refer to (Walker and al., 1998; Walker et al., 1997)) gives a useful and promising approach of how to compare different spoken dialogue strategies and different spoken dialogue systems via attribute value matrices (AVM), to compute the (normalized) task success measure (provided that a clearly defined task description is given to the user) define several (normalized) quality and quantity measures so-called cost functions , and to weigh their importance for the performance of the system via multiple linear regression dependent on the User Satisfaction value (cumulative function on the questionnaire completed by the subjects). This is not practicable, though, when dealing with a multimodal system like SmartKom**.**
Unfortunately, in dealing with multimodal systems we find a number of components which do not fit into the PARADISE approach, which are:

- The user is given a rather unprecise task definition, in order to enable a mostly natural interaction of user and system. Therefore there exist no static definitions of the "keys" (a PARADISE term) necessary to compute an AVM. Our solution is to extract different superordinate concepts depending on the task at hand. For example, when planning an evening watching TV, these superordinate concepts - we call them "information bits" - would contain movie title, genre, channel, timeslots, actors etc. Similar to a content-analysis, these "information bits" are carefully selected, categorized and weighted by hand before the tests start. This makes it possible for us to compute, normalize and compare across different tasks and scenarios.

- The number of information bits can vary within one completed task, but they must define a task unambiguously in order to finish it completely and successfully. For example, when a user asks to explicitly view a movie by name, assuming this movie is broadcasted at a set time and in only one channel, the number of information bits necessary to complete the task is one (the name of the movie). Whereas if a user doesn't know exactly what show he wants to watch, the number of information bits necessary to complete the task of planning an evening watching TV must be at least two (for example, time and channel).

In contrast to computing the task success via AVMs like PARADISE, in which case not completed tasks could implicitely influence the results, our information-bit-approach ensures that task success can only be calculated if the task has been completed.

3. How to deal with a bad performance due to user incooperativity?

One of the main problems of dialogue systems is an incooperative user. We consider only those users to be truly incooperative, who fall out of the role or purposely misuse the system. As an example, a user reading a book to the system or using his mobile phone will be classified as an "incooperative" user. These, of course, are not unique to multimodal system evaluation, but can occur in other situations as well. On a first cue, it is impossible to incorporate incooperative users in an evaluation without lowering task success and thus the system performance. To avoid this, there exist the following approaches:

- Only dialogues with cooperative users are evaluated using empirical methods

- Only dialogues which terminate with finished tasks are evaluated.

Both approaches will be used in conjunction, so that a clearly defined set of data can be evaluated. When deciding to follow the first idea, uncooperative users as defined above, are of course interesting for other than purely empirical reasons. Evaluating the data generated by these incooperative users, in the above sense, in order to improve the system for future development, though, is not part of our aim to judge the quality of the present state of the system SmartKom.

4. How to score multimodal inputs or outputs?

In contrast to interactive unimodal spoken dialogue systems, which are based on many component technologies like speech recognition, text-to-speech, natural language understanding, natural language generation and database query languages, multimodal dialogue systems consist of several such technologies which are functionally similar to each other and therefore could interfere with each other. To make this clear, just imagine the similar functions of ASR and Gesture Recognition: while interacting with a multimodal man-machine interactive system like SmartKom users have the possibility to say what information they want to have and to simultaneously give the same, an additional, or a more specific input by an "interactional gesture" (Steininger et al., 2001). There are several possible problem solving strategies for the system namely:

- First match: the information which was recognized first is taken for further system processing, regardless of the recognition method. This would of course not help in multimodal processing.

- "Mean" match: the system takes the information which is common to both of the recognition modules. This could be called multimodal verification.

- Additional match: take all the information given by several recognizers for further system processing. This would be the best solution, if we assume all recognizers to be highly accurate, which leads us to the next problem:

5. How to weight the several multimodal components of recognition systems?

How can we estimate the accuracy of different recognizers? For example, in talking about speech recognition,

we have to deal with a very complicated pattern match, whereas gesture recognition has a limited set of recognizable gestures which can be found in a given coordinate plane.

It should be clear, that

- the gesture recognizer will be more accurate than the ASR system but
- the ASR system must get a higher weight than the gesture recognition when evaluating the system, since the complexity of the gesture recognition is much lower than the complexity of the ASR sytem.
- Apart from the problems of how to weight the different multimodal system components in an end-to-end evaluation of a multimodal system, there is also the problem of synchrony:
- Are multimodal inputs synchronous or linear within the evaluation? Are inputs from different modalities synchronous, i.e. are they describing the same user intention, although they may not be synchronous in time? Or does the system have to cope with different inputs?

6. PROMISE - A Procedure for Multimodal Interactive System Evaluation

In the last sections we have identified the most characteristic problems which show the need for an extended framework for multimodal dialogue system evaluation. We already gave some examples of possible problem solving strategies. Within this section we will specify these ideas and present the current version of PROMISE. Given the normalized performance function of PARADISE

$$\text{performance} = \alpha\,\mathcal{N}(\kappa) \;-\; \sum_{i=1}^{n} \omega_i\,\mathcal{N}(c_i)$$

with α the weight for task success[1] κ, the assumed Gaussian cost functions [2] c_i weighted by ω_i, and $\mathcal{N}$ z-score normalization function. Weights are defined via a linear multiple regression over κ respectively the costs and the cumulative sum of the user satisfaction scores (see usability questionnaire (Walker and al., 1998)). PROMISE now splits this function in two parts in the way that the formula is reduced to normalized cost functions first. Instead of a multiple linear regression between the free cost variables and the dependent user satisfaction score, PROMISE searches correlations via Pearson correlation between User-Satisfaction - Cost pairs. This means that objective measurable costs will be addressed in the questionnaire to be answered by each user.

Tables 1 and 2 give an overview of the costs we defined in SmartKom, some of them equivalent to the PARADISE costs, some of them extended to deal with multimodality or to weed out user incooperativity.

[1] defined as the successful completion of a duty

[2] either mean or cumulative sum of one cost category (quantity and quality measures); differing from system to system

Quality measures	
system-cooperativity	measure of accepting misleading input
semantics	no.of multiple input possible misunderstandings of input/output semantical correctness of input/output
helps	no. of offered help for the actual interaction situation
recognition	speech facial expression gestures
transaction success	no. of completed sub-tasks
diagnostic error messages	percentage of error prompts
dialogue complexity	task complexity (needed information bits for one task) input complexity (used information bits) dialogue manager complexity (presentation of results)
ways of interaction	gestures/graphics vs.speech n-way communication (several modalities possible at the same time?)
synchrony	graphical and speech output
user/system turns	mixed initiative dialogue management incremental compatibility

Table 1: Quality measures for the SmartKom evaluation

Apart from measuring the quality of dialogue systems in general, like dialogue management (system directed, user directed or mixed initiative), elapsed time of input and output, task completion, mean response time, word count and turn count, we defined measures referring to the problems in section four and five above. Multiple inputs and outputs are scored via the "semantics" cost, to be precised the number of multiple input and the possible misunderstandings of input/output (due to multimodality).

The multimodal components of recognition systems are partly weighted via Pearson correlation using the corresponding user satisfaction scores for the recognition costs defined above. Comparing the accuracies of the different recognition systems for defining a cross-recognizer weight, we calculate "ways of interaction" and "helps". The latter defines the quality and quantity of dynamic help offered by the system in situations where the emotional status of the user changes.

[3] (Oppermann et al., 2001)

Quantity measures	
barge-in	no. of user and system overlap by means of backchanelling, negation of output, further information
cancels	planned system interrupts due to barge-in
off-talk [3]	no. of non-system directed user utterances
elapsed time	duration of input of the facial expression duration of gestural input duration of speech input duration of ASR duration of gesture recognition mean system response time mean user response time task completion duration of the dialogue
rejections	error frequency of input which require a repetition by the user
timeout	error rate of output error rate of input
user/system turns	no. of turns no. of spoken words no. of produced gestures percentage of appropriate/ inappropriate system directive diagnostic utterances percentage of explicit recovery answers

Table 2: Quantity measures for the SmartKom evaluation

The second step is to define another way to calculate the task success.

In PARADISE a set of defined static "keys" was used to measure task success via an attribute value matrix. Since "information bits" (see section two) are used, it makes no sense to calculate an AVM. As described above, these information bits can vary from situation to situation. A successful task is given, if the task was completed according to the necessary number of information bits. A task fails, if it has not been successfully completed. Therefore, we define task success in PROMISE as follows:

$\tau_j = +1$: task success; $\tau_j = -1$: task failure; where j is the index of the corresponding tests.

For each test the corresponding user satisfaction values are Pearson correlated with τ_j.

The system performance results in the following formula:

$$\text{performance} = \alpha\bar{\tau} \; - \; \sum_{i=1}^{n} \omega_i \mathcal{N}(c_i)$$

with α being the Pearson correlation between τ_j(task success) and the corresponding user satisfaction values,
$\bar{\tau}$ the mean value of all τ_j with j index of tests,
n the number of different cost functions,
c_i the assumed Gaussian cost functions - consistently either mean or cumulative sum of one cost category i (measured over all tests)
weighted by ω_i - the Pearson correlation between cost function c_i and defined associated user satisfaction values, and the z-score normalization function $\mathcal{N}(c_i) = \frac{c_i - \bar{c_i}}{\sigma_{c_i}}$, where σ_{c_i} as variance of c_i.

7. Conclusion and future work

Our aim was to roughly define an extended evaluation framework for a multimodal dialogue system evaluation which can deal with multimodal dialogue processing. However, there are still some unresolved or solely unsatisfactorily solved problems dealing with the timing of input in multimodal systems. We are currently specifying different approaches in order to satisfactorily solve the remaining problems which we hope to present at the LREC post-conference workshop on "Multimodal Resources and Multimodal Systems Evaluation" in June.

8. Acknowledgements

This work was funded by the German Federal Ministry for Research and Technology (BMBF) in the framework of the SmartKom project (01IL905E/6).

9. References

D. Oppermann, F. Schiel, S. Steininger, and N. Beringer. 2001. Off-talk - a problem for human-machine-interaction. *Proc. of EUROSPEECH 2001, Scandinavia, Aalborg*.

S. Steininger, B. Lindemann, and T. Paetzold. 2001. Labeling of gestures in smartkom - the coding system. *Springer Gesture Workshop 2001, London (to appear)*, LNAI 2298.

M.A. Walker and al. 1998. Evaluating spoken dialogue agents with paradise: Two case studies. *Computer Speech and Language*, 12.

M.A. Walker, D.J. Litman, C.A. Kamm, and A. Abella. 1997. Paradise: A framework for evaluation spoken dialogue agents. *Annnual Meeting of the Association of Computational Linguistics. ACL*.

MUMIN

A Nordic Network for MUltiModal INterfaces

Patrizia Paggio*, Kristiina Jokinen and Arne Jönsson*****

* Center for Sprogteknologi, Copehagen
email: patrizia@cst.dk
** University of Art and Design, Helsinki
email: Kristiina.Jokinen@uiah.fi
*** University of Linköping
email: arnjo@ida.liu.se

Abstract

This paper reports on a recent initiative undertaken under the language technology programme of the Nordic Research Training Academy (NORFA) to create a network of Nordic research institutes working with multimodal interfaces. In the paper we present the objectives of the network and give an overview of multimodal research and resources in the Nordic countries.

Keywords : multimodal integration, cognitive and usability studies, multimodal dialogue, multimodal research and resources in the Nordic countries

1. Motivation and objectives

In the Nordic countries as elsewhere, both the research and the industrial communities are showing a growing interest in multimodal interfaces. Therefore, there is a need to channel the efforts of individual organisations and countries in joint activities to establish a common research agenda and to define relevant standards and generic methodologies. The aim of the Nordic Network for MultiModal Interfaces (MUMIN) under the NORFA programme, is to stimulate Nordic research in this area and increase its visibility in the international research community. MUMIN, which was established in January 2002 and has funds for two years of activity, has the following goals:

? encouraging joint activities in building generic models and architectures as well as defining standards for the integration and development of multimodal interaction;

? encouraging investigations on the use of multimodality in various practical applications;

? providing a forum for sharing resources and results, and by encouraging network participants to make their research results available via the network's web site;

? organising PhD courses and research workshops on issues related to multimodal interaction;

? creating a network of contacts and a pool of shared knowledge that can be taken advantage of for the definition of collaborative research projects and for product development.

It will also be an important objective of the network to support investigation of the use of multimodal interaction in non-expert environments, and the accessibility of disabled people to IT technology. The network will thus contribute to the Nordic countries' social objectives and help them advance in their vision of building democratic and equal societies for everyone. This also conforms to the EU objectives of creating a user-friendly information society, with accessibility of IT benefits and services for all citizens.

A whole range of research issues, some of which have already received attention from the institutes engaged in the network, are relevant to this overall objective, and constitute topics of interest around which the network's activities will be organised.

2. Multimodal integration

A central issue is that of multimodal integration, which will be approached from different perspectives. A promising approach put forward by several researchers is that of using techniques known from NLP. A similar distinction to that made in NLP between grammar rules and parsing algorithms can be made between a multimodal grammar and an algorithm for applying the grammar to input from multiple modalities. By upholding this separation of process and data, the process of merging inputs from different modalities can be made more general, as the representation becomes media-independent. Furthermore, defining algorithms for modality integration independent of the specific

modalities used in a particular application, increases the chances that components of the system can be extended and reused. For example in the Danish research project Staging, Center for Sprogteknologi (CST) has developed a multimodal dialogue interface to a virtual environment (Paggio *et al*. 2000) where speech, keyboard and gestural inputs are merged by a feature-based parser. Relevant to this issue is also the work carried out by the Speech and Multimedia Communication (SMC) group from Center for PersonKommunikation, Aalborg, which also has extensive research and teaching experience in the area of multimodality, complemented with expertise in speech and image processing (Larsen & Brøndsted 2001).

Another promising approach to modality integration represented in MUMIN is the use of different machine learning techniques, in particular neural networks. As has been the case with many other application domains, also for multimodal integration hybrid systems mixing rule-based approaches with machine learning algorithms, may well provide the most interesting results. Although rule-based methods in general work reasonably well, it is a well-known problem that an explicit specification of the steps, i.e. rules that are required to control the processing of the input, is a difficult task, and when the domain becomes more complex, the rules become more complex too. Often the correlation between input and output is difficult to specify. This is the case e.g. with multimodal interfaces, and thus approaches which are both robust and able to adapt to new inputs are needed. Expertise in this domain is brought to the network by the Media Lab at the University of Art and Design in Helsinki (UIAH), and especially its Soft-Computing Interfaces Group which is devoted to designing adaptive interfaces and developing tools for human-machine interaction, relying on nature-like emergent knowledge that arises from subsymbolic, unsupervised processes of self-organizing nature (see e.g. Jokinen *et al*. 2001).

In a similar way as the rule-based integration of modalities can be enhanced using machine learning techniques, results obtained through pure probabilistic analysis methods may well be boosted by the addition of symbolic rules. An example relevant to multimodal interfaces are the algorithms for character and word prediction used in connection with eye-tracking, where the system tries to guess what the user is "typing with the eye". Although the performance of the probabilistic approaches implemented in current systems is promising, language technology techniques seem to constitute a valuable add-on. This is an issue that the IT University of Copenhagen is working on.

3. Neurocognitive basis and usability studies

To fully exploit multimodality in various interfaces, it is important to know how the neurocognitive mechanisms support multimodal and multisensory integration. In comparison to that devoted to single sensory systems, there has been very little research on the integration mechanisms of information received via different senses. However, the research group of Cognitive Science and Technology at the Helsinki University of Technology is using various methods to uncover the neurocognitive principles of multisensory integration with the purpose of developing mathematical models of this integration. The group is also developing a Finnish artificial person – a talking and gesturing audiovisual head model – which will serve as a well-controlled stimulus for neurocognitive studies (Sams *et al*. 1998).

A related issue is that of the impact of multimodality on the users. Relevant questions are which input and output modalities should be used for which task, which are the best combinations, and how different modalities are used by or for users with different degrees of expertise. There are in general two ways to use multimodal input: to react directly to the user's intentional input and to observe the user's unconscious use of certain modalities (e.g. eye-gaze). The former method is based on direct control and has been used in earlier conversational interface prototypes. Observing the user and understanding their intentions and mental states has not been extensively studied, and would add valuable information to the design of multimodal systems. The Computer Human Interaction group at the University of Tampere (TAUCHI) will bring to the network its extensive expertise in the design and use of innovative user interfaces and in usability testing, as well as the agent-based development platform Jaspis (Turunen and Hakulinen 2000). The Natural Interactive Systems Laboratory (NISLab) at the University of Southern Denmark, has also made pioneering contributions to the theoretical understanding of unimodal input/output modalities, and of the multiple conditions which determine the usability of individual modalities and their combinations (Bernsen 2001).

4. Multimodality and dialogue

A third issue, which encompasses a great deal of research work carried out by several of the network members, is that of multimodal dialogue. In this respect, it is interesting to note that the growing interest in multimodal interaction is opening a new perspective to Nordic research on dialogues, which is already acknowledged internationally. Several institutes in the Nordic countries have in fact contributed substantially to dialogue research, and developed dialogue models as well as implemented dialogue systems.

The Department of Linguistics at the University of Göteborg has extensive experience in corpus collection and dialogue management. They have developed tools for spoken language analysis and coding which can be applied to the collection and analysis of multimodal dialogues, thus providing empirical basis and insight for research on multimodal interaction: how different modalities are used in human-human communication (Allwood, 2001). NISLab has a strong background in dialogue management, dialogue systems evaluation, and spoken dialogue corpus coding from a number of EU projects. NISLab is currently addressing best practice in the development and evaluation of natural interactivity systems and components; surveying data resources, coding schemes and coding tools for natural interactivity;

and building a general-purpose coding tool for natural interactive communicative behaviour. The natural language processing research group (NLPLab) at the University of Linköping has for almost two decades conducted research on dialogue systems and now has a platform for the development of multimodal dialogue systems for various applications to be developed further towards an open source code repository (Degerstedt & Jönsson, 2001). Current focus is on integrating dialogue systems with intelligent document processing techniques in order to develop multimodal dialogue systems that can retrieve information from unstructured documents, where the request requires that the user, in a dialogue with the system, specifies their information needs (Merkel & Jönsson, 2001). Finally, the Centre for Speech Technology at the Royal Institute of Technology in Stockholm (KTH) has developed several multimodal dialogue systems with the motivation of studying speech technology as part of complete systems and the interaction between the different modules that are included in such systems. Their first system, Waxholm, was a multimodal system exploring an animated agent (Carlsson & Granström, 1996). Current work and interests involve research on multimodal output using animations and also to some extent multimodal input using both speech and pointing (Gustafson *et al*. 2000).

5. Participating groups

The groups participating in MUMIN are shown in Table 1 below. Currently, three Nordic countries are represented, but the network is interested in welcoming participants from other Nordic countries, as well as in cooperation with non-Nordic countries.

Denmark	Finland	Sweden
Center for Sprogteknologi, Copenhagen	University of Art and Design Helsinki , Media Lab, Helsinki	Linköpings Universitet, Institutionen för datavetenskap, Linköping
Syddansk Universitet , NISLab , Odense	University of Tampere, Department of Computer and Information Sciences, Tampere Unit for Computer-Human Interaction (TAUCHI), Tammerfors	KTH, CTT, Centre for Speech Technology, Stockholm
Aalborg Universitet , Center for PersonKommunikation, Aalborg	Helsinki University of Technology , Laboratory of Computational Engineering, HUT Espoo	Göteborgs Universitet, Institutionen för lingvistik, Göteborg
IT-Højskolen, Eye Gaze Research Team, Copenhagen		

Table 1: The groups participating in MUMIN

6. Conclusion

The MUMIN network is expected to play an important strategic role in the establishment of a common research agenda for Nordic researchers working with multimodal interfaces, but also to relate its activities to those of the international community, and to contribute to the general progress in the area. Therefore, it is highly relevant for MUMIN to participate in this workshop, and to provide a Nordic contribution to the discussion of a multimodal roadmap.

7. References

Allwood, J. (2001) Dialog Coding - Function and Grammar, Göteborg Coding Schemas, *Gothenburg Papers in Theoretical Linguistics, GPTL 85*. Göteborg University, Department of Linguistics.

Bernsen, N. O. (2001) Multimodality in language and speech systems - from theory to design support tool. Chapter to appear in Granström, B. (Ed.): *Multimodality in Language and Speech Systems.* Dordrecht: Kluwer Academic Publishers.

Carlson, R., Granström, B. (1996) The WAXHOLM spoken dialogue system. *Acta universitatis Carolinae philologica* 1, pp. 39-52.

Degerstedt, L. and Jönsson, A. (2001) A Method for Iterative Implementation of Dialogue Management , *IJCAI Workshop on Knowledge and Reasoning in Practical Dialogue Systems*, Seattle.

Gustafson, J, Bell, L, Beskow, J, Boye, J, Carlson, R, Edlund, J, Granström, B, House, D & Wirén M (2000) AdApt - a multimodal conversational dialogue system

in an apartment domain, In *Proc of ICSLP 2000*, Beijing, 2:134-137

Jokinen, K., Hurtig, T., Hynnä, K., Kanto, K., Kaipainen, M., and Kerminen, A. (2001) Dialogue Act classification and self-organising maps. In *Proceedings of the Neural Networks and Natural Language Processing Workshop* , Tokyo, Japan.

Larsen , L.B. and Brøndsted, T. (2001) A Multi Modal Pool Trainer. In *Proceedings of the International Workshop on Information Presentation and Natural Multimodal Dialogue IPNMD-2001*. Verona, Italy 14-15 December 2001, pp. 107-111.

Merkel, M. and Jönsson, A. (2001) Towards multimodal public informations systems , *Proceedings of 13th Nordic Conference on Computational Linguistics, NoDaLiDa '01*, Uppsala, Sweden.

Paggio P., Jongejan B. and Madsen C.B. (2000) Unification-based multimodal analysis in a 3D virtual world: the Staging project. In *Proceedings of the CELE-Twente Workshop on Language Technology: Interacting Agents*, pp. 71-82.

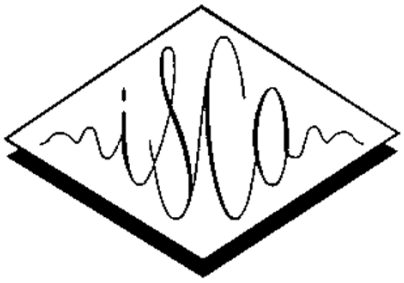
Sams , M., Manninen, P., Surakka , V., Helin , P. and Kättö, R. (1998) Effects of word meaning and sentence context on the integration of audiovisual speech. *Speech Communication*, 26, 75-87.

Turunen, M. and Hakulinen J. (2000) JASPIS - a Framework for Multilingual Adaptive Speech Applications. In P*roceedings of the 6th International Conference of Spoken Language Processing. ICSPL 2000*.

Support

LREC 2002

Co-operating Organisation



ISCA SALTMIL SIG: “Speech and Language Technology for Minority Languages”

The Workshop Programme

Saturday, 1st June 2002

14:30	Registration, and preparation of posters	
Oral Session: Portability Issues in Human Language Technologies		
14:50	Workshop Welcome and Introduction	Bojan Petek, University of Ljubljana, Slovenia
14:55	Multilingual Time Maps: Portable Phonotactic Models for Speech Technology	Julie Carson-Berndsen, University College Dublin, Ireland
15:20	Units for Automatic Language Independent Speech Processing	Jan Černocký, Brno University of Technology, Czech Republic
15:45	Some Issues in Speech Recognizer Portability	Lori Lamel, LIMSI-CNRS, France
16:10	Seven Dimensions of Portability for Language Documentation and Description	Steven Bird* and Gary Simons**, * Linguistic Data Consortium, University of Pennsylvania, USA ** SIL International, USA
16:35	Break	
Oral Session: HLT and the Coverage of Languages		
17:00	Challenges and Opportunities in Portability of Human Language Technologies	Bojan Petek, University of Ljubljana, Slovenia
17:25	The Atlantis Observatory: Resources Available on the Internet to Serve Speakers and Learners of Minority Languages	Salvador Climent*, Miquel Strubell*, Marta Torres*, and Glyn Williams** *Universitat Oberta de Catalunya, Spain **Foundation for European Research, Wales, Great Britain
17:50	Towards the Definition of a Basic Toolkit for HLT	Kepa Sarasola, University of the Basque Country, Spain
Poster Session		
18:15	Poster Session (see next page)	
20:00	End	

(continued next page)

(continued)

Poster Session	
Ubiquitous Multilingual Corpus Management in Computational Fieldwork	Dafydd Gibbon, Universität Bielefeld, Germany
A Theory of Portability	Hyo-Kyung Lee, University of Illinois at Urbana-Champaign, USA
A Requirement Analysis for an Open Set of Human Language Technology Tasks	Fredrik Olsson, Swedish Institute of Computer Science, Sweden
Taking Advantage of Spanish Speech Resources to Improve Catalan Acoustic HMMs	Jaume Padrell and José B. Mariño, Universitat Politècnica de Catalunya, Spain
Portability Issues of Text Alignment Techniques	António Ribeiro, Gabriel Lopes and João Mexia, Universidade Nova de Lisboa, Portugal
SPE Based Selection of Context Dependent Units for Speech Recognition	Matjaž Rodman*, Bojan Petek* and Tom Brøndsted** *University of Ljubljana, Slovenia **Center for PersonKommunikation (CPK), Aalborg University, Denmark
VIPTerm: The Virtual Terminology Information Point for the Dutch Language. A Supranational Project on Terminology Documentation and Resources.	Frieda Steurs, Lessius Hogeschool, Belgium

Workshop Organisers

Julie Carson-Berndsen, University College Dublin, Ireland
Steven Greenberg, International Computer Science Institute, USA
Bojan Petek, University of Ljubljana, Slovenia
Kepa Sarasola, University of the Basque Country, Spain

Workshop Programme Committee

Julie Carson-Berndsen, University College Dublin, Ireland
Steven Greenberg, International Computer Science Institute, USA
Bojan Petek, University of Ljubljana, Slovenia
Kepa Sarasola, University of the Basque Country, Spain

Table of Contents

Author Index

Multilingual Time Maps: Portable Phonotactic Models for Speech Technology

Julie Carson-Berndsen

Department of Computer Science
University College Dublin
Belfield, Dublin 4, Ireland
Julie.Berndsen@ucd.ie

Abstract

This paper addresses the notion of portability of human language technologies with respect to a computational model of phonology known as the *Time Map model*, focusing specifically on generic techniques for acquiring, representing and evaluating specific phonological information used by the model in multilingual speech technology applications. *Multilingual time maps* are multilevel finite state transducers which define various types of information with respect to their phonotactic context. A development environment for *multilingual time maps* is presented and an illustration of how such a multilevel finite state transducer can be constructed for a new language is given.

1. Introduction*

The extent to which human language technologies can be adapted for use in other application domains has become obvious in recent years with speech interfaces to a wide variety of information systems becoming more and more commonplace. However, the extent to which such technologies can be adapted to other languages, in particular minority languages, remains to be seen. Furthermore, little emphasis has been placed on developing generic technologies which can be applied not only to "new" languages but which can be employed in other task domains. While it is often considered practical to have a speech recognition system for a language if one is about to embark on developing a speech synthesis system for that language - the recognition can support data annotation - the fact that the linguistic knowledge which is being represented for the one task domain could be relevant for the other is largely ignored. In order to address the issue of portability adequately, linguistic representations must be integrated more explicitly into human language technologies. While a hidden Markov model can be trained as a speech recognizer perhaps for any language given a large data set, there is no potential for exploiting the commonalities of human languages or for using the same knowledge in the synthesis task domain.

This paper addresses the notion of portability with respect to a computational model of phonology known as the *Time Map model* focusing specifically on generic techniques for acquiring, representing and evaluating different types of phonological information used by the model in multilingual speech technology applications. Ubiquitous language technology concerns the development of language technologies for different purposes on different platforms so that they can be made available to everybody at all times rather than to a select group for specific purposes. Much of the further development of the *Time Map model* is aimed towards providing fine-grained representations for speech recognition and synthesis and developing computational models which will contribute to achieving this long-term goal. The techniques presented in this paper make way for the extension of current speech technology to languages which have received little attention thus far by modeling linguistic information at various levels of granularity.

In the context of this paper, portability refers to extending the functionality of a system to cater for another language. It does not cover issues such as re-applying the technology to new content domains (e.g. adapting an English spoken language interface for a football results information system to an English information system for accommodation in London). The model described below is not restricted to a specific application domain and therefore the main concern is adapting the system to another language. This involves parameterization of the system so that the language-specific components can be substituted in a "plug and play" fashion.

In the next section, the *Time Map model* is sketched briefly with particular attention to the language-specific knowledge components. Section 3 discusses how the model has been parameterized to allow extension to other languages by defining the notion of *multilingual time maps* specifying information at different levels of granularity and a development environment, *PhonoDeSK*, for acquiring and evaluating such *time maps* is presented. Section 4 describes an example illustrating the role which can be played by *PhonoDeSK* in the context of portability of human language technologies and section 5 concludes with some comments on future work.

2. Time Map Model

The *Time Map model* was proposed as a computational linguistic model for speech recognition by Carson-Berndsen (1998, 2000) and has been tested within a speech recognition architecture for German. More recently, the model has been extended to English and has been provided with an interface which allows users to define and evaluate phonotactic descriptions for other languages and sublanguages (Carson-Berndsen & Walsh, 2000). In extending the model to cater for

* This research was part funded by the HEA under the MMRP programme.

English, particular emphasis was placed on parameterizing the model so that knowledge components for other languages could readily be substituted.

The original motivation for the design of the *Time Map model* was to address specific problems in the area of speech recognition below the level of the word. In particular, the problem of out-of-vocabulary items, also termed the "new word" problem, is addressed explicitly in the model. This is done by including *complete* phonotactic descriptions of a language which describe not only those forms specified in some corpus lexicon, but also all potential forms which adhere to the phonotactic constraints imposed by the language. Another specific problem addressed by this approach is the modelling of coarticulation phenomena. This is done by assuming a non-segmental approach to the description and interpretation of speech utterances which avoids having to segment an utterance into non-overlapping units at any level of representation.

The *Time Map model* has two main language-specific components: the *phonotactic automaton* and the *time map lexicon*. These components each assume a particular representation of speech utterances in terms of a multilinear representation of features similar to an autosegmental score.

2.1. Multilinear Representations

Speech utterances are defined in the model in terms of a multilinear representation of tiers of features which are associated with signal time. The notion of tiers of features is not new in the area of phonology (cf. for example Goldsmith, 1990). However, recently there has been a significant upsurge in phonetic feature extraction and classification, and automatic transcription using the type of features proposed in our model (e.g. Chang, Greenberg & Wester (2001), Ali et al., (1999)). An example multilinear event representation using the Chang, Greenberg & Wester (2001) features is depicted in figure 1.

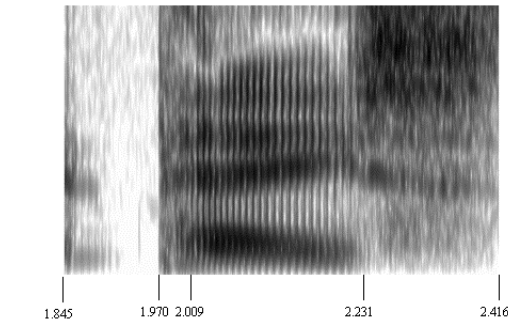


Figure 1: Multilinear representation of the word *pace*

As can be seen from figure 1, each feature in a multilinear event representation is associated with a specific tier (on the vertical axis) and with a specific time interval in terms of milliseconds (on the horizontal axis). The features do not all start and end simultaneously. An overlap of properties (coarticulation) exists in any time interval; for example, the feature *rd-* begins before the *voc* feature indicating that the lips have been spread during the plosive (*stp*) anticipating the following nonround vowel.

A multilinear event representation of a speech utterance is in fact highly constrained. It is not the case, that any combination of features can occur in any order. The allowable combinations of features are dictated partly by the phonological structure of the language, as defined by the phonotactics, and partly by predictable phonetic variation, which often results from limitations associated with human speech production.

2.2. Phonotactic Automata

The primary knowledge component of the *Time Map model* is a complete set of phonotactic constraints for a language which is represented in terms of a finite state automaton. A *phonotactic automaton* describes all permissible sound combinations of a language within the domain of a syllable. It can be phoneme-based (just specifying phonemes), feature-based (generalizing over phonemes) or event-based (specifying constraints on temporal relations between the features). A subsection of a phonotactic automaton depicting CC- clusters in English syllable onsets can be seen in figure 2.

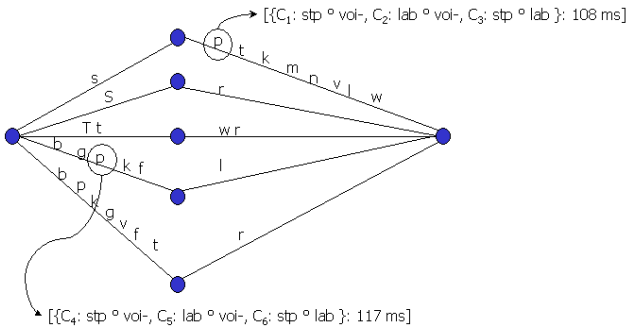


Figure 2: Subsection of a phonotactic automaton

The arcs in an event-based phonotactic automaton define a set of constraints on overlap relations which hold between features in a particular phonotactic context (i.e. the structural position within the syllable domain).[1] In the phonotactic automaton of figure 2, the constraint *C_1: stp ° voi-*, for example, states that the feature *stp* (a plosive) on the manner tier should overlap the feature *voi-* (voiceless) on the phonation tier. The millisecond values refer to the average durations for the sounds in this particular phonotactic context which have been calculated from a large corpus.

[1] The monadic symbols written on the arcs in figure 2 are purely mnemonic for the feature overlap constraints they represent; the ° symbol represents the overlap relation.

2.3. Time Map Lexicon

The *time map lexicon* defines fully specified multilinear event representation of each syllable in the corpus (or each lexicalised syllable in the language) together with their phonemic and orthographic forms. The *time map lexicon* is used online by the model to distinguish between actual and potential syllables and used offline for evaluation purposes with respect to a particular corpus.

The *time map lexicon* is compiled from a generic lexicon model (Carson-Berndsen, 1999). Generic lexical information is represented in DATR, a simple language designed specifically for lexical knowledge representation that allows the definition of nonmonotonic inheritance networks with path/value equations (cf. Evans & Gazdar, 1996). Varying degrees of granularity (syllables, tiers in a multilinear representation, consonants, vowels etc.) are specified as templates in DATR. For each language (see figure 3) specific word, syllable and segment inventories are defined which contain information such as frequency and average duration. Specific entries inherit regularities and sub-regularities from the templates while exceptions are specified in the entries themselves. Either individual or cascades of finite state transducers are then applied to generate individual lexicons for speech applications in an application specific format (cf. Cahill, Carson-Berndsen & Gazdar, 2000).

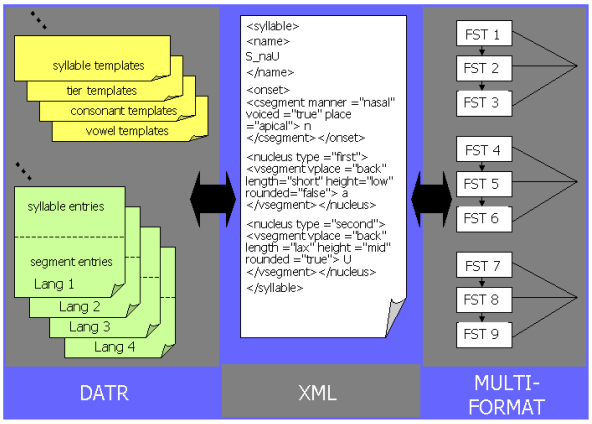


Figure 3: Generic lexicon architecture

2.4. Speech Recognition with the Time Map Model

In the context of speech recognition, input to the model is a multilinear representation of a speech utterance in terms of absolute time events, i.e. features with start and end points which are extracted from the speech signal. Phonological parsing in the *Time Map model* is guided by the *phonotactic automaton* which provides top-down constraints on the interpretation of the multilinear representation, specifying which overlap and precedence relations are expected by the phonotactics. If the constraints are satisfied, the parser moves on to the next state in the automaton. Each time a final state of the automaton is reached, a well-formed syllable has been found. This well-formed syllable may be underspecified, however, since some of the constraints in the phonotactic automaton may have been relaxed. It is then compared with a fully specified multilinear representation in the *time map lexicon* which allows the system to distinguish between actual (lexicalised) and potential syllables. The architecture of the model in the context of speech recognition is depicted in figure 4.

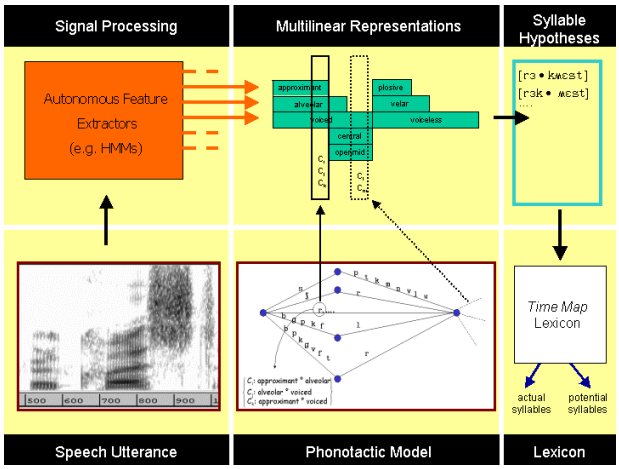


Figure 4: *Time Map* speech recognition architecture

Speech synthesis based on the *Time Map model* is also currently under investigation (see Bohan et al., 2001). This involves generating multilinear representations from a lexical representation of an utterance using a cascade of finite state transducers mapping from phonemes to allophones to event representations. The aim of this research is to investigate the application of the *Time Map model* in the synthesis domain and to a language with a significantly different phonology, namely Irish. The methodology used to port the model to Irish is discussed in section 4 below.

3. Multilingual Time Maps

The *Time Map model* has been parameterized to allow the language-specific components to be substituted by components describing other languages. There are two issues involved in this process. The first issue concerns how to represent the language-specific information in a uniform way so that it can be used immediately by the model and also be made available for use with other technologies. The second issue concerns the acquisition of the language-specific components. In what follows, both of these issues are discussed in turn with respect to the *phonotactic automaton* and the *time map lexicon*. The language-specific configuration of the model is defined by a *multilingual time map*. The *time map* defines mappings between different types of information and constraints on overlap relations between features. It is termed *multilingual* because on the one hand, it provides a framework for developing the language-specific knowledge components for the *Time Map model* either by using knowledge of a related language already available to the system to predict the relevant structures of a "new" language or by learning these directly. On the other hand, it has a uniform structure which allows for cross-language comparisons and the generation of *time maps* which cover a number of languages.

3.1. Representation

A *multilingual time map* comprises language-specific information at various levels of granularity represented as a multilevel finite state transducer. The advantage of this representation is that it is declarative, bidirectional and efficient to process. The multilevel finite state transducer can be viewed as an extension of the phonotactic automaton to include (at least) the following levels:

1. Graphemes
2. Phonemes
3. Allophones
4. Features
5. Constraints on Overlap Relations
6. Average Duration
7. Frequency
8. Probability

Each arc specifies information on all of these levels (although some of this information may not be available in all cases but can be readily updated at any time). For example, figure 5 depicts a single arc of a *multilingual time map* for English.

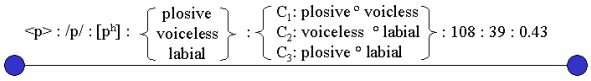


Figure 5: Arc in a *multilingual time map*

The first level is the grapheme level, the second is the phoneme level, the third is the allophone level (i.e. *p* is aspirated in this phonotactic context). The fourth level is the feature level specifying the features for this phoneme (which can be selected from a number of different possible feature sets). The fifth level specifies the constraints on the overlap relations between the features; the sixth level specifies the average duration of the [p] in this phonotactic context. The seventh level specifies the frequency of this sound in this phonotactic context and the eighth level specifies the probability of the arc.

A generic transducer interpreter[2] is used to extract the levels required for different purposes from the *multilingual time map*. To construct the phonotactic automaton for a speech recognition application of the *Time Map model*, for example, the generic transducer interpreter takes level 2 as input and outputs level 5 and level 6 from the transducer. Note that it is also possible to map between other levels in the transducer to obtain other types of information (e.g. input graphemes and output phonemes; input phonemes and output allophones etc.).

3.2. Acquisition

The real challenge for the portability of the *Time Map* model lies in acquisition of the *multilingual time maps* (i.e. not just to be able to use them to generate the *phonotactic automaton* and the *time map lexicon* for a particular language, but to be able to construct them efficiently). *PhonoDeSK* (see figure 6) is a suite of tools which has been designed specifically for acquiring and evaluating *multilingual time maps* (see Ashby, Carson-Berndsen & Joue, (2001 for an initial specification). These tools are used by *PhonoDeSK agents*[3] which collaborate with each other in order to define an optimal phonological description of the language.

PhonoDeSK foresees three strategies for structured data acquisition; user-driven, data-driven and data-driven with user prompting. That is to say, *multilingual time maps* can either be produced manually by a trained linguist or can be learned from a data set with or without user intervention. *PhonoDeSK* is web-based and can thus be accessed anywhere at any time. The user is also viewed as an agent in the context of *PhonoDeSK* – the *verification agent*.

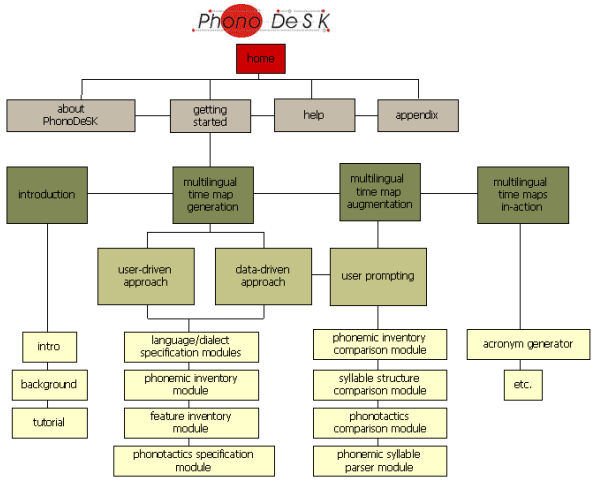


Figure 6*: PhonoDeSK*

When constructing a *multilingual time map* for a "new" language, a number of inventories are created by an *inventory agent:*

1. Phoneme Inventory
2. Allophone Inventory
3. Feature Inventory
4. Syllable Inventory

In each case, any available resources may be used directly. For example, a phonemically labeled data set can be used to extract the phoneme inventory and, together with a *learning age*nt for phonotactic automata, PAL (Kelly, 2001), to predict the syllable inventory. Using an existing *multilingual time map* for a related language, predictions may be made which can be accepted or rejected by a native speaker (*verification agent*) of the language. The acceptances/rejections are then incorporated into the learning procedure. If no resources whatsoever are available for the "new" language then much more manual input is required by the user. The first pass *multilingual time map*

[2] This has been implemented by Robert Kelly, University College Dublin.

[3] The notion of agent will not be discussed further in this paper. Further details on the agent approach assumed here can be found at http://said.ucd.ie.

constructed using *PhonoDeSK* will be, in general, underspecified on some of the transducer levels.

This section has discussed *multilingual time maps*, how they are represented and how they can be acquired for new languages. The next section illustrates how a *multilingual time map* can be constructed using *PhonoDeSK agents,* taking Irish as an example.

4. An Example

In this section, an example of a *multilingual time map* is presented which has been constructed from an initial corpus of tri-syllabic Irish words. This is work in progress and therefore as stated above not all levels in the *time map* are fully specified at present. In *PhonoDeSK*, a *phonotactic agent* and a *lexicon agent* collaborate with each other and with other learning and generalization agents to construct a *multilingual time map* which can be used with the *Time Map model* for speech recognition and synthesis.

4.1. Observation

The corpus was recorded and labeled phonetically including syllable boundaries and a distinction between stressed and unstressed syllables was made. The phoneme inventory and the feature inventory were specified manually by an expert on Irish phonology (a human *verification agent* for Irish). The corpus was input to the *learning agent*. The first pass produced a deterministic phonotactic automaton which included average durations for each sound in each phonotactic context, frequency of each sound in each phonotactic context with respect to the corpus and a probability of each arc. The initial *multilingual time map* for Irish specifies levels 2 to 8.

Since this initial *multilingual time map* specifies all the forms in the corpus, it automatically contains all the forms which should be included in the *time map lexicon*. Here a distinction is being made between a corpus lexicon and a complete lexicon of the Irish language. The *lexicon agent* uses all the information which is available in the *multilingual time map* to define new syllable and segment entries together with syllable, tier, consonant and vowel templates in the DATR-based lexicon.

4.2. From Observation to Generalization

The initial *multilingual time map* is input to a *generalization agent* which extends and optimizes the *time map*. The *generalization agent* interacts with 4 other agents until an optimal description is reached.

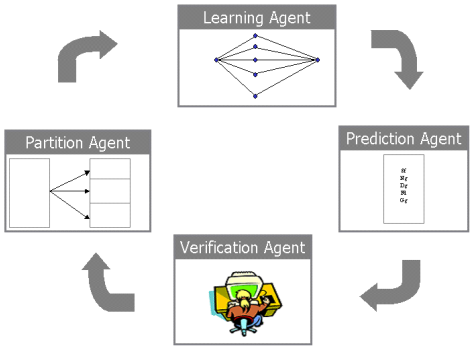


Figure 7: Generalisation cycle

For the purposes of generalization over substructures, the data is partitioned into initial consonant cluster (onset), vowel and final consonant cluster (rhyme) and further into CC- onsets; CCC-onsets etc. A phonotactic automaton for each of the partitions is learned separately by the *learning agent*. An example for the resulting structure for stressed syllable CC- onsets in Irish is depicted in figure 8, whereby only level 2 (phonemes) is specified. Note that Irish has both palatalized (represented in this figure by uppercase) and plain consonants.

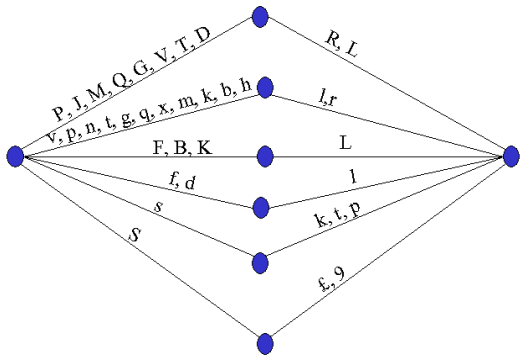


Figure 8: Learned CC- onset of Irish syllable

This onset automaton has been learned purely on the basis of the data set. It does not claim to cover all CC-onsets of Irish, only those represented in the data. In order to extend this to a complete onset description, (idiosyncratic) gaps must be identified in the representation which could also be permissible onsets of the language. There are two methods for identifying idiosyncratic gaps in the automaton. The first involves examining general distributional properties evident in the automaton. To the eye, one possible gap is obvious: there is a path representing the combinations [fl] and [dl] and [f] and [d] stand out as being the only plain consonants followed by [l] but not followed by [r] in the onset. The *prediction agent* identifies such gaps and the combinations [fr] and [dr] are presented to native speakers of the language (*verification agents*) to verify whether these are permissible combinations or not. The arcs in the *multilingual time map* are then generalized with respect to the feature level (level 4) in order to determine the commonalities between the phonemes in a particular phonotactic context. The *prediction agent* requests a *phonoclass agent* to group phonemes into natural classes (based on the intersection of the their features). Using the complete phoneme and feature inventories, the *prediction agent* presents other phonemes which are part of the natural class but are not found in that phonotactic position to the *verification agent*.

This is performed until all partitions of the data have been generalized or until the verification agent decides that the *multilingual time map* is optimal[4].

[4] In this context, optimal means deemed suitable for use in some application.

4.3. Alternative Routes

The description of the *multilingual time map* construction using *PhonoDeSK* has assumed thus far that phonemically labeled data is available as a starting point for learning. Clearly, this will not always be the case and there are two alternative routes which can be taken. Firstly, it may be possible to predict using the *multilingual time map* of a related language, what phonotactic combinations are permissible in the new language. These can be input directly to the *prediction agent* and the forms can be accepted or rejected by the *verification agent*. Secondly it is also possible for the user to specify the canonical form of the syllable and the phoneme inventory of the "new" language and this will be used by the *prediction agent* to elicit permissible combinations from the *verification agent*.

There are a number of additional tools available in *PhonoDeSK* which support the *verification agent* in constructing a new *multilingual time map* from an existing *time map*: for example, all possible forms represented by the *time map* can be generated; two descriptions may be compared directly with each other at the phoneme and syllable level (cf. Ashby, Carson-Berndsen & Joue, 2001); a single parse or all possible parses of a given phonemic representation can be generated and presented to the user for verification. The *verification agent* thus provides important information for preferences which are used in turn to update the probability of a particular parse.

The level which remains to be included in the *multilingual time map* after generalization of the phonotactics is complete is the grapheme level (level 1). This task is performed by the *lexicon agent*. The phonemic forms are presented to the *verification agent* to elicit a correct orthographic form for the lexicon, possibly using the *prediction agent* to suggest mappings based on the original corpus. Once the orthographic forms are available in the lexicon, the *phonotactic agent* requests the *learning agent* to learn the grapheme-phoneme mapping of the words in the corpus. This can later be used to estimate a grapheme-phoneme mapping for new forms.

5. Conclusion

This paper has presented the concept of a *multilingual time map* which has evolved out of a desire for portability of a computational phonological model for use in various human language technology tasks. The development environment, *PhonoDeSK*, has been designed specifically for acquiring, representing and applying phonological information at various levels of granularity. It combines finite state techniques with automatic and manual data acquisition through the use of agents which collaborate to instantiate the various levels of the *multilingual time map*. The *multilingual time maps* have been designed specifically for use with the *Time Map model* but they also represent an important step on the road to the realisation of ubiquitous language technology in general, by providing a framework which allows portability to new languages. However, the information represented in the *multilingual time maps* can be used directly by other technologies for structural fine tuning. Future work is concerned with extending *PhonoDeSK* agents and with applying the technology to other languages.

6. References

Ali, A.M..A.; J. Van der Spiegel; P. Mueller; G. Haentjaens & J. Berman (1999): An Acoustic-Phonetic Feature-Based System for Automatic Phoneme Recognition in Continuous Speech. In: *IEEE International Symposium on Circuits and Systems (ISCAS-99)*, III-118 - III-121, 1999.

Ashby, S.; J. Carson-Berndsen, & G. Joue (2001): A testbed for the development of multilingual phonotactic descriptions. In: *Proceedings of Eurospeech 2001*, Aalborg.

Bohan, A.; E. Creedon, J. Carson-Berndsen & F. Cummins (2001): Application of a Computational Model of Phonology to Speech Synthesis, In: *Proceedings of AICS2001*, Maynooth, September 2001.

Cahill, L; J. Carson-Berndsen & G. Gazdar (2000), Phonology-based Lexical Knowledge Representation. In: F. van Eynde & D. Gibbon (eds.) *Lexicon Development for Speech and Language Processing*, Kluwer Academic Publishers, Dordrecht.

Carson-Berndsen, J. (1998): *Time Map Phonology: Finite State Models and Event Logics in Speech Recognition*. Kluwer Academic Publishers, Dordrecht, 1998.

Carson-Berndsen, J. (2000): Finite State Models, Event Logics and Statistics in Speech Recognition, In: Gazdar, G.; K. Sparck Jones & R. Needham (eds.): *Computers, Language and Speech: Integrating formal theories and statistical data*. Philosophical Transactions of the Royal Society, Series A, 358(1770), 1255-1266.

Carson-Berndsen, J. (1999): A Generic Lexicon Tool for Word Model Definition in Multimodal Applications. *Proceedings of EUROSPEECH 99, 6th European Conference on Speech Communication and Technology*, Budapest, September 1999

Carson-Berndsen, J. and Walsh, M. (2000): Generic techniques for multilingual speech technology applications, *Proceedings of the 7th Conference on Automatic Natural Language Processing*, Lausanne, Switzerland, 61-70.

Chang, S.; S. Greenberg & M. Wester (2001): An Elitist Approach to Articulatory-Acoustic Feature Classification. In: *Proceedings of Eurospeech 2001*, Aalborg.

Evans, R & G. Gazdar (1996), DATR: A language for lexical knowledge representation. In: *Computational Linguistics* 22, 2, pp. 167-216.

Goldsmith, J. (1990): *Autosegmental and Metrical Phonology*. Basil Blackwell, Cambridge, MA.

Kelly, R. (2001): PAL: Phonotactic Automaton Learning. Technical Report, Department of Computer Science, University College Dublin.

Units for Automatic Language Independent Speech Processing

Jan Černocký

Brno University of Technology, Faculty of Information Technology
Božetěchova 2, 61266 Brno, Czech Republic
cernocky@fit.vutbr.cz

Abstract

Many current systems for automatic speech processing rely on sub-word units defined using phonetic knowledge. Our paper presents an alternative to this approach – determination of speech units using ALISP (Automatic Language Independent Speech Processing) techniques. Such units were experimentally tested in a very low bit rate phonetic vocoder, where mean bit rates of hundreds bps for unit encoding were achieved. Improvements of the proposed coder and some links to "classical" approaches of speech synthesis are discussed. Based on the results of comparison of an ALISP segmentation with a phonetic alignment, we comment on the potential use of automatically derived units in speech recognition, speaker verification and language identification.

1. Introduction

The International Phonetic Association (IPA) sets up as one of its objectives the definition of a symbolic representation of speech for any of the speakers of any language in the world: the International Phonetic Alphabet[1]. However, despite efforts devoted to this topic, some substantial problems persist in the adequacy of this alphabet for spoken speech.

Recent advances in ALISP (Automatic Language Independent Speech Processing) (Chollet et al., 1999) led us to the idea of defining such a set of units *automatically*, without an a-priori knowledge; to let it emerge uniquely from the speech data. For this purpose, a number of tools which proved their efficiency in automatic speech processing (coding, recognition, synthesis, language identification, speaker verification) have been developed: temporal decomposition (TD), non-supervised clustering, Hidden Markov Models (HMM) and others. Basic information about these tools with references are given in Section 2.

On contrary to IPA, where it is difficult to find an objective criterion, the set of units can be evaluated using *very low bit rate (VLBR) speech coding* at about 200 bps (Černocký et al., 1998). At these rates, a symbolic representation of the incoming speech is required. If the decoded speech is intelligible, one must admit that the symbolic representation is capable of capturing the significant acoustic-phonetic structure of the message. Moreover, the coding rate in bps and dictionary size give an idea of *efficiency* of the description while the quality of decoded speech is related to its *precision*. Section 3. gives an overview of our VLBR coding experiments in three languages and their results. It also contains a description of recent advances in VLBR coding using ALISP units.

However, the domain with the greatest need of optimized and automatically derivable units is the large vocabulary continuous speech recognition (LVCSR) based in current systems on phones or their derivatives (context-dependent phones, syllables). Section 4. presents a comparison of two alignments of data: phonetic and ALISP in terms of a confusion matrix. It also contains some reflections on encoding of target vocabulary using data-driven units.

Section 5. contains conclusions and some comments on the use of ALISP units in other domains (speaker verification and language identification).

[1] http://www.arts.gla.ac.uk/IPA/ipa.html

2. ALISP tools

Classical speech processing suffers from the need of large phonetically labeled, or at least orthographically annotated corpora. This is making the current algorithms unpractical when used in a condition for which a database does not exist, or is very costly (rare language, environmental noise, channel, application domain). The main goal in ALISP processing is to find data-driven units with as little supervision as possible. We will see, that for coding, the process can be fully automated. Steps that should be taken to use such units in recognition are discussed in section 4. The expected result of unit creation is:

- a set of units (that can be compared to a set of phonemes).
- labeling of the training data using those units.
- models of units to detect them automatically in unseen data.

The tools used to find units are:

The *temporal decomposition (TD)* is a representative of algorithms able to detect quasi-stationary parts in the parametric representation of speech. This method, introduced by Atal (Atal, 1983) and refined by Bimbot (Bimbot, 1990), approximates the trajectories of parameters $x_i(n)$ by a sum of m *targets* a_{ik} weighted by *interpolation functions* (IF):

$$\hat{x}_i(n) = \sum_{k=1}^{m} a_{ik}\phi_k(n), \quad \text{for} \quad i = 1, \ldots, P, \qquad (1)$$

where P is the dimension of the parameter vectors. Equation 1 can be written:

$$\underset{(P \times N)}{\hat{\mathbf{X}}} = \underset{(P \times m)}{\mathbf{A}} \; \underset{(m \times N),}{\mathbf{\Phi}} \qquad (2)$$

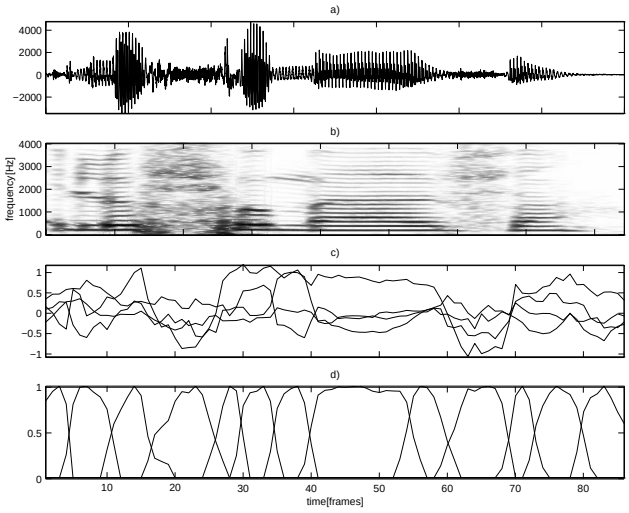


Figure 1: Illustration of temporal decomposition of French word “le chômage”: a) signal. b) spectrogram. c) trajectories of first 4 LPCC parameters. d) TD interpolation functions.

where the lower line indicates matrix dimensions. The initial interpolation functions are found using local Singular Value Decomposition with adaptive windowing, followed by post-processing (smoothing, decorrelation and normalization). Target vectors are then computed by: $\mathbf{A} = \mathbf{X}\mathbf{\Phi}^{\#}$, where $\mathbf{\Phi}^{\#}$ denotes the pseudo-inverse of IFs matrix. IFs and targets are locally refined in iterations minimizing the distance of $\mathbf{X}$ and $\hat{\mathbf{X}}$. Intersections of interpolation functions permit to define speech segments. An example of TD can be seen in Fig. 1. Critically speaking, any of automatic segmentation procedures, based for example on spectral variation function (SVF), could be used. We chose TD because the algorithm and software were readily available in the lab.

Unsupervised clustering assigns segments to classes. Vector quantization (VQ) is used for automatic determination of classes: class centroids are minimizing the overall distortion on the training set. The VQ codebook $\mathbf{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_L\}$ is trained by K-means algorithm with binary splitting. Training is performed using vectors positioned in gravity centers of TD interpolation functions, while the *quantization* takes into account entire segments using cumulated distances between all vectors of a segment and a code-vector. TD with VQ produce a phone-like segmentation of speech.

Hidden Markov models (HMM) can be used to model the units. HMM parameters are *initialized* using context-free and context-dependent Baum-Welch training (Young et al., 1996) with TD+VQ transcriptions, and *refined* in successive steps of corpus segmentation (using HMMs) and model parameters re-estimation. The speech represented by observation vector string can then be aligned with models by standard likelihood maximization. At this point, we obtain the three desired outputs of the unit determination algorithm: units, their models and training data alignments. The units can be used in further processing.

3. Very low bit rate coding

The VLBR coding using ALISP units has been the first verification of our approach. It turned out however, that after some modifications to improve the output speech quality (discussed later in this section), it can have potential applications.

The coder performs the *recognition* of input unseen speech into ALISP units, that we call *coding units*. For the *synthesis* in the decoder, however, another type of units called *synthesis units* can be defined – units can be for example designed in such a way that the synthesis units spans a speech segment between two spectrally stable parts, so that the concatenation becomes easier. Finally, the decoder must dispose of a certain number of *representatives* of each synthesis unit. The coder must send the index of best-matching representative (DTW-distance was used as distortion measure) and information on the prosody: timing and pitch and energy contours.

The *decoder* receives the information on coding units and derives the information on synthesis units, then it retrieves the representative from its memory. The synthesis modifies the prosody of the representative and produces output speech.

3.1. Basic coding tests

This approach was first tested in speaker-dependent experiments on American English (Černocký, 1998), French (Černocký et al., 1998) and Czech (Černocký et al., 1999). The speech parameterization was done by a set of LPC-cepstral coefficients on 20 ms frames with 10 ms frame-shift. Temporal decomposition was set to produce 15–17 targets per second in average (corresponding to average phoneme rate). The VQ codebook had 64 code-vectors that were trained using the original vectors (not TD-targets) located in gravity centers of TD interpolation functions. After initial labeling using the TD+VQ tandem, first “generation” of HMMs (3 emitting states, no state-skip, single-Gaussian) was trained. The training corpus was aligned with those models, and 5 iterations of retraining-alignment were run.

In the coding, synthesis units corresponded to the coding ones, and for each, 8 longest representatives were searched in the training data. The number of bits per unit was therefore $\log_2 64$ (unit) $+ \log_2 8$ (representative) $= 9$. This led to the average bit rate for unit encoding of 100–200 bps. The prosody was not coded in those experiments, and the physical synthesis in the decoder was done by a rudimentary LPC synthesizer.

Intelligible speech was obtained for the three languages – low speech quality was attributed mainly to rudimentary LPC synthesis rather than the units themselves. Those experiments justified our approach – they proved that a “phonetic-like” speech coder can be trained without ever seeing any transcriptions of the speech data.

3.2. Harmonic Noise Model synthesis

In basic structure of the coder, LPC synthesis has been used to produce the output speech. It was found to be highly responsible for the low quality of the resulting speech (that can be proved by a copy LPC analysis-synthesis). Therefore, the Harmonic-Noise Model (HNM) which brings

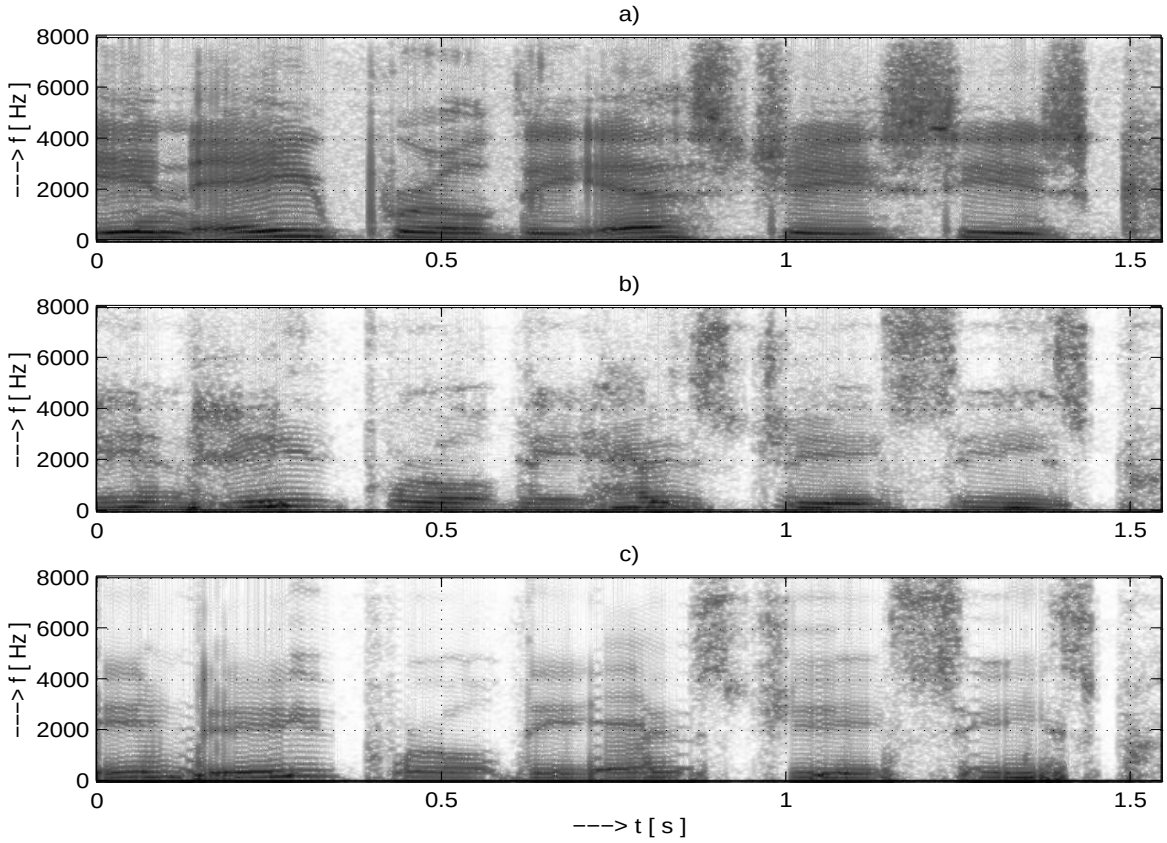


Figure 2: Spectrograms. a) original speech signal. b) coded speech synthesized by HNM. c) coded speech synthesized by LPC.

much higher quality of the synthesized speech, is applied. The principle of HNM is in detail described in (Oudot, 1996; Stylianou, 1996). The HNM is built on following representation of signal $x(n)$:

$$x(n) = \underbrace{\sum_{k=1}^{P} \alpha_k \cos(2\pi f_k n - \phi_k)}_{\text{Harmonics}} + \underbrace{b(n)}_{\text{Noise}}, \qquad (3)$$

where P is the number of harmonics, α_k are the amplitudes, f_k the multiples of pitch and ϕ_k phases of harmonic part. $b(n)$ expresses components of noise.

Eq. 3. describes both parts of HNM. The first part "Harmonics" decomposes the speech signal into a sum of sinusoids. In fact, a combination of harmonically related and non-harmonically related sinusoids can also be used. "Noise" in Eq. 3 represents non-harmonic part of speech signal. The parameters for the noise and harmonic part are estimated separately. The fundamental frequency estimation is isolated from the estimation of amplitudes and phases and the interdependence of the parameters in neighboring frames is alleviated through the hypothesis of the quasi-stationary signal. Thus, the first step of the analysis process consists of estimating the fundamental frequency for the voiced frames. In our work, a classical method based on normalized cross correlation function (NCCF) (Talkin, 1995) has been applied.

The estimation of amplitudes and phases of the harmonics is done using the method of least mean squares (Charbit and Paulsson, 2000). The noise of *voiced frames* is obtained by subtracting the previously computed harmonics from the input signal. Its spectrum is modeled by LPC auto-recursive filter of 12^{th} order. In *unvoiced frames*, only parameters of the noise model are estimated. Auto-recursive filter of 12^{th} order is used, as above. In synthesis, the source signal is represented by white noise filtered by the estimated LPC filter.

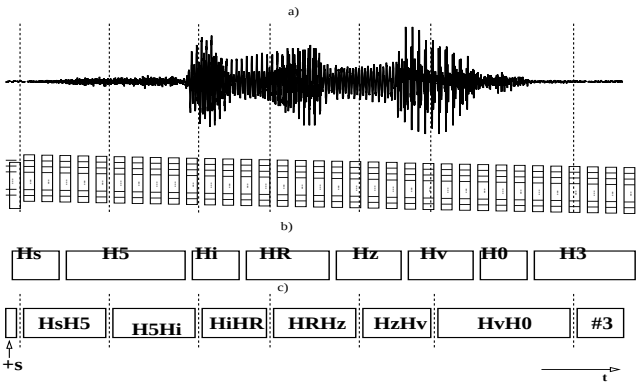


Figure 3: Example of re-segmentation according to middle frames of original units. Minimal length of new units is 4 frames: a) speech signal with its splitting into the frames. b) original segmentation recognized by HMMs. c) new re-segmentation.

Spectral envelope is needed to perform pitch modifica-

tion in the synthesized speech. The log of spectral envelope is computed from the estimated amplitudes of the harmonics using real-cepstrum coefficients (Charbit and Paulsson, 2000).

The results (Motlíček et al., 2001) have demonstrated, that the replacement LPC synthesis by HNM version is highly responsible for great improvement of quality of resulting speech, as can be seen in Fig. 2, where spectrograms from the same part of speech signal are compared[2].

3.3. Synthesis units

The units initialized by the temporal decomposition are inherently unstable at their boundaries (remember, that the center of TD-units tends to be stable). Such units are therefore not very suitable for synthesis as they do not have good concatenation properties. We have therefore tested two approaches to make *synthesis units* units closer to diphone-based or corpus-based speech synthesis.

First, *selection of longer synthesis units* based on the original coding ones was tested (Motlíček et al., 2001). This approach is illustrated in Fig. 3. These long units can be constructed by aggregation of short ALISP coding units with re-segmentation in spectrally stable parts of the extremity units. The synthesizer is similar to a diphone one. The results were however not satisfactory, some concatenation noise was still audible and due to the limitation of the training corpus, some synthesis units were missing and difficult to replace.

Therefore, a different method called *short synthesis units with dynamic selection* was developed (Baudoin et al., 2002). Here, for each ALISP class, a large number of representatives is extracted from the training corpus. These synthesis representatives are determined in order to fulfill criteria of good representation of a given segment to be coded and criteria of good concatenation of successive segments.

For each coding unit H_j, we define sub-classes called H_iH_j containing all the speech segments of class H_j that were preceded by a segment belonging to the class H_i in the training corpus. It is possible to keep as synthesis representatives all the segments of the training corpus organized in classes and sub-classes as described above or to limit the size of each sub-class to some maximal value K.

During coding, if a segment is recognized as belonging to class H_j and is preceded by a segment in class H_i, the representative is searched in the subclass H_iH_j of class H_j. The selection of the best representative in the sub-class is done on the distance D_C of good representation of the segment. The D_C distance is based on a spectral comparison by DTW between the segment to code and the potential synthesis representatives. The distance D_C can also include a distance on prosody parameters.

We have verified that this approach provides superior speech quality than the "short" coding units or re-segmented longer ones.

3.4. Toward speaker independent ALISP coder

First results of speaker-independent (SI) coding on large French database BREF have been reported in (Baudoin et al., 2002). Coding units were trained on 33 male speakers from this corpus, and the corresponding representatives were selected from all available speakers in similar fashion to speaker-dependent coding presented in section 3.1. The resulting speech was intelligible, though with lower quality than the speaker-dependent counterpart. This confirmed the possibility to use the ALISP scheme also in SI environment.

Two problems are crucial for the SI operation: speaker clustering or speaker adaptation in the coder and voice modification in the decoder. For the first problem, the TSD article (Baudoin et al., 2002) presents speaker-independent coding with VQ-based speaker clustering. Here, the reference speakers are *pre-clustered*, in order to select the closest speaker or the closest subset of speakers for HMM refinements and/or adaptation of synthesis units. A VQ-based inter-speaker distance using the unsupervised hierarchical VQ algorithm was used (Furui, 1989). The basic assumption is that training speech material from the processed speaker is available during a short training phase for running the VQ adaptation process. The inter-speaker distance is defined as the cumulated distance between centroids of the non-aligned code-books, using the correspondence resulting from the aligned code-books obtained through the adaptation process. This distance is used in the off-line pre-training phase for clustering the reference speakers, and during the on-line training phase for selecting the closest cluster to the user. From the distance matrix, subclasses are extracted using a simplified split-based clustering method.

The proposed concept has been validated on the BREF corpus (phonetically balanced sentences), 16 LPCC coefficients and 64 classes were used. Illustration of the clustering process is given for the largest class, (left panel of Fig. 4), a typical class (middle panel) and an isolated speaker (right panel) in terms of relative distance to the other speakers. One could note the similar positioning of speakers belonging to the same cluster.

The obtained results in terms of speaker clustering using a small amount of data are encouraging. In our future works, we will study a speaker-independent VLBR structure derived from this concept, by adding HMM adaptation at the encoder, and voice conversion techniques at the decoder.

4. ALISP units in recognition

4.1. ALISP–phonetic correspondence

To investigate the potential usability of ALISP units in speech recognition, we performed several experiments on the comparisons of ALISP and phonetic alignments (Černocký et al., 2001).

Such alignments were available with the Boston University corpus of American English (a database that we used for the initial VLBR coding experiments). They were obtained at BU using a segmental HMM recognizer constrained by possible pronunciations of utterances (Ostendorf et al., 1995). The measure of correspondence was the relative overlap r of ALISP unit with a phoneme (see Fig. 5 for illustration). The results are summarized in *confusion*

[2] http://www.fee.vutbr.cz/∼motlicek/speech_hnm.html contains examples of speech after coding/decoding.

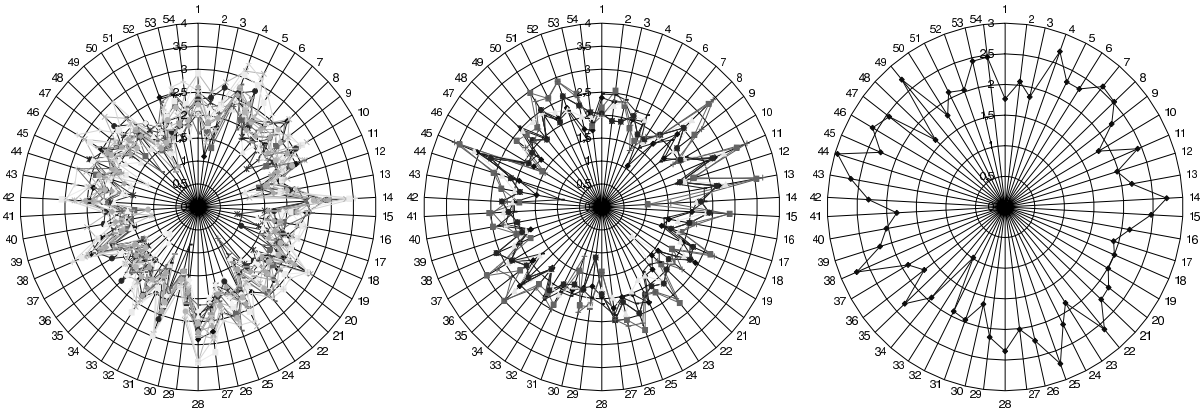


Figure 4: Left panel: Relative distance of speakers from the largest cluster. Middle panel: Relative distance of speakers from a typical cluster (indexes 6, 14, 21, 29, 31, 43). Right panel: Relative distance of speakers from an isolated speaker (index 33).

matrix $\mathbf{X}$ ($n_p \times n_a$), whose elements are defined:

$$x_{i,j} = \frac{\sum_{k=1}^{c(p_i)} r(p_{i_k}, a_j)}{c(p_i)}. \quad (4)$$

n_p and n_a are respectively the sizes of phoneme and ALISP unit dictionaries, p_i is the i-th phoneme, a_j is the j-th ALISP unit, $c(p_i)$ is the count of p_i in the corpus and $r(p_{i_k}, a_j)$ is the relative overlapping of k-th occurrence of p_i with ALISP unit a_j. The columns of $\mathbf{X}$ are rearranged to let the matrix have a quasi-diagonal form. As for the phoneme set, on contrary to BU alignments, where stressed vowels are differentiated from unstressed ones, we used the original TIMIT set. The ALISP set had 64 units. The resulting matrix is given in Fig. 6.

This matrix shows, that the correspondence between ALISP units and phonemes is consistent, but not unique. We can for example see, that the ALISP unit `a` corresponds to closures, but also to the pause. The unit `$` has a strong correlation with `SH` but it is also linked to its voices counterpart `ZH` and to affricates `JH` et `CH`, which are acoustically very closed.

4.2. Using ALISP units in recognition

Although the above mentioned experiments showed a correlation of phonemes and ALISP units, an ALISP recognition system should probably not be based on direct phoneme–ALISP mapping. Stochastic mapping of *sequences* of phonemes to *sequences* of ALISP units would be one solution. This approach was studied in (Deligne, 1996): likelihood maximization is applied to joint segmentation of two streams of observations, where the first can be the phonetic labeling and the second the sequence of automatically derived units. When the testing data are processed, the method finds the segmentation together with optimal "transcription" into sequences of phonemes. The observations can be either symbolic (in this case, the method is "discrete") or vectorial (here, not only statistics of sequences, but also emission PDFs come into mind).

Another option is the *composition* of ALISP units into word and phone models, proposed in (Fukada et al., 1996). Here, the basic units are first derived in an unsupervised manner. Then, phonetic transcription is compared to ALISP segmentation and composite models are constructed for the *words* of the corpus. In case the data do not contain sufficient number of examples of a word, the method can "back-up" to *phoneme models* composed in similar manner as the word ones.

Third solution was proposed for triphone models, but it would generalize well also with ALISP units. This approach does not require phonetic transcriptions but a large database with word boundaries. ALISP labels are generated for this DB and the ALISP-pronunciation dictionary is created. It is however necessary to develop an expert system for the transcription of unseen words in terms of ALISP units.

5. Conclusions

The algorithm of unit search produces set of consistent units but is far from optimal. As for the feature extraction, we have for example not investigated the perceptually motivated features used by Hermansky and his group (Hermansky, 1997). The distance used in VQ could be replaced by the Kullback-Leibler one, that has shown superior performances in selection of units for synthesis (Stylianou and Syrdal, 2001). The training of unit models could be done completely without initialization of time boundaries (currently temporal decomposition) and of labels (VQ) by using an Ergodic Hidden Markov model (EHMM) for both tasks simultaneously. Finally, it is necessary to think about "shaping" the units for the target application.

The first part of the paper demonstrates that *speech coding*, at transmission rate lower than 400 bps, can be achieved using automatically derived units. The drawback of our proposal is the size of the memory required both in coder and decoder and the delay introduced by the maximal duration of the segments (several hundreds msec). There are many applications which could tolerate both a large memory (let say 200 Mbytes) and the delay. Among such applications are the multimedia mobile terminal of the future (including the electronic book), the secured mobile phone, the compression of conferences (including distance education), etc. More work is necessary on voice transformation so that only typical voices will be kept in memory.

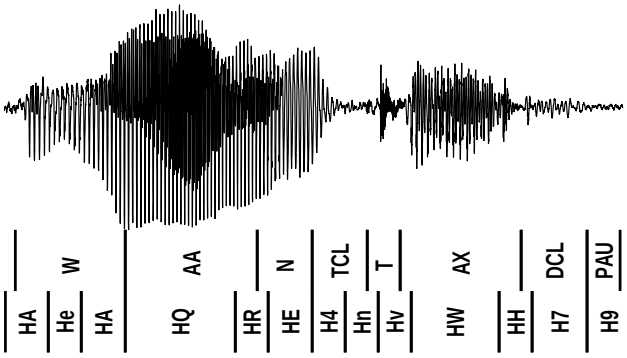


Figure 5: Illustration of comparison of ALISP and phonetic segmentations: word "wanted" from female speaker of Boston University corpus.

Characterization of a voice based on limited data and use of this characterization to transform another voice is an interesting research topic.

As for the *recognition*, we can conclude that building of ALISP-based recognizer will not be a straightforward task. The invested efforts should however be generously recompensed by the limitation of human efforts needed to create or modify such a recognizer. If we obtain an efficient scheme, and in the same time we succeed in limiting the human labor (annotations, pronunciation dictionaries, etc.), it will be a great step toward the *real automating* of speech processing, and it will also open the way to its easier implementation in different languages.

ALISP unit use should not be limited to coding or recognition. In (Petrovska-Delacrétaz et al., 2000), we have reported results of a speaker-verification system with pre-segmentation in ALISP units before actual scoring. The performance of our system on 1999 NIST data was not optimal, but we believe that pre-segmentation of speech into classes and determination of their speaker-characterization performances can aid the verification system. Results obtained from class-specific models can be then combined using appropriate weighting factors before taking the decision.

The last proposed application domain is the *language identification*. Most current language identification (LI) systems are based on the approach of extracting the phonotactic language specific information. The phonotactics is related to the modeling of the statistical dependencies inherent in the phonetic chains. Unfortunately, transcribed databases should be available to train the required phonetic recognizer, and the transcription step is a major bottleneck for the adaptation of systems to new languages or services (as it is for the other domains). We propose to replace the widely used phonetic-based recognizers by an ALISP-based recognizer, and to extract from the automatically segmented speech units the necessary information for solving the problem of language identification. The advantage of the proposed method is its portability to new languages, for which we do not have annotated databases.

6. Acknowledgments

Several colleagues have contributed to this article. Fréderic Bimbot (IRISA Rennes, France) contributed the `td95` package used for the temporal decomposition. Geneviève Baudoin (ESIEE Paris) supervised initial stages of the work and with Fadi El Chami (Université Libanaise Tripoli and ESIEE), she is the author of dynamic unit selection for the synthesis. Petr Motlíček (VUT Brno) worked on the creation of long synthesis units and HNM synthesis. François Capman (Thales Communications) is the author of VQ-based speaker clustering. Dijana Petrovska-Delacrétaz (University of Fribourg, Switzerland) did the speaker-independent experiments and is working on ALISP units deployment in speaker verification. Maurice Charbit and his students (ENST Paris) contributed to the HNM synthesis. Vladimír Šebesta and Richard Menšík (VUT Brno) helped with the visualization of confusion matrices. Finally, thanks to Gérard Chollet (ESNT Paris) for having been at the original ALISP idea and for many constructive comments and supervision.

This research has been supported by Grant Agency of Czech Republic under project No. 102/02/0124 and by the RNRT (Réseau National de Recherche en Télécommunications) project SYMPATEX.

7. References

B. S. Atal. 1983. Efficient coding of LPC parameters by temporal decomposition. In *Proc. IEEE ICASSP 83*, pages 81–84.

G. Baudoin, F. Capman, J. Černocký, F. El Chami, M. Charbit, and Gérard Chollet. 2002. Advances in very low bit rate speech coding using recognition and synthesis techniques. In *submitted to TSD 2002*, Brno, Czech Republic, September.

F. Bimbot. 1990. An evaluation of temporal decomposition. Technical report, Acoustic research department AT&T Bell Labs.

M. Charbit and N. Paulsson. 2000. Boîte à outils harmoniques plus bruit. Technical report, ENST Paris, France, November.

G. Chollet, J. Černocký, A. Constantinescu, S. Deligne, and F. Bimbot. 1999. Towards ALISP: a proposal for Automatic Language Independent Speech Processing. In K. Ponting, editor, *Computational models of speech pattern processing*, NATO ASI Series, pages 375–388. Springer Verlag.

S. Deligne. 1996. *Modèles de séquences de longueurs variables: Application au traitement du langage écrit et de la parole*. Ph.D. thesis, École nationale supérieure des télécommunications (ENST), Paris.

T. Fukada, M. Bacchiani, K. Paliwal, and Y. Sagisaka. 1996. Speech recognition based on acoustically derived segment units. In *Proc. ICSLP 96*, pages 1077–1080.

S. Furui. 1989. Unsupervised speaker adaptation method based on hierarchical spectral clustering. In *Proc. ICASSP'89*, pages 286–289.

H. Hermansky. 1997. Should recognizers have ears? In *Proc. Tutorial and Research Workshop on Robust speech recognition for unknown communication channels*, pages 1–10, Pont-a-Mousson, France, April. ESCA-NATO.

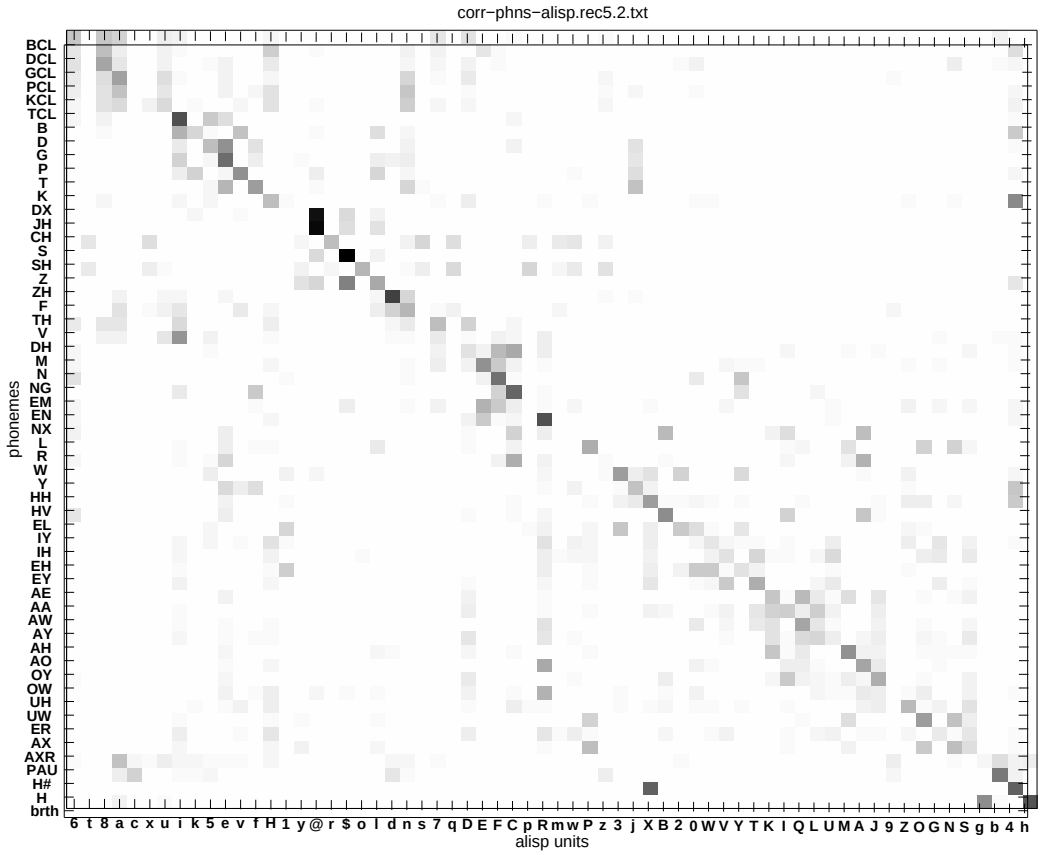


Figure 6: Correspondence of ALISP segmentation and phonetic alignment for speaker F2B in BU corpus. White color corresponds to zero correlation, black to maximum value $x_{i,j}$=0.806

P. Motlíček, J. Černocký, G. Baudoin, and G. Chollet. 2001. Minimization of transition noise and HNM synthesis in very low bit rate speech coding. In V. Matoušek, P. Mautner, P. Mouček, and K. Taušer, editors, *Proc. of 4th International Conference Text, Speech,Dialogue - TSD 2001*, number 2166 in Lecture notes in artificial intelligence, pages 305–312, Železná Ruda, Czech Republic, September. Springer Verlag.

M. Ostendorf, P.J. Price, and S. Shattuck-Hufnagel. 1995. The Boston University radio news corpus. Technical report, Boston University, February.

M. C. Oudot. 1996. *Etude du modèle "Sinusoides et bruit" pour le traitement des signaux de parole, estimation robuste de l'enveloppe spectrale*. Ph.D. thesis, École nationale supérieure des télécommunications (ENST), Paris.

D. Petrovska-Delacrétaz, J. Černocký, J. Hennebert, and G. Chollet. 2000. Segmental approaches for automatic speaker verification. *Digital Signal Processing*, 10:1–3, January/April/July. Special Issue: NIST 1999 Speaker Recognition Workshop.

Y. Stylianou and A.K. Syrdal. 2001. Perceptual and objective detection of discontinuities in concatenative speech synthesis. In *Proc. ICASSP'01*, volume 2, pages 837–840, Salt Lake City, Utah, USA.

I. Stylianou. 1996. *Modèles harmoniques plus bruit combinés avec des méthodes statistiques, pour la modification de la parole et du locuteur*. Ph.D. thesis, École nationale supérieure des télécommunications (ENST), Paris, January.

D. Talkin. 1995. A robust algorithm for pitch tracking (rapt). In W. B. Kleijn and K. Paliwal, editors, *Speech Coding and Synthesis*, New York. Elseviever.

J. Černocký, G. Baudoin, and G. Chollet. 1998. Segmental vocoder - going beyond the phonetic approach. In *Proc. IEEE ICASSP 98*, pages 605–608, Seattle, WA, May. http://www.fee.vutbr.cz/~cernocky/Icassp98.html.

J. Černocký, I. Kopeček, G. Baudoin, and G. Chollet. 1999. Very low bit rate speech coding: comparison of data-driven units with syllable segments. In V. Matoušek, P. Mautner, J. Ocelíková, and P. Sojka, editors, *Proc. of Workshop on Text Speech and Dialogue (TSD'99)*, number 1692 in Lecture notes in computer science, pages 262–267, Mariánské Lázně, Czech Republic, September. Springer Verlag.

J. Černocký, G. Baudoin, D. Petrovska-Delacrétaz, and G. Chollet. 2001. Vers une analyse acoustico-phonétique de la parole indépendante de la langue, basée sur ALISP. *Revue Parole*, 2001(17,18,19):191–226.

J. Černocký. 1998. *Speech Processing Using Automatically Derived Segmental Units: Applications to Very Low Rate Coding and Speaker Verification*. Ph.D. thesis, Université Paris XI Orsay, December.

S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland. 1996. *The HTK book*. Entropics Cambridge Research Lab., Cambridge, UK.

Some Issues in Speech Recognizer Portability

Lori Lamel

Spoken Language Processing Group,
LIMSI-CNRS, France
lamel@limsi.fr

Abstract

Speech recognition technology has greatly evolved over the last decade. However, one of the remaining challenges is reducing the development cost. Most recognition systems are tuned to a particular task and porting the system to a new task (or language) requires substantial investment of time and money, as well as human expertise. Todays state-of-the-art systems rely on the availability of large amounts of manually transcribed data for acoustic model training and large normalized text corpora for language model training. Obtaining such data is both time-consuming and expensive, requiring trained human annotators with substantial amounts of supervision. This paper addresses some of the main issues in porting a recognizer to another task or language, and highlights some some recent research activities aimed a reducing the porting cost and at developing generic core speech recognition technology.

1. Introduction

Speech recognition tasks can be categorized by several dimensions: the number of speakers known to the system, the vocabulary size, the speaking style, and the acoustic conditions. Concerning speakers, the most restrictive is when only one speaker can use the system and the speaker is required to enroll with the system in order to be recognized (speaker-dependent). The system may be able to recognize speech from several speakers, but still requires enrollment data (multiple speaker) or the system can recognize the speech from nominally any speaker without any training data (speaker-independent).

A decade ago the most common recognition tasks were either small vocabulary isolated word or phrases or speaker dependent dictation, whereas today speech recognizers are able to transcribe unrestricted continuous speech from broadcast data in multiple languages with acceptable performance. The increased capabilities of todays recognizers is in part due to the improved accuracy (and increased complexity) of the models, which are closely related to the availability of large spoken and text corpora for training, and the wide availability of faster and cheaper computational means which have enabled the development and implementation of better training and decoding algorithms. Despite the extent of progress over the recent years, recognition accuracy is still quite sensitive to the environmental conditions and speaking style: channel quality, speaker characteristics, and background noise have a large impact on the acoustic component of the speech recognizer, whereas the speaking style and discourse domain largely influence the linguistic component. In addition, most systems are both task and language dependent, and bringing up a system for a different task or language is costly and requires human expertise.

Only for small vocabulary, speaker-dependent isolated word or phrase speech recognizers, such as name dialing on mobile telephones, portability is not really an issue. With such devices, all of the names must be entered by the user according to the specific protocol - such systems typically use whole word patterns and do not care who the speaker or what the language is. For almost all more complex tasks, portability is a major concern. Some speech technology companies have been addressing the language localization problem for many years, and some research sites have also been investigating speech recognition in multiple languages (4; 13; 14; 21; 35; 37) as well as speech recognition using multi-lingual components (19; 33). Multi-lingual speech processing has been the subject of several special sessions at conferences and workshops (see for example, (1; 2; 3; 20)). The EC CORETEX project (http://coretex.itc.it) is investigating methods to improve basic speech recognition technology, including fast system development, as well as the development of systems with high genericity and adaptability. Fast system development refers to both language support, i.e., the capability of porting technology to different languages at a reasonable cost; and task portability, i.e. the capability to easily adapt a technology to a new task by exploiting limited amounts of domain-specific knowledge. Genericity and adaptability refer to the capacity of the technology to work properly on a wide range of tasks and to dynamically keep models up to date using contemporary data. The more robust the initial generic system is, the less there is a need for adaptation.

In the next section an overview of todays most widely used speech recognition technology is given. Following subsections address several approaches to reducing the cost of porting, such as improving model genericity, and reducing the need for annotated training data. An attempt is made to give an idea of the amount of data and effort required to port to a different language or task.

2. Speech Recognition Overview

Speech recognition is concerned with converting the speech waveform into a sequence of words. Today's most performant approaches are based on a statistical modelization of the speech signal (16; 31; 32; 38). The basic modeling techniques have been successfully applied to a number of languages and for a wide range of applications.

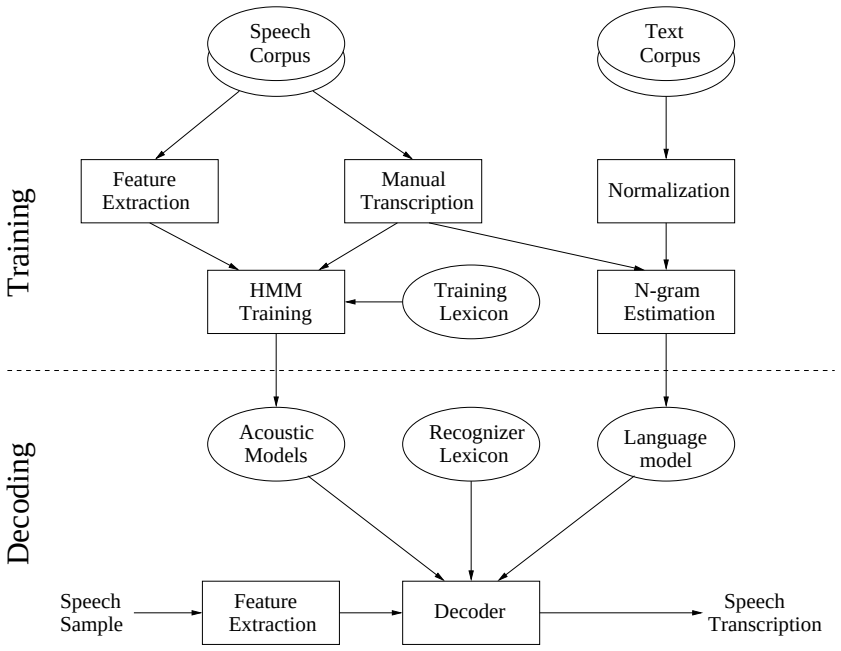


Figure 1: System diagram of a generic speech recognizer based using statistical models, including training and decoding processes.

The main components of a speech recognition system are shown in Figure 1. The elements shown are the main knowledge sources (speech and textual training materials and the pronunciation lexicon), the feature analysis (or parameterization), the acoustic and language models which are estimated in a training phase, and the decoder. The training and decoding algorithms are largely task and language independent, the main language dependencies are in the knowledge sources (the training corpora).

The first step of the acoustic feature analysis is digitization, in which the continuous speech signal is converted into discrete samples. Acoustic feature extraction is then carried out on a windowed portion of speech [1], with the goal of reducing model complexity while trying to maintain the linguistic information relevant for speech recognition. Most recognition systems use short-time cepstral features based either on a Fourier transform or a linear prediction model. Cepstral parameters are popular because they are a compact representation, and are less correlated than direct spectral components. Cepstral mean removal (subtraction of the mean from all input frames) is commonly used to reduce the dependency on the acoustic recording conditions, and delta parameters (obtained by taking the first and second differences of the parameters in successive frames) are often used to capture the dynamic nature of the speech signal. While the details of the feature analysis differs from system to system, most of the commonly used analyses can be expected to work reasonably well for most languages and tasks.

Most state-of-the-art systems make use of hidden Markov models (HMM) for acoustic modeling, which consists of modeling the probability density function of a sequence of acoustic feature vectors (32). These models are popular as they are performant and their parameters can be efficiently estimated using well established techniques. The Markov model is described by the number of states and the transitions probabilities between states. The most widely used acoustic units in continuous speech recognition systems are phone-based[2], and typically have a small number of left-to-right states in order to capture the spectral change across time. Since the number of states imposes a minimal time duration for the unit, some configurations allow certain states to be skipped. The probability of an observation (i.e. a speech vector) is assumed to be dependent only on the state, which is known as the 1st order Markov assumption.

Phone based models offer the advantage that recognition lexicons can be described using the elementary units of the given language, and thus benefit from many linguistic studies. It is of course possible to perform speech recognition without using a phonemic lexicon, either by use of "word models" (a commonly used approach for isolated word recognition) or a different mapping such as the fenones (7). Compared with larger units, small subword units reduce the number of parameters, and more importantly can be associated with back-off mechanisms to model rare or unseen, contexts, and facilitate porting to new vocabularies. Fenones offer the additional advantage of automatic training which is of interest for language porting, but lack the ability to include *a priori* linguistic models.

A given HMM can represent a phone without consideration of its neighbors (context-independent or mono-

[1] An inherent assumption is that due to physical constraints on the rate at which the articulators can move, the signal can be considered quasi-stationary for short periods (on the order of 10ms to 20ms).

[2] Phones usually correspond to phonemes, but may also correspond to allophones such as flaps or glottal stop.

phone model) or a phone in a particular context (context-dependent model). The context may or may not include the position of the phone within the word (word-position dependent), and word-internal and cross-word contexts may or may not be merged. Different approaches can be used to select the contextual units based on frequency or using clustering techniques, or decision trees, and different types of contexts have been investigated. The model states are often clustered so as to reduce the model size, resulting in what are referred to as "tied-state" models.

Acoustic model training consists of estimating the parameters of each HMM. For continuous density Gaussian mixture HMMs, this requires estimating the means and covariance matrices, the mixture weights and the transition probabilities. The most popular approaches make use of the Maximum Likelihood criterion, ensuring the best match between the model and the training data (assuming that the size of the training data is sufficient to provide robust estimates). Since the goal of training is to find the best model to account of the observed data, the performance of the recognizer is critically dependent upon the representativity of the training data. Speaker-independence is obtained by estimating the parameters of the acoustic models on large speech corpora containing data from a large speaker population. Since there are substantial differences in speech from male and female talkers arising from anatomical differences it is thus common practice to use separate models for male and female speech in order to improve recognition performance (requiring automatic gender identification).

2.1. Lexical and pronunciation modeling

The lexicon is the link between the acoustic-level representation and the word sequence output by the speech recognizer (34). Lexical design entails two main parts: definition and selection of the vocabulary items and representation of each pronunciation entry using the basic acoustic units of the recognizer. Recognition performance is obviously related to lexical coverage, and the accuracy of the acoustic models is linked to the consistency of the pronunciations associated with each lexical entry. Developing a consistent pronunciation lexicon requires substantial language specific knowledge from a native speaker of the language and usually entails manual modification even if grapheme-to-phoneme rules are reasonably good for the language of interest. The lexical units must be able to be automatically extracted from a text corpus or from speech transcriptions and for a given size lexicon should optimize the lexical coverage for the language and the application. Since on average, each out-of-vocabulary (OOV) word causes more than a single error (usually between 1.5 and 2 errors), it is important to judiciously select the recognition vocabulary. The recognition word list is to some extent dependent on the conventions used in the source text (punctuation markers, compound words, acronyms, case sensitivity, ...) and the specific language. The lexical units can be chosen to explicitly model observed pronunciation variants, for example, using compound words to represent word sequences subject to severe reductions such as "dunno" for "don't know". The vocabulary is usually comprised of a simple list of lexical items as observed in the text. Attempts have been made to use other units, for example, to use a list of root forms (stems) augmented by derivation, declension, composition rules. However, while more powerful in terms of language coverage, such representations are more difficult to integrate in present state-of-the-art recognizer technology.

These pronunciations may be taken from existing pronunciation dictionaries, created manually or generated by an automatic grapheme-phoneme conversion software. Alternate pronunciations are sometimes used to explicitly represent variants that cannot be easily modeled by the acoustic units, as is the case for homographs (words spelled the same, but pronounced differently) which reflect different parts of speech (verb or noun) such as *excuse*, *record*, *produce*. While pronunciation modeling is widely acknowledged to be a challenge to the research community, there is a lack of agreement as to what pronunciation variants should be modeled and how to do so. Adding a large number of pronunciation variants to a recognition lexicon without accounting for their frequency of occurrence can reduce the system performance. An automatic alignment system is able to serve as an analysis tool which can be used to quantify the occurrence of events in large speech corpora and to investigate their dependence on lexical frequency (5).

2.2. Language modeling

Language models (LMs) are used in speech recognition to estimate the probability of word sequences. Grammatical constraints can be described using a context-free grammars (for small to medium size vocabulary tasks these are usually manually elaborated) or can be modeled stochastically, as is common for LVCSR. The most popular statistical methods are *n*-gram models, which attempt to capture the syntactic and semantic constraints by estimating the frequencies of sequences of *n* words. The assumption is made that the probability of a given word string $(w_1, w_2, ..., w_k)$ can be approximated by $\prod_{i=1}^{k} \Pr(w_i | w_{i-n+1}, ..., w_{i-2}, w_{i-1})$, therefore reducing the word history to the preceding $n-1$ words. A back-off mechanism is generally used to smooth the estimates of the probabilities of rare *n*-grams by relying on a lower order *n*-gram when there is insufficient training data, and to provide a means of modeling unobserved word sequences (17).

Given a large text corpus it may seem relatively straightforward to construct *n*-gram language models. Most of the steps are pretty standard and make use of tools that count word and word sequence occurrences. The main differences arise in the choice of the vocabulary and in the definition of words, such as the treatment of compound words or acronyms, and the choice of the back-off strategy. There is, however, a significant amount of effort needed to process the texts before they can be used.

One of the main motivations for text normalization is to reduce lexical variability so as to increase the coverage for a fixed vocabulary size. The normalization decisions are generally language-specific. Much of speech recognition research for American English has been supported by ARPA and has been based on text materials which were

processed to remove upper/lower case distinction and compounds. Thus, for instance, no lexical distinction is made between *Gates, gates* or *Green, green*. However with increased interest in going beyond transcription to information extraction tasks (such as finding named entities or locating events in the audio signal) such distinctions are important. In our work at LIMSI for other languages (French, German, Portuguese) capitalization of proper names is distinctive with different lexical items for the French words *Pierre, pierre* or *Roman, roman*.

The main conditioning steps are text mark-up and conversion. Text mark-up consists of tagging the texts (article, paragraph and sentence markers) and garbage bracketing (which includes not only corrupted text materials, but all text material unsuitable for sentence-based language modeling, such as tables and lists). Numerical expressions are typically expanded to approximate the spoken form ($150 $\rightarrow$ one hundred and fifty dollars). Further semi-automatic processing is necessary to correct frequent errors inherent in the texts (such as obvious mispellings *milllion, officals*) or arising from processing with the distributed text processing tools. Some normalizations can be considered as "decompounding" rules in they modify the word boundaries and the total number of words. These concern the processing of ambiguous punctuation markers (such as hyphen and apostrophe), the processing of digit strings, and treatment of abbreviations and acronyms (ABCD $\rightarrow$ A. B. C. D.). Another example is the treatment of numbers in German, where decompounding can be used in order to increase lexical coverage. The date 1991 which in standard German is written as *neunzehnhunderteinundneunzig* can be represented by word sequence *neunzehn hundert ein und neunzig*. Generally speaking, the choice is a compromise between producing an output close to correct standard written form of the language and lexical coverage, with the final choice of normalization being largely application-driven.

In practice, the selection of words is done so as to minimize the system's OOV rate by including the most useful words. By useful we mean that the words are expected as an input to the recognizer, but also that the LM can be trained given the available text corpora. There is the sometimes conflicting need for sufficient amounts of text data to estimate LM parameters and assuring that the data is representative of the task. It is also common that different types of LM training material are available in differing quantities. One easy way to combine training material from different sources is to train a language model per source and to interpolate them, where the interpolation weights are estimated on some development data.

2.3. Decoding

The aim of the decoder is to determine the word sequence with the highest likelihood given the lexicon and the acoustic and language models. Since it is often prohibitive to exhaustively search for the best solution, techniques have been developed to reduce the computational load by limiting the search to a small part of the search space. The most commonly used approach for small and medium vocabulary sizes is the one-pass frame-synchronous Viterbi beam search which uses a dynamic programming algorithm. This basic strategy has been extended to deal with large vocabularies by adding features such as dynamic decoding, multipass search and N-best rescoring. Multi-pass decoding strategies progressively add knowledge sources in the decoding process and allows the complexity of the individual decoding passes to be reduced. Information between passes is usually transmitted via word graphs, although some systems use N-best hypotheses (a list of the most likely word sequences with their respectives scores). One important advantage of multi-pass is the possibility to adapt the models between decoding passes. Acoustic model adaptation can be used to compensate mismatches between the training and testing conditions, such as due to differences in acoustic environment, to microphones and transmission channels, or to particular speaker characteristics. Attempts at language model adaptation have been less successful. However, multi-pass approaches are not well suited to real-time applications since no hypothesis can be returned until the entire utterance has been processed.

3. Language porting

Porting a recognizer to another language necessitates modification of some of the system parameters, i.e. those incorporating language-dependent knowledge sources such as the phone set, the recognition lexicon (alternate word pronunciations), and phonological rules and the language model. Different languages have different sets of units and different coarticulation influences amomg adjacent phonemes. This influences the way of choosing context-dependent models and of tying distributions. Other considerations are the acoustic confusability of the words in the language (such as homophone, monophone, and compound word rates) and the word coverage of a given size recognition vocabulary.

One important aspect in developing a transcription system for a different language is obtaining the necessary resources for training the acoustic and language models, and a pronunciation lexicon. The Linguistic Data Consortium (LDC http://www.ldc.upenn.edu) and the European Language Resources Association (ELRA http://www.elda.fr) have greatly aided the creation and distribution of language resources. The number and diversity of language resources has grown substantially over recent years. However, most of the resources are only available for the most interesting languages from the commercial or military perspectives.

There are two predominant approaches taken to bootstrapping the acoustic models for another language. The first is to use acoustic models from an existing recognizer and a pronunciation dictionary to segment manually annotated training data for the target language. If recognizers for several languages are available, the seed models can be selected by taking the closest model in one of the available language-specific sets. An alternative approach is to use a set of global acoustic models, that cover a wide number of phonemes (33). This approach offers the advantage of being able to use the multilingual acoustic models to provide additional training data, which is particularly interesting when only very limited amounts of data ($<$ 10 hours)

for the target language are available.

A general rule of thumb for the necessary resources for speaker independent, large vocabulary continuous speech recognizers is that the minimal data requirements are on the order of 10 hours transcribed audio data for training the acoustic models and several million words of texts (transcriptions of audio if available) for language modeling. Depending upon the application, these resources are more or less difficult to obtain. For example, unannotated data for broadcast news type tasks can be easily recorded via standard TV, satellite or cable and data of this type is becoming more easily accessible via the Internet. Related text materials are also available from a variety of on-line newspapers and new feeds. The manual effort required to transcribe broadcast news data is roughly 20-40 hours per hour of audio data, depending upon the desired precision (8).

Data for other applications can be much more difficult to obtain. In general, for spoken language dialog systems, training data needs to be obtained from users interacting with the system. Often times an initial corpus is recorded from a human-human service (should it exist) or using simulations (Wizard-of-OZ) or an initial prototype system. The different means offer different advantages. For example, WOz simulations help in making design decisions before the technology is implemented and allow alternative designs to be simulated quickly. However, the amount of data that can be collected with a WOz setup is limited by the need for a human wizard. Prototype systems offer the possibility of collection much larger corpora, albeit somewhat limited by the capacity of the current system. We have observed that the system's response generation has a large influence on the naturalness of the data collected with a prototype system.

Other application areas of growing interest are the transcription of conversational speech from telephone conversations and meetings, as well as voicemail. Several sources of multilingual corpora are available (for example, the CallHome and CallFriend corpora from LDC). This data is quite difficult to obtain and costly to annotate due to its very spontaneous nature (hesitations, interruptions, use of jargon). The manual effort involved is higher than that required for broadcast news transcription, and the transcriptions are less consistent and accurate.

The application-specific data is useful for accurate modeling at different levels (acoustic, lexical, syntactic and semantic). Acquiring sufficient amounts of text training data is more challenging than obtaining acoustic data. With 10k queries relatively robust acoustic models can be trained, but these queries contain only on the order of 100k words, which probably yield an incomplete coverage of the task (ie. they are not sufficient for word list development) and are insufficient for training *n*-gram language models.

At LIMSI broadcast news transcription systems have been developed for the American English, French, German, Mandarin, Spanish, Arabic and Portuguese languages. The Mandarin language was chosen because it is quite different from the other languages (tone and syllable-based), and Mandarin resources are available via the LDC as well as reference performance results from DARPA benchmark tests. To give an idea of the resources used in developing these systems, the training material are shown in Table 1. It can be seen that there is a wide disparity in the available language resources for a broadcast news transcription task: for American English, 200 hours of manually transcribed acoustic training were available from the LDC, compared with only about 20-50 hours for the other languages. Obtaining appropriate language model training data is even more difficult. While newspaper and newswire texts are becoming widely available in many languages, these texts are quite different than transcriptions of spoken language. Over 10k hours of commercial transcripts are available for American English (from PSMedia), and many TV stations provide closed captions. Such data are not available for most other languages, and in some countries it is illegal to sell transcripts. Not shown here, manually annotated broadcast news corpora are also available for the Italian (30 hours) and Czech (30 hours) languages via ELRA and LDC respectively, and some text sources can be found on the Internet.

Some of the system characteristics are shown in Table 2, along with indicative recognition performance rates for these languages. State-of-the-art systems can transcribe unrestricted American English broadcast news data with word error rates under 20%. Our transcription systems for French and German have comparable error rates for news broadcasts (6). The character error rate for Mandarin is also about 20% (10). Based on our experience, it appears that with appropriately trained models, recognizer performance is more dependent upon the type and source of data, than on the language. For example, documentaries are particularly challenging to transcribe, as the audio quality is often not very high, and there is a large proportion of voice over.

4. Reducing the porting cost

4.1. Improving Genericity

In the context of the EC CORETEX project, research is underway to improve the genercity of speech recognition technology, by improving the basic technolgoy and exploring rapid adaptation methods which start with the initial robust generic system and enhance performance on particular tasks. To this extent, cross task recognition experiments have been reported where models from one task are used as a starting point for other tasks (24; 9; 15; 26; 30; 11). In (26) broadcast news (BN) (28) acoustic and language models to decode the test data for three other tasks (TI-digits (27), ATIS (12) and WSJ (29)). For TI-digits and ATIS the word error rate increase was shown to be primarily due to a linguistic mismatch since using task-specific language models greatly reduces the error rate. For spontaneous WSJ dictation the BN models out-performed task-specific models trained on read speech data, which can be attributed to a better modelization of spontaneous speech effects (such as breath and filler words).

Methods to improve genericity of the models via multi-source training have been investigated. Multi-source training can be carried out in a variety of ways – by pooling data, by interpolating models or via single or multi-step model adaptation. The aim of multi-source training is to ob-

	Audio			*Text (words)*	
Language	*Radio-TV sources*	*Duration*	*Size*	*News*	*Com.Trans.*
English	ABC, CNN, CSPAN, NPR, PRI, VOA	200h	1.9M	790M	240M
French	Arte, TF1, A2, France-Info, France-Inter	50h	0.8M	300M	20M
German	Arte	20h	0.2M	260M	-
Mandarin	VOA, CCTV, KAZN	20h	0.7M(c)	200M(c)	-
Portuguese	9 sources	3.5h	∼35k	70M	-
Spanish	Televisa, Univision, VOA	30h	0.33M	295M	-
Arabic	tv: Aljazeera, Syria; radio: Orient, Elsharq, ...	50h	0.32M	200M	-

Table 1: Approximate sizes of the transcribed audio data and text corpora used for estimating acoustic and language models. For the text data, newspaper texts (News) and commercial transcriptions (Com.Trans.) are distinguished in terms of the millions of words (or characters for Mandarin). The American English, Spanish and Mandarin data are distributed by the LDC. The German data come from the EC OLIVE project and the French data partially from OLIVE and from the DGA. The Portuguese data are part of the 5h, 11 source Pilot corpus used in the EC ALERT project (data from 2 sources 24Horas and JornalTarde were reserved for the test set). The Arabic data were produced by the Vecsys company in collaboration with the DGA.

	Lexicon			*Language Model*		*Test*	
Language	*#phon.*	*size (words)*	*coverage*	*N-gram*	*ppx*	*Duration*	*%Werr*
English	48	65k	99.4%	11M fg, 14M tg, 7M bg	140	3.0h	20
French	37	65k	98.8%	10M fg, 13M tg, 14M bg	98	3.0h	23
German	51	65k	96.5%	10M fg, 14M tg, 8M bg	213	2.0h	25(n)-35(d)
Mandarin	39	40k+5k(c)	99.7%	19M fg, 11M tg, 3M bg	190	1.5h	20
Spanish	27	65k	94.3%	8M fg, 7M tg, 2M bg	159	1.0h	20
Portuguese	39	65k	94.0%	9M tg, 3M bg	154	1.5h	40
Arabic	40	60k	90.5%	11M tg, 6M bg	160	5.7h	20

Table 2: Some language characteristics. Specified for each language are: the number of phones used to represent lexical pronunciations, the approximate vocabulary size in words (characters for Mandarin) and lexical coverage (of the test data), the language model size and the perplexity, the test data duration (in hours) and the word/character error rates. For Arabic the vocabulary and language model are vowelized, however the word error rate does not include vowel or gemination errors. For German, separate word error rates are given for broadcast news (n) and documentaries (d).

tain generic models which are comparable in performance to the respective task-dependent models for all tasks under consideration. Compared to the results obtained with task-dependent acoustic models, both data pooling and sequential adaptation schemes led to better performance for ATIS and WSJ read, with slight degradations for BN and TI-digits (25).

In (9) cross-task porting experiments are reported for porting from an Italian broadcast news speech recognition system to two spoken dialogue domains. Supervised adaptation was shown to recover about 60% of the WER gap between the broadcast news acoustic models and the task-specific acoustic models. Language model adaptation using just 30 minutes of transcriptions was found to reduce the gap in perplexity between the broadcast news and task-dependent language models by 90%. It was also observed that the out-of-vocabulary rates for the task-specific language models are 3 to 5 times higher than the best adapted models, due to the relatively limited amount of task-specific data and the wide coverage of the broadcast news domain.

Techniques for large-scale discriminative training of the acoustic models of speech recognition systems using the maximum mutual information estimation (MMIE) criterion in place of conventional maximum likelihood estimation (MLE) have studied and it has been demonstrated that MMIE-based systems can lead to sizable reductions in word error rate on the transcription of conversational telephone speech (30). Experiments on discriminative training for cross-task genericity have made use of recognition systems trained on the low-noise North American Business News corpus of read newspaper texts and tested on television and radio Broadcast News data. These experiments showed that MMIE-trained models could indeed provide improved cross-task performance (11).

4.2. Reducing the need for annotated training data

With today's technology, the adaptation of a recognition system to a new task or new language requires the availability of sufficient amount of transcribed training data. When changing to new domains, usually no exact transcriptions of acoustic data are available, and the generation of such transcribed data is an expensive process in terms of manpower and time. On the other hand, there often exist incomplete information such as approximate transcriptions, summaries or at least key words, which can be used to provide supervision in what can be referred to as "informed speech

Amount of training data		*Language Model*
Raw	*Usable*	News.Com.Cap
10min	10min	53.1
1.5h	1h	33.3
50h	33h	20.7
104h	67h	19.1
200h	123h	18.0

Table 3: Supervised acoustic model training: Word error rate (%) on the 1999 evaluation test data for various conditions using one set of gender-independent acoustic models trained on subsets of the HUB4 training data with detailed manual transcriptions. The language model is trained on the available text sources, without any detailed transcriptions of the acoustic training data. The raw data reflects the size of the audio data before partitioning, and the usable data the amount of data used in training the acoustic models.

recognition". Depending on the level of completeness, this information can be used to develop confidence measures with adapted or trigger language models or by approximate alignments to automatic transcriptions. Another approach is to use existing recognizer components (developed for other tasks or languages) to automatically transcribe task-specific training data. Although in the beginning the error rate on new data is likely to be rather high, this speech data can be used to re-train a recognition system. If carried out in an iterative manner, the speech data base for the new domain can be cumulatively extended over time *without* direct manual transcription. This approach has been investigated in (18; 22; 23; 36; 39).

In order to give an idea of the influence of the amount of training data on system performance, Table 3 shows the performance of a 10xRealTime American English BN system for different amounts of manually annotated training data. The language model News.Com.Cap is trained on large text corpora, and results from the interpolation of individual language models trained on newspaper and newswires tests (790M words), commercially produced transcripts and closed-captions predating the test epoch (240M words). The word error is seen to rapidly decrease initially, with only a relatively small improvement above 30 hours of usable data. However, there is substantial information available in the language models. Table 4 summarizes supervised training results using substantially less language model training material. The second entry is for a language model estimated only on the newpaper texts (790M words), whereas for the remaining two language models were estimated on only 30 M words of texts (the last 2 months of 1997) and 1.8 M words (texts from December 26-31, 1997). It can be seen that the language model training texts have a large influence on the system performance, and even 30 M words is relatively small for the broadcast news transcription task.

The basic idea of light supervision is to use a speech recognizer to automatically transcribe unannotated data, thus generating "approximate" labeled training data. By iteratively increasing the amount of training data, more accurate acoustic models are obtained, which can then be used to transcribe another set of unannotated data. The manual work is considerably reduced, not only in generating the annotated corpus but also during the training procedure, since it is no longer necessary to extend the pronunciation lexicon to cover all words and word fragments occurring in the training data. In (22) it was found that somewhat comparable acoustic models could be estimated on 400 hours automatically annotated data from the TDT-2 corpus and 150 hours of carefully annotated data.

	Raw Acoustic training data		
Language model	*200 hours*	*1.5 hours*	*10 min*
News.Com.Cap, 65k	18.0	33.3	53.1
News, 65k	20.9	36.1	55.6
30 M words, 60k	24.1	40.8	60.2
1.8 M words, 40k	28.8	46.9	65.3

Table 4: Supervised acoustic model training: Reference word error rates (%) on the 1999 evaluation test data with varying amounts of manually annotated acoustic training data and a language model trained on 1.8 M and 30 M words of news texts from 1997.

Raw Acoustic training data		*WER (%)*
bootstrap models	10 min manual	65.3
1 (6 shows)	4 h	54.1
2 (+12 shows)	12 h	47.7
3 (+23 shows)	27 h	43.7
4 (+44 shows)	53 h	41.4
5 (+60 shows)	103 h	39.2
6 (+58 shows)	135 h	37.4

Table 5: Unsupervised acoustic model training: Word error rate (%) on the 1999 evaluation test data with varying amounts of automatically transcribed acoustic training data and a language model trained on 1.8 M words of news texts from 1997.

The effects of reducing the amount of supervision are summarized in Table 5. The first observation that can be made, is that even using a recognizer with an initial word error of 65% the procedure is converging properly by training acoustic models on automatically labeled data. This is even more surprising since the only supervision is via a language model trained on a small amount of text data predating the raw acoustic audio data. As the amount of automatically transcribed acoustic data is successively doubled, there are consistent reductions in the word error rate. While these error rates are still quite high compared to supervised training, retranscribing the same data (36) can be expected to reduce the word error rate further. (Recall that even with supervised acoustic model training trained on 200 hours of raw data the word error rate is 28.8% with this language model.)

4.3. Unsupervised Cross-Task Adaptation

An incremental unsupervised adaptation scheme was investigated for cross-task adaptation from the broadcast news task to the ATIS task (26). In this system-in-loop adaptation scheme, a first subset of the training data is automatically transcribed using the generic system. The acoustic and linguistic models of the generic system are then adapted with these automatically annotated data and the resulting models are used to transcribe another portion of the training data. One obvious use of this scheme is for online model adaptation in a dialog system.

Using about one-third (15 hours) of the ATIS training corpus transcribed with a BN system to adapt both the acoustic and language models, the word error rate is reduced from 20.8% to 6.9%. Transcribing the remaining data, and readapting the models reduces the word error to 5.5% (which can be compared to 4.7% for a task-specific system). Contrastive experiments have shown that this gain is somewhat equally split between adaptation of the acoustic and language models.

4.4. Cross Language Portability

The same basic idea was used to develop BN acoustic models for the Portuguese language for which substantially less manually transcribed data are available. RTP and INESC, partners in the Alert project (http:alert.uni-duisburg.de) provided 5 hours of manually annotated data from 11 different news programs. Two of the programs (82 minutes) were reserved for testing purposes (JornalTarde_20_04_00 and 24Horas_19_07_00). The remaining 3.5 hours of data were used for acoustic model training. The language model texts were obtained from the following sources: the Portuguese Newswire Text Corpus distributed by LDC (23M words from 1994-1998); Correio da Manha (1.6M words), Expresso (1.9M words from 2000-2001), and Jornal de Noticias (46M words, from 1996-2001), The recognition lexicon contains 64488 words. The pronunciations are generated by grapheme-to-phoneme rules, and use 39 phones.

Initial acoustic model trained on the 3.5 hours of available data were used to transcribe 30 hours of Portuguese TV broadcasts. These acoustic models had a word error rate of 42.6%. By training on the 30 hours of data using the automatic transcripts the word error was reduced to 39.1%. This preliminary experiment supports the feasibility of lightly supervised and unsupervised acoustic model training.

5. Conclusions

This paper has discussed the main issues in speech recognizer development and portability across languages and tasks. Today's most performant systems make use of statistical models, and therefore require large corpora for acoustic and language model training. However, acquiring these resources is both time-consuming, costly, and may be beyond the economic interest for many languages. Research is underway to reduce the need for manually annotated training data, thus reducing the human investment needed for system development when porting to another task or language. By eliminating the need for manual transcription, automated training can be applied to essentially unlimited quantities of task-specific training data.

The pronunciation lexicon still requires substantial manual effort for languages without straightfoward letter-to-sound correspondences, and to handle foreign words and proper names. For languages or dialects without a written form, the challenge is even greater, since important language modeling data are simply unavailable. Even if a transliterated form can be used, it is likely to be impractical to transcribe sufficient quantities of data for language model training.

In summary, our experience is that although general technologies and development strategies appear to port from one language to another, to obtain optimal performance language specificities must be taken into account. Efforts underway to improve the genericity of speech recognizers, and to reduce training costs will certainly help to enable the development of language technologies for minority languages and less economically promising applications.

Acknowledgments

This work has been partially financed by the European Commission and the French Ministry of Defense. I would like to thank the members of the Spoken Language Processing group at LIMSI who have participated to the research reported here and from whom I have borrowed experimental results.

REFERENCES

[1] *IEEE Workshop on Automatic Speech Recognition*, Special session on Multilingual Speech Recognition, Snowbird, Dec. 1995.

[2] *ICSLP'96*, Special session on "Multilingual Speech Processing," Philadelphia, PA, Oct. 1996.

[3] *Multi-Lingual Interoperabilty in Speech Technology*, RTO-NATO and ESCA ETRW, Leusden, Holland, Sept. 1999.

[4] M. Adda-Decker, "Towards Multilingual Inoperability in Speech Recognition," *Multi-Lingual Interoperabilty in Speech Technology*, RTO-NATO and ESCA ETRW, Leusden, Holland, 69-76, Sept. 1999.

[5] M. Adda-Decker, L. Lamel, "Pronunciation Variants Across Systems, Languages and Speaking Style," *Speech Communication*, "Special Issue on Pronunciation Variation Modeling", **29**(2-4): 83-98, Nov. 1999.

[6] M. Adda-Decker, G. Adda, L. Lamel, "Investigating text normalization and pronunciation variants for German broadcast transcription," *ICSLP'2000*, Beijing, China, Oct. 2000.

[7] L.R. Bahl, P. Brown, P. de Souza, R.L. Mercer, M. Picheny, "Acoustic Markov Models used in the Tangora Speech Recognition System," *ICASSP-88* **1**, pp. 497-500.

[8] C. Barras, E. Geoffrois, Z. Wu, M. Liberman, "Transcriber: development and use of a tool for assisting speech corpora production," *Speech Communication*, **33**(1-2): 5-22, Jan. 2001.

[9] N. Bertoldi, F. Brugnara, M. Cettolo, M. Federico, D. Giuliani, "From Broadcast News to Spontaneous Dialogue Transcription: Portability Issues," *ICASSP'01*, Salt Lake City, May 2001.

[10] L. Chen, L. Lamel, G. Adda, J.L. Gauvain, "Broadcast News Transcription in Mandarin," *ICSLP'2000*, Beijing, China, Oct. 2000.

[11] R. Cordoba, P. Woodland, M. Gales "Improved Cross-Task Recognition Using MMIE Training" *ICASSP'02*, Orlando, Fl, May 2002.

[12] D. Dahl, M. Bates *et al.*, "Expanding the Scope of the ATIS Task : The ATIS-3 Corpus," *ARPA Spoken Language Systems Technology Workshop*, Plainsboro, NJ, 3-8, 1994.

[13] C. Dugast, X. Aubert, R. Kneser, "The Philips Large Vocabulary Recognition System for American English, French, and German," *Eurospeech'95*, 197-200, Madrid, Sept. 1995.

[14] J. Glass, G. Flammia, D. Goodine, M. Phillips, J. Polifroni, J. Sakai, S. Seneff, V. Zue, "Multilingual spoken language understanding in the MIT Voyager system," *Speech Communication*, **17**(1-2): 1-18, Aug. 1995.

[15] D. Giuliani, M. Federico, "Unsupervised Language and Acoustic Model Adaptation for Cross Domain Portability" *ISCA ITRW 2001 Adaptation Methods For Speech Recognition*, Sophia-Antipolis, France, Aug. 2001.

[16] F. Jelinek, "Statistical Methods for Speech Recognition," Cambirdge: MIT Press, 1997.

[17] Katz, S.M. 1987. "Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer". *IEEE Trans. Acoustics, Speech, and Signal Processing*. **ASSP-35**(3): 400-401.

[18] T. Kemp, A. Waibel, "Unsupervised Training of a Speech Recognizer: Recent Experiments," *ESCA Eurospeech'99*, Budapest, Hungary, **6**, 2725-2728, Sept. 1999.

[19] J. Köhler, "Language-adaptation of multilingual phone models for vocabulary independent speech recognition tasks," *ICASSP'98*, **I**, 417-420, Seattle, May 1998.

[20] J. Kunzmann, K. Choukri, E. Janke, A. Kiessling, K. Knill, L. Lamel, T. Schultz, S. Yamamoto, "Portability of ASR Technology to new Languages: multilinguality issues and speech/text resources," slides from the panel discussion at *IEEE ASRU'01*, Madonna di Campiglio, Dec. 2001. (http://www.cs.cmu.edu/t̃anja/Papers/asru2001.ppt)

[21] L. Lamel, M. Adda-Decker, J.L. Gauvain, G. Adda, Spoken Language Processing in a Multilingual Context," *ICSLP'96*, 2203-2206, Philadelphia, PA, Oct. 1996.

[22] L. Lamel, J.L. Gauvain, G. Adda, "Lightly Supervised and Unsupervised Acoustic Model Training," *Computer, Speech & Language*, Jan. 2002.

[23] L. Lamel, J.L. Gauvain, G. Adda, "Unsupervised Acoustic Model Training," *IEEE ICASSP'02*, Orlando, Fl, May 2002.

[24] L. Lamel, F. Lefevre, J.L. Gauvain, G. Adda, "Portability issues for speech recognition technologies," *HLT'2001*, 9-16, San Diego, March 2001.

[25] F. Lefevre, J.L. Gauvain, L. Lamel, "Improving Genericity for Task-Independent Speech Recognition," *EuroSpeech'01*, Aalborg, Sep. 2001.

[26] F. Lefevre, J.L. Gauvain, L. Lamel, "Genericity and Adaptability Issues for Task-Independent Speech Recognition," *ISCA ITRW 2001 Adaptation Methods For Speech Recognition*, Sophia-Antipolis, France, Aug. 2001.

[27] R.G. Leonard, "A Database for speaker-independent digit recognition," *ICASSP*, 1984.

[28] D.S. Pallett, J.G. Fiscus, *et al.* "1998 Broadcast News Benchmark Test Results," *DARPA Broadcast News Workshop*, 5-12, Herndon, VA, Feb. 1999.

[29] D.B. Paul, J.M. Baker, "The Design for the Wall Street Journal-based CSR Corpus," *ICSLP'92*, Kobe, Nov. 1992.

[30] D. Povey, P. Woodland, "Improved Discriminative Training Techniques For Large Vocabulary Continuous Speech Recognition", *IEEE ICASSP'01*, Salt Lake City, May 2001.

[31] L.R. Rabiner, "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, **77**(2): 257-286. Feb, 1989.

[32] L.R. Rabiner, B.H. Juang, "An Introduction to Hidden Markov Models. *IEEE Acoustics Speech and Signal Processing ASSP Magazine*, **ASSP-3**(1): 4-16, Jan. 1986.

[33] T. Schultz, A. Waibel, "Language-independent and language-adaptive acoustic modeling for speech recognition," *Speech Communication*, **35** (1-2): 31-51, Aug. 2001.

[34] F. Van Eynde, D. Gibbon, eds., *Lexicon Development for Speech and Language Processing*, Dordrecht: Kluwer, 2000.

[35] A. Waibel, P. Geutner, L. Mayfield Tomokiyo, T. Schultz, M. Woszczyna, "Multilinguality in Speech and Spoken Language Systems," *Proceedings of the IEEE*, Special issue on Spoken Language Processing, **88**(8): 1297-1313, Aug. 2000.

[36] F. Wessel, H. Ney, "Unsupervised training of acoustic models for large vocabulary continuous speech recognition," *ASRU'01*, Madonna di Campiglio, Italy, Dec. 2001.

[37] S. Young, M. Adda-Decker, X. Aubert, C. Dugast, J.L. Gauvain, D.J. Kershaw, L. Lamel, D.A. van Leeuwen, D. Pye, A.J. Robinson, H.J.M. Steeneken, P.C. Woodland, "Multilingual Large Vocabulary Speech Recognition: The European SQALE Project," *Computer Speech and Language*, **11**(1): 73-89, Jan. 1997.

[38] S. Young, G. Bloothooft, eds., "Corpus Based Methods in Language and Speech Processing," Dordrecht: Kluwer, 1997.

[39] G. Zavaliagkos, T. Colthurst, "Utilizing Untranscribed Training Data to Improve Performance," *DARPA Broadcast News Transcription and Understanding Workshop*, Landsdowne, VA, 301-305, Feb. 1998.

Seven Dimensions of Portability for Language Documentation and Description

Steven Bird* **and Gary Simons†**

*Linguistic Data Consortium, University of Pennsylvania, 3615 Market Street, Philadelphia, PA 19104, USA
†SIL International, 7500 West Camp Wisdom Road, Dallas, TX 75236, USA

Abstract

The process of documenting and describing the world's languages is undergoing radical transformation with the rapid uptake of new digital technologies for capture, storage, annotation and dissemination. However, uncritical adoption of new tools and technologies is leading to resources that are difficult to reuse and which are less portable than the conventional printed resources they replace. We begin by reviewing current uses of software tools and digital technologies for language documentation and description. This sheds light on how digital language documentation and description are created and managed, leading to an analysis of seven portability problems under the following headings: content, format, discovery, access, citation, preservation and rights. After characterizing each problem we provide a series of value statements, and this provides the framework for a broad range of best practice recommendations.

1. Introduction

It is now easy to collect vast quantities of language documentation and description and store it in digital form. It is getting easier to transcribe the material and link it to linguistic descriptions. Yet how can we ensure that such material can be re-used by others, both now and into the future? While today's linguists can access documentation that is over 100 years old, much digital language documentation and description is unusable within a decade of its creation.

The fragility of digital records is amply demonstrated. For example, the interactive video disks created by the BBC Domesday Project are inaccessible just 15 years after their creation.[1] In the same way, linguists who are quick to embrace new technologies and create digital materials in the absence of archival formats and practices soon find themselves in technological quicksand.

The uncritical uptake of new tools and technologies is encouraged by sponsors who favor projects that promise to publish their data on the web with a search interface. However, these projects depend on technologies with life cycle of 3-5 years, and the resources they create usually do not outlive the project any longer than this.

This paper considers portability in the broadest sense: across different software and hardware platforms; across different scholarly communities (e.g. field linguistics, language technology); across different purposes (e.g. research, teaching, development); and across time. Portability is frequently treated as an issue for software, but here we will focus on data. In particular, we address portability for language documentation and description, and interpret these terms following Himmelmann:

> The aim of a language documentation is to provide a comprehensive record of the linguistic practices characteristic of a given speech community. Linguistic practices and traditions are manifest in two ways: (1) the observable linguistic behavior, manifest in everyday interaction between members of the speech community, and (2) the native speakers' metalinguistic knowledge, manifest in their ability to provide interpretations and systematizations for linguistic units and events. This definition of the aim of a language documentation differs fundamentally from the aim of language descriptions: a language description aims at the record of A LANGUAGE, with "language" being understood as a system of abstract elements, constructions, and rules that constitute the invariant underlying structure of the utterances observable in a speech community. (Himmelmann, 1998, 166)

We adopt the cover term DATA to mean any information that documents or describes a language, such as a published monograph, a computer data file, or even a shoebox full of hand-written index cards. The information could range in content from unanalyzed sound recordings to fully transcribed and annotated texts to a complete descriptive grammar. Beyond data, we are be concerned with language resources more generally, including tools and advice. By TOOLS we mean computational resources that facilitate creating, viewing, querying, or otherwise using language data. Tools include software programs, along with the digital resources that they depend on such as fonts, stylesheets, and document type definitions. By ADVICE we mean any information about what data sources are reliable, what tools are appropriate in a given situation, and what practices to follow when creating new data (Bird and Simons, 2001).

This paper addresses seven dimensions of portability for digital language documentation and description, identifying problems, establishing core values, and proposing best practices. The paper begins with a survey of the tools and technologies (§2), leading to a discussion of the problems that arise with the resources created using these tools and technologies (§3). We identify seven kinds of portability problem, under the headings of content, format, discovery, access, citation, preservation and rights. Next we give statements about core values in digital language documentation and description, leading to a series of "value statements", or requirements for best practices (§4), and followed up with collection of best practice recommendations (§5). The structure of the paper is designed to build consensus. For instance, readers who take issue with a best practice recommendation in §5 are encouraged to review the corresponding statement of values in §4 and either suggest a different practice which better implements the values, or else take issue with the value statement (then back up to the corresponding problem statement in §3, and so forth).

2. Tools and Technologies for Language Documentation and Description

Language documentation projects are increasing in their reliance on new digital technologies and software tools. This section contains a comprehensive survey of the range of practice, covering general purpose software, specialized tools, and digital technologies. Reviewing the available tools gives us a snapshot of how digital language documentation and description is created and managed, and provides a backdrop for our analysis of data portability problems.

2.1. General purpose tools

The most widespread practice in language documentation involves the use of office software. This software is readily available, often pre-installed, and familiar. Word processors have often been used as the primary storage for large lexical database, including a Yoruba lexicon with 30,000 entries split across 20 files. Frequently cited benefits are the WYSIWYG editing, the find/replace function, the possibility of cut-and-paste to create sublexicons, and the ease of publishing. Of course, a large fraction of the linguist's time is spent on maintaining consistency across multiple copies of the same data. Word processors have also been used for interlinear text, with three main approaches: fixed width fonts with hard spacing, manual setting of tabstops, and tables.[2] All methods require manual linebreaking, and significant labor if line width or point size are changed. Another kind of office software is the spreadsheet, which is often used for wordlists. Language documentation created using office software is normally stored in a secret proprietary format that is unsupported within 5-10 years. While other export formats are supported, they may loose some of the structure. For instance, part of speech may be distinguished in a lexical entry through the use of a particular font, and this information may be lost when the data is exported. Also, the portability of export formats may be compromised, by being laden with presentational markup.

A second class of general purpose software is the hypertext processors. Perhaps the first well-known application to language documentation was the original Macintosh hypercard stacks of *Sounds of the World's Languages* (Ladefoged and Maddieson, 1996). While it was easy to create a complex web of navigable pages, nothing could overcome the limitations of a vendor-specific hypertext language. More recently, the HTML standard and universal, free browsers have encouraged the creation of large amounts of hypertext for a variety of documentation types. For instance, we have interlinear text with HTML tables (e.g. Austin's Jiwarli fieldwork[3]), interlinear text with HTML frames (e.g. Culley's presentation of Apache texts[4]), HTML markup for lexicons, with hyperlinks from glossed examples and a thesaurus (e.g. Austin and Nathan's Gamilaraay lexicon[5]), gifs for representing IPA transcriptions (e.g. Bird's description of tone in Dschang[6]), and Javascript for image annotations (e.g. Poser's annotated photographs of gravestones engraved with Déné syllabics[7]). In all these cases, HTML is used as the primary storage format, not simply as a view on an underlying database. The intertwining of content and format makes this kind of language documentation difficult to maintain and re-use.

The third category of general purpose software is database packages. In the simplest case, the creator shares the database with others by requiring them to purchase the same package, and by shipping them a full dump of the database (e.g. the StressTyp database, which requires users to buy a copy of "4th Dimension"[8]). A more popular approach is to put the database on a web-server, and create a forms-based web interface that allows remote users to search the database without installing any software (e.g. the Comparative Bantu Online Lexical Database[9] and the Maliseet-Passamaquoddy Dictionary.[10]) Recently, some sites have started allowing database updates via the web (e.g. the Berkeley Interlinear Text Collector[11] and the Rosetta Project's site for uploading texts, wordlists and descriptions[12]).

2.2. Specialized tools

Over the last two decades, several dozen tools have been developed having specialized support for language documentation and description. We list a representative sample here; more can be found on SIL's page on *Linguistic Computing Resources*,[13] on the *Linguistic Exploration* page,[14] and on the *Linguistic Annotation* page.[15]

Tools for linguistic data management include Shoebox[16] and the Fieldworks Data Notebook.[17] Speech analysis tools include Praat[18] and SpeechAnalyzer.[19] Many specialized signal annotation tools have been developed, including CLAN,[20] EMU,[21] TableTrans, InterTrans, TreeTrans.[22] There are many orthographic transcription tools, including Transcriber[23] and MultiTrans.[24] There are morphological analysis tools, such as the Xerox finite state toolkit.[25] There are a wealth of concordance tools. Finally, some integrated multi-function systems have been created, such as LinguaLinks Linguistics Workshop.[26]

In order to do their specialized linguistic processing, each of these tools depends on some model of linguistic information. Time-aligned transcriptions, interlinear texts, syntax trees, lexicons, and so forth, all require suitable data structures and file formats. Given that most of these tools have been developed in isolation, they typically employ incompatible models and formats. For example, data created with an interlinear text tool cannot be subsequently annotated with syntactic information without losing the interlinear annotations. When interfaces and formats are open and documented, it is occasionally possible to cobble the tools together in support of a more complex need. However, the result is a series of increasingly baroque and decreasingly portable approximations to the desired solution. Computational support for language documentation and description is in disarray.

2.3. Digital technologies

A variety of digital technologies are now used in language documentation thanks to sharply declining hardware costs. These include technologies for digital signal capture (audio, video, physiological) and signal storage (hard disk, CD-R, DVD-R, minidisc). Software technologies are also playing an influential role as new standards are agreed. The

most elementary and pervasive of these is the hyperlink, which makes it possible to connect linguistic descriptions to the underlying documentation (e.g. from an analytical transcription to a recording). Such links streamline the descriptive process; checking a transcription can be done with mouse clicks instead of digging out a tape or finding an informant. The ability to navigate from description to documentation also facilitates analysis and verification. Software technologies and standards have given rise to the internet which permits low-cost dissemination of language resources. Notably, it is portability problems with these tools and formats that prevents these basic digital technologies from having their full impact. The download instructions for the Sumerian lexicon[27] typify the problems (hyperlinks are underlined):

> Download the Sumerian Lexicon as an Adobe Acrobat PDF file. In order to minimize downloads of this large file, once you have it, please use your Acrobat Reader to save it and retrieve it to and from your own desktop.
>
> Download the Sumerian Lexicon as a Word for Windows 6.0 file in a self-extracting WinZip archive.
>
> Download the same contents in a non-executable zip file.
>
> Includes version 2 of the Sumerian True Type font for displaying transliterated Sumerian. Add the font to your installed Windows fonts at Start, Settings, Control Panel, Fonts. To add the Sumerian font to your installed Windows fonts, you select File and Add New Font. Afterwards, make sure that when you scroll down in the Fonts listbox, it lists the Sumerian font. When you open the SUMERIAN.DOC file, ensure that at File, Templates, or at Tools, Templates and Add-Ins, there is a valid path to the enclosed SUMERIAN.DOT template file. If you do not have Microsoft's Word for Windows, you can download a free Word for Windows viewer at Microsoft's Web Site.
>
> Download Macintosh utility UnZip2.0.1 to uncompress IBM ZIP files. To download and save this file, you should have Netscape set in Options, General Preferences, Helpers to handle hqx files as Save to Disk. Decode this compressed file using Stuffit Expander. Download Macintosh utility TTconverter to convert the IBM format SUMERIAN.TTF TrueType font to a System 7 TrueType font. Decode this compressed file using Stuffit. Microsoft Word for the Macintosh can read a Word for Windows 6.0 document file. There is no free Word for Macintosh viewer, however.

2.4. Digital Archives

Recently several digital archives of language documentation and description have sprung up, such as the Archive of the Indigenous Languages of Latin America,[28] and the Rosetta Project's Archive of 1000 Languages.[29] These exist alongside older archives which are in various stages of digitizing their holdings: the Archive of the Alaska Native Language Center,[30] the LACITO Linguistic Data Archive,[31] and the US National Anthropological Archives.[32] These archives and many others are surveyed on the *Language Archives* page.[33] Under the aegis of OLAC, the *Open Language Archives Community*,[34] the notion of language archive has been broadened to include archives of linguistic software, such as the Natural Language Software Registry[35]

These archives face many challenges, the most significant being the lack of funding. Other challenges may include: identifying, adapting and deploying digital archiving standards; setting up key operational functions such as offsite backup, migration to new digital formats and media over time, and the support of new access modes (e.g. search facilities) and delivery formats (e.g. streaming media); and obtaining the long-term support of a major institution to assure contributors and users that the materials will be available over the long term.

3. Seven Problems for Portability

With the rapid uptake of new digital technologies, many creators of language documentation and description are ignoring the question of portability, with the unfortunate consequence that the fruits of their labors are likely to be unusable within 5-10 years. In this section we discuss seven critical problems for the portability of this data.

3.1. Content

Many potential users of language data are interested in assimilating multiple descriptions of a single language to gain an understanding of the language which is as comprehensive as possible. Many users are interested in comparing the descriptions of different languages in order to apply insights from one analysis to another or to test a typological generalization. However, two descriptions may be difficult to compare or assimilate because they have used terminology differently, or because the documentation on which the descriptions are based is unavailable.

Language documentation and description of all types depends critically on technical vocabulary, and ambiguous terms compromise portability. For instance, the symbols used in phonetic transcription have variable interpretation depending on the descriptive tradition: "it is crucial to be aware of the background of the writer when interpreting an unexplained occurrence of [y]" (Pullum and Ladusaw, 1986, 168). In morphosyntax, the term "absolutive" can refer to one of the cases in an ergative language, or to the unpossessed form of a noun (in the Uto-Aztecan tradition) (Lewis et al., 2001, 151), and a correct interpretation of the term depends on an understanding of the linguistic context.

This terminological variability leads to problems for retrieval. Suppose that a linguist wanted to search the full-text content of a large collection of digital language data, in order to discover which other languages have relevant phenomena. Since there are no standard ontologies, the user will discover irrelevant documents (low precision) and will fail to discover relevant documents (low recall). In order to carry out a comprehensive search, the user must know all the ways in which a particular phenomena is described. Even once a set of descriptions are retrieved, it will generally not be possible to draw reliable comparisons between the descriptions of different languages.

The content of two descriptions may also be difficult to reconcile because it is not possible to verify them with respect to the language documentation that they cite. For example, when two descriptions of the same language provide different phonetic transcriptions of the same word, is this the result of a typographical error, a difference in transcription practice, or a genuine difference between two speech varieties? When two descriptions of different languages report that the segmental inventories of both languages contain a [k], what safe conclusions can be drawn about how similar the two sounds are? Since the underlying documentation is not available, such questions cannot be resolved, making it difficult to re-use the resources.

While the large-scale creation of digital language resources is a recent phenomenon, the language documentation community has been active since the 19th century, and much earlier in some instances. At

risk of oversimplifying, a widespread practice over this extended period has been to collect wordlists and texts and to write descriptive grammars. With the arrival of new digital technologies it is easy to transfer the whole endeavor from paper to computer, and from tape recorder to hard disk, and to carry on just as before. Thus, new technologies simply provide a better way to generate the old kinds of resources. Of course this is a wasted opportunity, since the new technologies can also be used to create digital multimedia recordings of rich linguistic events. Such rich recordings often capture items which turn out to be useful in later linguistic analysis, and have immense intrinsic value as a record of cultural heritage for future generations. However, managing digital technologies in less controlled situations leads to many technical and logistical issues, and there are no guidelines for integrating new technologies into new documentary practices.

3.2. Format

Language data frequently ends up in a secret proprietary format using a non-standard character encoding. To use such data one must often purchase commercial software then install it on the same hardware and under the same operating system used by the creator of the data.

Other formats, while readable outside the tool that created them, remain non-portable when they are not explicitly documented. For example, the interpretation of the field names in Shoebox format may not be documented, or the documentation may become separated from the data file, making it difficult to guess what the different fields signify.

The developers of linguistic tools must frequently parse presentational formats. For example, the occurrence of `<b>[n]</b>` in a lexical entry might indicate that this is an entry for a noun. More difficult cases involve subtle context-dependencies. This presentational markup obscures the structure and interpretation of the linguistic content. Conversely, in the absence of suitable browsing and rendering tools, end-users must attempt to parse formats that were designed to be read only by machines.

3.3. Discovery

Digital language data is often presented as a physical or digital artefact with no external description. Like a book without a cover page or a binary file called `dict.dat`, one is forced to expend considerable effort to discover the subject matter and the nature of the content. Organized collections – such as the archive of a university linguistics department – may provide some metadescription, but it is likely to use a parochial format and idiosyncratic descriptors. If they are provided, key descriptors like *subject language* and *linguistic type* are usually given in free text rather than a controlled vocabulary, reducing precision and recall. As a consequence, discovering relevant language resources is extremely difficult, and depends primarily on word-of-mouth and queries posted to electronic mailing lists. Thus, new resource creation efforts may proceed in ignorance of prior and concurrent efforts, wasting scarce human resources.

In some cases, one may obtain a resource only to discover upon closer inspection that it is in an incompatible format. This is the flip-side of the discovery problem. Not only do we need to know that a resource exists, but also that it is relevant. When resources are inadequately described, it is difficult (and often impossible) to find a relevant resource, a huge impediment to portability.

3.4. Access

In the past, primary documentation was usually not disseminated. To listen to a field recording it was often necessary to visit the laboratory of the person who collected the materials, or to make special arrangements for the materials to be copied and posted. Digital publication on the web has alleviated this problem, although projects usually refrain from full dissemination by limiting access via a restrictive search interface. This means that only selected portions of the documentation can be downloaded, and that all access must use categories predefined by the provider. Moreover, these web forms only have a lifespan of 3-5 years, relying on ad hoc CGI scripts which may cease working when the interpreter or webserver are upgraded. Lack of full access means that materials are not portable. More generally, people have often conflated digital publication with web publication, and publish high-bandwidth materials on the web which would be more usable if published on CD or DVD.

Many language resources have applications beyond those envisaged by their creators. For instance, the Switchboard database (Godfrey et al., 1992), collected for the development of speaker-independent automatic speech recognition, has since been used for studies of intonation and disfluency. Often this redeployment is prevented through the choice of formats. For instance, publishing conversation transcripts in the Hub-4 SGML format does not facilitate their reuse in, say, conversational analysis. In other cases, redeployment is prevented by the choice of media. For instance, an endangered language dictionary published only on the web will not be accessible to speakers of that language who live in a village without electricity.

One further problem for access deserves mention here. It sometimes happens that an ostensibly available resource turns out not to be available after all. One may discover the resource because its creator cited it in a manuscript or an annual research report. Commonly, a linguist wants to derive recognition for the labor that went into creating primary language documentation, but does not want to make the materials available to others until deriving maximum personal benefit. Two tactics are to cite unresolved, non-specific intellectual property rights issues, and to repeatedly promise but to never finally deliver. Despite its many guises, this problem has two distinguishing features: someone draws attention to a resource in order to derive credit for it – "parading their riches" as Mark Liberman (pers. comm.) has aptly described it – and then applies undocumented or inconsistent restrictions to prevent access. The result may be frustration that a needed resource is withheld, leading to wasted effort or a frozen project, or to suspicion that the resource is defective and so must be protected by a smoke screen.

3.5. Citation

Research publications are normally required to provide full bibliographic citation of the materials used in conducting the research. Citation standards are high for conventional resources (such as other publications), but are much lower for language resources which are usually incorrectly cited, or not cited at all. This makes it difficult to find out what resource was used in conducting the research and, in the reverse direction, it is impossible to use a citation index to discover all the ways in which a given resource has been applied.

Often a language resource is available on the web, and it is convenient to have the uniform resource locater (URL) since this may offer the most efficient way to obtain the resource. However, URLs can fail as a persistent citation in two ways: they may simply break, or they may cease to reference the same item. URLs break when the resource is moved or when some piece of the supporting infrastructure, such as a database server, ceases to work. Even if a URL does not break, the item it references may be mutable, changing over time. Language resources published on the web are usually not versioned, and a third-party description of some item may cease to be valid if that item is changed. Publishing a digital artefact, such as a CD, with a unique identifier, such as an ISBN, avoids this problem.

Citation goes beyond bibliographic citation of a complete item. We may want to cite some component of a resource, such as a specific narrative or lexical entry. However, the format may not support durable citations to internal components. For instance, if a lexical entry is cited by a URL which incorporates its lemma, and if the spelling of the lemma is altered, then the URL will not track the change. In sum, language documentation and description is not portable if the incoming and outgoing links to related materials are fragile.

3.6. Preservation

The digital technologies used in language documentation and description greatly enhance our ability to create data while simultaneously compromising our ability to preserve it. Relative to paper copy which can survive for hundreds of years, digitized materials are evanescent because they use some combination of binary formats with undocumented character encodings saved on non-archival media and physically stored with no ongoing administration for backups and migration to new media. Presentational markup with HTML and interactive content with Javascript and specialized browser plugins require future browsers to be backwards-compatible. Furthermore, primary documentation may be embodied in the interactive behavior of the resource (e.g. the gloss of the text under the mouse may show up in the browser status line, using the Javascript "mouseover" effect). Consequently, digital resources – especially dynamic or interactive ones – often have a short lifespan, and typically become unusable 3-5 years after they are actively maintained.

3.7. Rights

A variety of individuals and institutions may have intellectual property vested in a language resource, and there is a complex terrain of legal, ethical and policy issues (Liberman, 2000). In spite of this, most digital language data is disseminated without identifying the copyright holder and without any license delimiting the range of acceptable uses of the material. Often people collect or redistribute materials, or create derived works without securing the necessary permissions. While this is often benign (e.g. when the resources are used for research purposes only), the researcher risks legal action, or having to restrict publication, or even having to destroy primary materials. To avoid any risk one must avoid using materials whose property rights are in doubt. In this way, the lack of documented rights restrict the portability of the language resource.

Sometimes resources are not made available on the web for fear that they will get into the wrong hands or be misused. However, this confuses medium with rights. The web supports secure data exchange between authenticated parties (through data encryption) and copyright statements together with licenses can be used to restrict uses. More sophisticated models for managing digital rights are emerging (Iannella, 2001). The application of these techniques to language resources is unexplored, and we are left with an all-or-nothing situation, in which the existence of any restriction prevents access across the board.

3.8. Special challenges for little-studied languages

Many of the problems reported above also apply to little-studied languages, though some are greatly exacerbated in this context. The small amount of existing work on the language and the concomitant lack of established documentary practices and conventions may lead to especially diverse nomenclature. Inconsistencies within or between language descriptions may be harder to resolve because of the lack of significant documentation, the limited access to speakers of the language, and the limited understanding of dialect variation. Open questions in one area of description (e.g. the inventory of vowel phonemes) may multiply the indeterminacies in another (e.g. for transcribed texts). More fundamentally, existing documentation and description may be virtually impossible to discover and access, owing to its fragmentary nature.

The acuteness of these portability problems for little-studied languages can be highlighted by comparison with well-studied languages. In English, published dictionaries and grammars exist to suit all conceivable tastes, and it therefore matters little (relatively speaking) if none of these resources is especially portable. However, when there is only one dictionary for the language, it must be pressed into a great range of services, and significant benefits will come from maximizing portability.

This concludes our discussion of portability problems arising from the way new tools and technologies are being used in language documentation and description. The rest of this paper responds to these problems, by laying out the core values that lead to requirements for best practices (§4) and by providing best practice recommendations (§5).

4. Value Statements

Best practice recommendations amount to a decision about which of several possible options is best. The notion of best always involves a value judgment. Therefore, before making our recommendations, we articulate the values which motivate our choices. Our use of "we" is meant to include the reader and the wider language resources community who share these values.

4.1. Content

TERMINOLOGY. We value the ability of users to identify the substantive similarities and differences between two resources. Thus the best practice is one that makes it easy to associate the comparable parts of unrelated resources.

ACCOUNTABILITY. We value the ability of researchers to verify language descriptions. Thus the best practice is one that provides the documentation that lies behind the description.

RICHNESS. We value the documentation of little-studied languages. Thus the best practice is one that establishes a record that is sufficiently broad in scope and rich in detail that future generations can experience and study the language, even when no speakers remain.

4.2. Format

OPENNESS. We value the ability of any potential user to make use of a language resource without needing to obtain unique or proprietary software. Thus the best practice is one that puts data into a format that is not proprietary.

DOCUMENTATION. We value the ability of potential users of a language resource to understand its internal structure and organization. Thus the best practice is one that puts data into a format that is documented.

MACHINE-READABLE. We value the ability of users of a language resource to write programs to process the resource. Thus the best practice is one that puts the resource into a well-defined format which can be submitted to automatic validation.

HUMAN-READABLE. We value the ability of users of a language resource to browse the content of the resource. Thus the best practice is one that provides a human-digestible version of a resource.

4.3. Discovery

EXISTENCE. We value the ability of any potential user of a language resource to learn of its existence. Thus the best practice is one that makes it easy for anyone to discover that a resource exists.

RELEVANCE. We value the ability of potential users of a language resource to judge its relevance without first having to obtain a copy. Thus the best practice is one that makes it easy for anyone to judge the relevance of a resource based on its metadescription.

4.4. Access

COMPLETE. We value the ability of any potential user of a language resource to access the complete resource, not just a limited interface to the resource. Thus the best practice is one that makes it easy for anyone to obtain the entire resource.

UNIMPEDED. We value the ability of any potential user of a language resource to follow a well-defined procedure to obtain a copy of the resource. Thus the best practice is one in which all available resources have a clearly documented method by which they may be obtained.

UNIVERSAL. We value the ability of potential users to access a language resource from whatever location they are in. Thus the best practice is one that makes it possible for users to access some version of the resource regardless of physical location and access to computational infrastructure.

4.5. Citation

CREDIT. We value the ability of researchers to be properly credited for the language resources they create. Thus the best practice is one that makes it easy for authors to correctly cite the resources they use.

PROVENANCE. We value the ability of potential users of a language resource to know the provenance of the resources it is based on. Thus the best practice is one that permits resource users to navigate a path of citations back to the primary linguistic documentation.

PERSISTENCE. We value the ability of language resource creators to endow their work with a permanent digital identifier which resolves to an instance of the resource. Thus the best practice is one that associates resources with persistent digital identifiers.

IMMUTABILITY. We value the ability of potential users to cite a language resource without that resource changing and invalidating the citation. Thus the best practice is one that makes it easy for authors to freeze and version their resources.

COMPONENTS. We value the ability of potential users to cite the component parts of a language resource. Thus the best practice is one that ensures each sub-item of a resource has a durable identifier.

4.6. Preservation

LONG-TERM. We value access to language resources over the very long term. Thus the best practice is one which ensures that language resources will still be usable many generations into the future.

COMPLETE. We value the ability of future users of a language resource to access the complete resource as experienced by contemporary users. Thus the best practice is one which preserves fragile aspects of a resource (such as dynamic and interactive content) in a durable form.

4.7. Rights

DOCUMENTATION. We value the ability of potential users of a language resource to know the restrictions on permissible uses of the resource. Thus the best practice is one that ensures that potential users know exactly what they are able to do with any available resource.

RESEARCH. We value the ability of potential users of a language resource to use it in personal scholarship and academic publication. Thus the best practice is one that ensures that the terms of use on resources do not hinder individual study and academic research.

5. Best Practice Recommendations

This section recommends best practices in support of the values set out in §4. We believe that the task of identifying and adopting best practices rests with the community, and we believe that OLAC, the *Open Language Archives Community*, provides the necessary infrastructure for identifying community-agreed best practices. Here, however, we shall attempt to give some broad guidelines to be fleshed out in more detail later, by ourselves and also, we hope, by other members of the language resources community.

5.1. Content

TERMINOLOGY. Map linguistic terminology and descriptive markup elements to a common ontology of linguistic terms. This applies to the obvious candidates such as morphosyntactic abbreviations and structural markup, but also to less obvious cases such as the phonological description of the symbols used in transcription. (NB vocabularies can be versioned and archived in an OLAC archive; archived descriptions cite their vocabularies using the `Relation` element.)

ACCOUNTABILITY. Provide the full documentation on which language descriptions are based. For example, where a narrative is transcribed, provide the primary recording (without segmenting it into multiple sound clips). Create time-aligned transcriptions to facilitate verification.

RICHNESS. Make rich records of rich interactions, especially in the case of endangered languages or genres. Document the "multimedia linguistic field methods" that were used. Provide theoretically neutral descriptions of a wide range of linguistic phenomena.

5.2. Format

OPENNESS. Store all language documentation and description in an open format. Prefer formats supported by multiple third-party software tools. NB some proprietary formats are open, e.g. Adobe Portable Document Format (PDF) and MPEG-1 Audio Layer 3 (MP3).

DOCUMENTATION. Provide all language documentation and description in a self-describing format (preferably XML). Provide detailed documentation of the structure and organization of the format. Encode the characters with Unicode. Try to avoid Private Use Area characters, but if they are used document them fully. Document any 8-bit character encodings. (OLAC will be providing detailed guidelines for documenting non-standard character encodings.)

MACHINE-READABLE. Use open standards such as XML and Unicode, along with Document Type Definitions (DTDs), XML Schemas and/or other definitions of well-formedness which can be verified automatically. Archive the format definition, giving each version its own unique identifier. When archiving data in a given format, reference the archived definition of that format. Avoid freeform editors for structured information (e.g. prefer Excel or Shoebox over Word for storing lexicons).

HUMAN-READABLE. Provide one or more human readable version of the material, using presentational markup (e.g. HTML) and/or other convenient formats. Proprietary formats are acceptable for delivery as long as the primary documentation is stored in a non-proprietary format.

N.B. Format is a critical area for the definition of best practices. We propose that recommendations in this area be organized by type (e.g. audio, image, text), possibly following the inventory of types identified in the Dublin Core metadata set.[36]

5.3. Discovery

EXISTENCE. List all language resources with an OLAC data provider. Any resource presented in HTML on the web should contain metadata with keywords and description for use by conventional search engines.

RELEVANCE. Follow the OLAC recommendations on best practice for metadescription, especially concerning language identification and linguistic data type. This will ensure the highest possibility of discovery by interested users in the OLAC union catalog hosted by Linguist.[37]

5.4. Access

COMPLETE. Publish complete primary documentation. Publish the documentation itself, and not just an interface to it, such as a web search form.

UNIMPEDED. Document all access methods and restrictions along with other metadescription. Document charges and expected delivery time.

UNIVERSAL. Make all resources accessible by any interested user. Publish digital resources using appropriate delivery media, e.g. web for small resources, and CD/DVD for large resources. Where appropriate, publish corresponding print versions, e.g. for the dictionary of a little-studied language.

5.5. Citation

CREDIT, PROVENANCE. Furnish complete bibliographic data for all language resources created. Provide complete citations for all language resources used. Document the relationship between resources in the metadescription (NB in the OLAC context, use the `Relation` element).

PERSISTENCE. Ensure that resources have a persistent identifier, such as an ISBN or a persistent URL (e.g. a Digital Object Identifier[38]). Ensure that at least one persistent identifier resolves to an instance of the resource or to detailed information about how to obtain the resource.

IMMUTABILITY. Provide fixed versions of a resource, either by publishing it on a read-only medium, and/or submitting it to an archive which ensures immutability. Distinguish multiple versions with a version number or date, and assign a distinct identifier to each version.

COMPONENTS. Provide a formal means by which the components of a resource may be uniquely identified. Take special care to avoid the possibility of ambiguity, such as arises when lemmas are used to identify lexical entries, and where multiple entries can have the same lemma.

5.6. Preservation

LONG-TERM. Commit all documentation and description to a digital archive which can credibly promise long-term preservation and access. Ensure that the archive

satisfies the key requirements of a well-founded digital archive (e.g. implements digital archiving standards, provides offsite backup, migrates materials to new formats and media/devices over time, is committed to supporting new access modes and delivery formats, has long-term institutional support, and has an agreement with a national archive to take materials if the archive folds). Archive physical versions of the language documentation and description (e.g. printed versions of documents; any tapes from which online materials were created). Archive electronic documents using type 1 (scalable) fonts in preference to bitmap fonts.

COMPLETE. Ensure that all aspects of language documentation and description accessible today are accessible in future. Ensure that any documentary information conveyed via dynamic or interactive behaviors is preserved in a purely declarative form.

5.7. Rights

DOCUMENTATION. Ensure that the intellectual property rights relating to the resource are fully documented.

RESEARCH. Ensure that the resource may be used for research purposes.

6. Conclusion

Today, the community of scholars engaged in language documentation and description exists in a cross-over period between the paper-based era and the digital era. We are still working out how to preserve knowledge that is stored in digital form. During this transition period, we observe unparalleled confusion in the management of digital language documentation and description. A substantial fraction of the resources being created can only be re-used on the same software/hardware platform, within the same scholarly community, for the same purpose, and then only for a period of a few years. However, by adopting a range of best practices, this specter of chaos can be replaced with the promise of easy access to highly portable resources.

Using tools as our starting point, we described a diverse range of practices and discussed their negative implications for data portability along seven dimensions, leading to a collection of advice for how to create portable resources. These three categories, tools, data, and advice, are three pillars of the infrastructure provided by OLAC, the Open Language Archives Community (Bird and Simons, 2001). Our best practice recommendations are preliminary, and we hope they will be fleshed out by the community using the OLAC Process.[39]

We leave off where we began, namely with tools. It is our use of the new tools which have led to data portability problems. And it is only with new tools, supporting the kinds of best practices we recommend, which will address these problems. An archival format is useless unless there are tools for creating, managing and browsing the content stored in that format. Needless to say, no single organization has the resources to create the necessary tools, and no third party developing general-purpose office software will address the unique needs of the language documentation and description community. We need nothing short of an open source revolution, leading to new specialized tools based on shared data models for all of the basic linguistic types, and connected to portable data formats.

Acknowledgements

This research is supported by NSF Grant No. 9983258 "Linguistic Exploration" and Grant No. 9910603 "International Standards in Language Engineering."

Notes

[1] http://www.observer.co.uk/uk_news/story/0,6903,661093,00.html
[2] http://www.linguistics.ucsb.edu/faculty/cumming/WordForLinguists/Interlinear.htm
[3] http://www.linguistics.unimelb.edu.au/research/projects/jiwarli/gloss.html
[4] http://etext.lib.virginia.edu/apache/ChiMesc2.html
[5] http://www3.aa.tufs.ac.jp/~austin/GAMIL.HTML
[6] http://www.ldc.upenn.edu/sb/fieldwork/
[7] http://www.cnc.bc.ca/yinkadene/dakinfo/dulktop.htm
[8] http://fonetiek-6.leidenuniv.nl/pil/stresstyp/stresstyp.html
[9] http://www.linguistics.berkeley.edu/CBOLD/
[10] http://ultratext.hil.unb.ca/Texts/Maliseet/dictionary/index.html
[11] http://ingush.berkeley.edu:7012/BITC.html
[12] http://www.rosettaproject.org:8080/live/
[13] http://www.sil.org/linguistics/computing.html
[14] http://www.ldc.upenn.edu/exploration/
[15] http://www.ldc.upenn.edu/annotation/
[16] http://www.sil.org/computing/shoebox/
[17] http://fieldworks.sil.org/
[18] http://fonsg3.hum.uva.nl/praat/
[19] http://www.sil.org/computing/speechtools/speechanalyzier.htm
[20] http://childes.psy.cmu.edu/
[21] http://www.shlrc.mq.edu.au/emu/
[22] http://sf.net/projects/agtk/
[23] http://www.etca.fr/CTA/gip/Projets/Transcriber/
[24] http://sf.net/projects/agtk/
[25] http://www.xrce.xerox.com/research/mltt/fst/
[26] http://www.sil.org/LinguaLinks/LingWksh.html
[27] http://www.sumerian.org/
[28] http://www.ailla.org/
[29] http://www.rosettaproject.org/
[30] http://www.uaf.edu/anlc/
[31] http://195.83.92.32/index.html.en
[32] http://www.nmnh.si.edu/naa/
[33] http://www.ldc.upenn.edu/exploration/archives.html
[34] http://www.language-archives.org/
[35] http://registry.dfki.de/
[36] http://dublincore.org/
[37] http://www.linguistlist.org/
[38] http://www.doi.org/
[39] http://www.language-archives.org/OLAC/process.html

7. References

Steven Bird and Gary Simons. 2001. The OLAC metadata set and controlled vocabularies. In *Proceedings of ACL/EACL Workshop on Sharing Tools and Resources for Research and Education*. http://arXiv.org/abs/cs/0105030.

J. J. Godfrey, E. C. Holliman, and J. McDaniel. 1992. Switchboard: A telephone speech corpus for research and development. In *Proceedings of the IEEE Conference on Acoustics, Speech and Signal Processing*, volume I, pages 517–20. http://www.ldc.upenn.edu/Catalog/LDC93S7.html.

Nikolaus P. Himmelmann. 1998. Documentary and descriptive linguistics. *Linguistics*, 36:161–195.

Renato Iannella. 2001. Digital rights management (DRM) architectures. *D-Lib Magazine*, 7(6), June.

Peter Ladefoged and Ian Maddieson. 1996. *The Sounds of the World's Languages*. Cambridge, MA: Blackwell.

William Lewis, Scott Farrar, and D. Terence Langendoen. 2001. Building a knowledge base of morphosyntactic terminology. In Steven Bird, Peter Buneman, and Mark Liberman, editors, *Proceedings of the IRCS Workshop on Linguistic Databases*, pages 150–156. http://www.ldc.upenn.edu/annotation/database/.

Mark Liberman. 2000. Legal, ethical, and policy issues concerning the recording and publication of primary language materials. In Steven Bird and Gary Simons, editors, *Proceedings of the Workshop on Web-Based Language Documentation and Description*. http://www.ldc.upenn.edu/exploration/expl2000/papers/.

Geoffrey K. Pullum and William A. Ladusaw. 1986. *Phonetic Symbol Guide*. The University of Chicago Press.

Challenges and Opportunities in Portability of Human Language Technologies

Bojan Petek

Interactive Systems Laboratory
University of Ljubljana, Faculty of Natural Sciences and Engineering
Snežniška 5, 1000 Ljubljana, Slovenia
Bojan.Petek@Uni-Lj.si

Abstract

Availability of language resources (LR) is a decisive element that influences the vital issues such as linguistic and cultural identity and use of a particular language in information society. Specifically, natural interactivity in information age relies on the existence of mature Human Language Technologies (HLT) that need substantial amount of appropriate LR to be developed. Additional challenge is that research addressing portability issues in HLT is still in its infancy. In perspective, it is reasonably to expect that advances in construction of the multilingual LR [Bird, 2001] and insights into the portability issues of HLT [Lamel, 2002] could potentially lower the digital divide and increase the visibility of a much larger pool of languages than experienced today. In order to achieve this challenging goal, it is proposed to initiate an international network of excellence (NoE) on HLT portability that would complement the already established activities on HLT resources. Such NoE could actively contribute to and advise the national HLT efforts aiming to achieve the grand goal of a non-exclusive information society.

1. Introduction

Several research programs have already focused towards the next generation of intelligent conversational interfaces. Their fundamental goal is to create speech-enabled multi-modal systems that scale gracefully across modalities. Such interfaces typically include speech, graphics, gesture, and computer vision. They are capable of supporting complex conversational interaction comparable to the human-human natural interactivity.

The natural interactive systems integrate spoken language dialogue systems, multimodal communication systems, and web-based data handling tools. The long-term goal of computer-mediated natural interactivity is to transform the present computer systems to become transparent in communication tasks and to support similar communication patterns as those experienced in usual interpersonal communication.

From the theoretical point of view, traditional Human-Computer Interaction (HCI) model has recently evolved towards the enhanced Human-Human Computer Interaction (HHCI) model that also includes the information and communication technologies. The HHCI model positions the computer system as a networked facilitator of information access and sharing. Typical applications include video conferencing or distributed multimedia information systems.

In the following, this paper aims to reflect the on-going research efforts by providing an overview of the challenges and opportunities addressed by the EU HLT projects and the US DARPA programs with relevance to the portability issues in HLT.

2. EU HLT Projects

At the time of writing about 50 EU HLT projects are detailed on the HLTCentral, the gateway to speech and language technology opportunities. Some projects with high relevance to the focus of this paper (ie, portability issues in human language technologies) are overviewed in order to outline the challenges (ie, project objectives) and opportunities (ie, expected outcome and innovation perspectives) these projects describe at the HLTCentral.

2.1. CORETEX

Improving Core Speech Recognition Technology, [wwwCORETEX]

The CORETEX project aims to improve the core speech recognition technologies. This 3 year EU project started in April 2000. The project consortium includes RWTH Aachen (Germany), University of Cambridge (UK), Istituto Trentino di Cultura ITC - IRST (Italy), and Centre National de la Recherche Scientifique – CNRS (France). The proposed work is motivated by observation that the current commercial speech recognition systems perform fairly well for a limited number of tasks and languages. On the other hand, these systems are very difficult to adapt to new domains, languages, and/or changing acoustic environmental conditions. The main obstacle in efficient porting of speech applications to new tasks, languages, or new environments is a requirement for substantial investment of time, money, and expertise.

Therefore, the overall project objective is to devise generic speech recognition technologies that perform well in a task independent way. List of the CORETEX project objectives mentioned on the web is the following:

- To develop generic speech recognition technologies for a wide range of tasks with minimum domain dependencies.
- To devise methods for a rapid portability to new languages with a limited amount of training data.
- To research techniques for producing enriched symbolic speech transcription for higher level symbolic processing.
- To improve language models and provide automatic pronunciation generation.
- To integrate the methods into showcases and validate them in relevant applications.
- To propose an evaluation framework and define objective measures to assess improvements.
- To disseminate the CORETEX research results and to facilitate contact with the interested users in order to widely exploit the project results.

The project is expected to provide significant insights into how to develop conceptually new HLT that is *generic, adaptable and portable***.** Generic design of technology is analyzed by evaluating a system trained on one corpus and tested on another one. Aim of this research is to assess performance degradation under the non-optimal conditions with respect to the training and testing conditions, and in the context of new languages. Initial project phase has already defined objective evaluation criteria and measures, including common test suites and the protocol.

In summary, opportunities from the CORETEX project are an improved HLT that are less sensitive to the environmental and linguistic factors as well as efficiently portable to many languages. Evaluation and demonstration frameworks were already proposed and serve to analyze the progress on the project. Detailed descriptions of the project achievements are given in the CORETEX Annual Reports for the years 2000 and 2001.

2.2. ENABLER

*European National Activities for Basic Language Resources***,** [wwwENABLER]

The ENABLER project aims to improve collaboration activities that provide national language resources in Europe. Researchers, industry, and service providers identified the LR to be a critical issue in national HLT programs and that these efforts need to be supported by appropriate national funding. The LR are of central importance to any kind of HLT-based infrastructure. They are also of vital importance in the development of HLT applications and products, thereby fundamental for the overall industrial growth. Availability of the adequate LR for as many languages as possible is of paramount importance in the HLT development for a non-exclusive multilingual information society.

This 22-month project started in November 2001. The consortium includes Università di Pisa (Italy), Institute for Language and Speech Processing - ILSP (Greece), European Language Resources Distribution Agency – ELDA (France), Center for Sprogteknologi – CST (Denmark), as well as members from Belgium, Czech Republic, Germany, Portugal, Spain, Sweden, and the Netherlands.

The ENABLER project goals are:

- To strengthen the current network of national initiatives, creating links among them, thereby providing a regular, updated, structured and public repository of organizational and technical information.
- To provide an official and general coordination forum for exchange of information, data, best practices, sharing of tools, multilateral and bilateral co-operation on specific issues.
- To gradually enlarge the existing network by identifying representatives of national initiatives.
- To promote synergies across national activities, to enhance the compatibility and interoperability of the results, thereby facilitating efficient transfer of technologies between languages.
- To maintain compatibility across various national LR.
- To increase visibility and strategic impact of the national activities.
- To provide a forum for discussion of innovative research issues and to propose medium- and long-term research priorities.
- To provide a forum to assess industry needs and to formulate common medium and long-term priorities.
- To promote exchange of tools, specifications, validation protocols produced by the national projects.
- To create an EU center for the harmonization of metadata description of speech, text, multimedia and multi-modal LR.
- To promote industrial exploitation of LR.
- To contribute to the internationally agreed cooperative framework for the provision of LR.

In perspective, ENABLER will contribute to the natural interactivity by providing multimodal LR, and to the multilinguality by fostering harmonization of national LR.

2.3. FAME

*Facilitating Agent for Multicultural Exchange***,** [wwwFAME]

New information technology tools for human-human communication integrate speech understanding, computer vision and dialog modeling and enable communication between people from different cultures who use different languages. The FAME project aims to address the problem of integrating multiple communication modes, such as vision, speech and object manipulation. Communication support is provided by the integration of physical and virtual worlds in multi-cultural communication and problem solving. The major identified project challenges are in automatic perception of human action and in understanding of free dialog between the people from different cultures.

Consortium of Universität Karlsruhe - Interactive Systems Labs (Germany), Institut National Polytechnique de Grenoble - Laboratoire GRAVIR-IMAG (France), Université Joseph Fourier - Laboratoire CLIPS (France), Istituto Trentino di Cultura - ITC-IRST (Italy), Universitat Politècnica de Catalunya (Spain), Sony International - Europe (Germany), and Applied Technologies on Language and Speech (Spain) envisions to construct an information butler that will demonstrate the context of awareness in the problem solving scenario. This goal will be achieved by integration of computer vision, speech understanding and dialog modeling. The demonstration prototype in form of an enhanced computer human-to-human communication model will be developed for the 2004 Barcelona Cultural Fair.

2.4. HOPE/EUROMAP3

*HLT Opportunity Promotion in Europe***,** [wwwHOPE]

This project aims to accelerate the rate of technology transfer from the research to the market. The project

contains 11 National Focal Points (NFPs) from Austria, Belgium/Netherlands, Bulgaria, Denmark, Finland, France, Germany, Greece, Italy, Spain and UK. The Bulgarian, French and UK partners joined the project in October 2001. Each NFP will build on skills and expertise from the previous HLT awareness-raising actions. It will strive to achieve the following objectives:

- To increase the number of projects that deliver market-ready results.
- To accelerate awareness of benefits of the HLT systems, services and applications within the user sectors, policy makers and national administrations.
- To increase the number of state-of-the-art technology developers participating in the research projects.
- To improve the relevance of project targets, technology supplier and user needs.
- To improve the match between the HLT design, supplier and end user expectations.
- To enable user partnerships for beta testing, demonstration and other market application activities.

In perspective, the project also aims to include the EU accession countries. HOPE is a 36-month project and started in February 2000.

2.5. ISLE-HLT

International Standards for Language Engineering, [wwwISLE-HLT]

The ISLE-HLT is the most recent initiative of the *Expert Advisory Group for Language Engineering Standards* (EAGLES, [wwwEAGLES]). This 36-month project started in January 2000. Consortium consists of Consorzio Pisa Ricerche (Italy), University of Southern Denmark, Institute Dalle Molle pour les Etudes Sémantiques et Cognitives (Switzerland), Center for Sprogteknologi (Denmark), University of Pennsylvania - Computer and Information Science (USA), University of Pennsylvania - Linguistic Data Consortium (USA), New York University - Computer Science Department (USA), and University of Southern California - Information Sciences Institute (USA).

The overall project aim is to develop HLT standards within a global (EU-US) international collaboration and continuing the success of EAGLES by developing, disseminating and promoting de facto standards and guidelines for the HLT language resources, tools and products. The policy of the EAGLES/ISLE is to closely interact with academia and industry, users and providers, funding bodies and research organisations. The project objectives are put on the following three areas judged to be of a long-term significance:

- *Multilingual computational lexicons*. Initial work in this area presented survey of bi- and multilingual lexicons covering publishers' dictionaries. Next, specification of the Multilingual Isle Lexical Entry (MILE) was made. This involved work on complex Italian-English word-pairs, better understanding of word sense representation and cross-language linkages, extraction and classification of sense indicators, and development of a prototype tool to manage MILE-based lexicons. The ISLE also contributed to recommendations for MILE bilingual dictionary entries. A prototype tool for management of computational lexicons conforming to ISLE recommendations was developed.
- *Natural interaction and multimodality (NIMM)*. This work extended the previous EAGLES work on textual and spoken language resources. Surveys were done on resources, annotation schemes and tools, as well as on metadata descriptions and tools. A prototype tool was developed for NIMM data annotation. XML schemas were developed that handle ISLE metadata descriptions. Editing and browsing tools were devised using these descriptions, including across distributed resources. Future work is concentrated on producing draft guidelines for best practice in the areas covered by the project, and in refining and documenting the tools and resources intended to help users in applying the guidelines.
- *Evaluation of the HLT systems*. This work focuses on methods and metrics for Machine Translation (MT). User feedback was collected within three international workshops. This led towards a refined version of the ISLE evaluation framework.

2.6. NESPOLE!

Negotiating through Spoken Language in E-commerce, [wwwNESPOLE!]

The NESPOLE! project aims to integrate speech-to-speech translation in eCommerce and eService environments by extrapolating from the results of the large research projects (C-STAR and Verbmobil). This EU project started in January 2000 and has a duration of 30 months. Consortium includes ITC - IRST, Centro per la Ricerca Scientifica e Tecnologica (Italy), Universität Karlsruhe (Germany), Carnegie Mellon University (USA), Université Joseph Fourier (France), Aethra (Italy), and Azienda per la Promozione Turistica del Trentino (Italy). It uses standard communication protocols that allow for seamless integration of the multilinguality with the existing videoconferencing software.

NESPOLE! aims to understand issues related to the ability of people communicating ideas, concepts, thoughts and to solve problems in a collaborative framework. It also includes non-verbal communication facilities in the form of multimedia presentations, shared collaborative work spaces, multimodal interactivity and manipulation of objects. These facilities allow for sharing text, graphics, audio, video, therefore providing an improved interpersonal communication. The languages addressed in the NESPOLE! project are Italian, English, German and French.

NESPOLE! identifies the following dimensions that should allow construction of the effective eCommerce and eBusiness environments

- *Robustness*: ability to cope with distractions of spontaneous speech (interruptions, corrections, repetitions, false starts).

- *Scalability*: ability to ensure an adequate level of system performance when the number of users increases.
- *Cross-domain portability*: defined as an easy and cost-effective porting of a speech-to-speech translation system to a new domain.
- *Multimedia and multimodal support*: facilitates the close integration of, and interaction between, speech-based communication and visual cues and content.

The NESPOLE! project envisions to build three different speech to-speech translation systems, including

- A system for tourism applications, embedding multimedia features.
- A system for tourism with a larger coverage of the domain, richer interaction modalities, more sophisticated multimedia support. This should demonstrate the progress on the scalability issue.
- A system for an advanced multilingual help desk. This system should highlight the results concerning the cross-domain portability.

These demonstration systems will support the multilingual negotiations between a tourist service provider and a customer aiming to organize eg, her or his holidays. Portability is addressed by porting the developed system consisting of a video help-desk for technical support, troubleshooting and repair to a different domain.

2.7. ORIENTEL

Multilingual access to interactive communication services for the Mediterranean and the Middle East, [wwwORIENTEL]

The ORIENTEL project explores potential of the multilingual communication services for Mediterranean and the Middle East. Emphasis is put on the mobile applications that are on rise globally. Neither resources nor sufficient expertise are currently available to cope with the linguistic research challenges of the area and the problems posed for Automatic Speech Recognition technology.

The project started in June 2001 with the consortium of Philips Speech Processing (Germany), European Language Resources Distribution Agency (France), IBM Deutschland (Germany), Knowledge (Greece), Natural Speech Communication (Israel), Siemens (Germany), Universitat Politecnica de Catalunya (Spain), and Lucent Technologies Network Systems (UK).

Main objectives of the ORIENTEL project are to:

- Outline survey analysis of markets, technologies, languages and users of mobile communication.
- Gain fundamental knowledge about linguistic structure of the target languages.
- Develop strategies and standards for phonetic and orthographic transcriptions.
- Collect 23 speech databases to support mobile communication applications.
- Research for language, dialect and foreign accent adaptation techniques.
- Develop demonstrator applications.

The project outcome will therefore significantly contribute to the spoken language resources distributed by the ELRA/ELDA.

3. US DARPA Projects

The US DARPA supports a large pool of projects under the Translingual Information Detection, Extraction and Summarization (TIDES) umbrella [wwwTIDES]. These research projects also address the core issues in portability of HLT mentioned above.

4. Conclusions

This paper reflected some of the on-going research and development efforts towards the challenges and opportunities in portability of HLT. It advocates for the view that *every* language of the world contributes to the cultural richness of the information society. This vision should also be applied when the HLT support need to be developed for a small-market or non-prevalent languages [Ostler, 1999]. Furthermore, research in portability issues of HLT for prevalent languages has recently shown that a system developed with a bigger set of languages may exhibit better performance than a system trained with a large set of the target language task-specific data.

Since *every* language is constantly changing while adapting to the influences brought by globalization and increased human mobility, it is reasonably to expect that the state-of-the-art performance in HLT could only be achieved when the HLT development phase included a grand pool of languages instead of only a particular one. Additionally, robust HLT needs to be adaptive to the user and the task involved.

In conclusion, research in portability issues of HLT should be encouraged and strengthened. This could be achieved by forming a network of excellence on HLT portability under the forthcoming 6th EU framework program. Since HLT portability is a very important, difficult and challenging research problem, such NoE should include all interested major players in the field, as many national HLT entities as possible, as well as researchers concerned with the non-prevalent languages (eg., the ISCA SALTMIL SIG) [wwwSALTMIL].

5. References

Bird, S. (2001). Annotation graphs in theory and practice, http://media.nis.sdu.dk/elsnet/annotationgraphs/ . *9th ELSNET European Summer School on Language and Speech Communication, Text and Speech Corpora.* Lectures are accessible at http://media.nis.sdu.dk/elsnet/

Lamel, L. (2002). Some Issues in Speech Recognizer Portability. *This Proceedings*, pp. 14-22.

Ostler, N (1999). Does Size Matter? Language Technology and the Smaller Language. *ELRA Newsletter, 4(2): pp. 3-5.* Paris. (ISSN 1026-8200)

URL list accessed in April 2002:

[wwwCORETEX] http://coretex.itc.it/

[wwwEAGLES] http://www.ilc.pi.cnr.it/EAGLES/ home.html

[wwwEANBLER] http://www.HLTCentral.org/projects/ENABLER/

[wwwFAME] http://www.HLTCentral.org/projects/ FAME/

[wwwHLTCentral] http://www.hltcentral.org/

[wwwHOPE] http://www.HLTCentral.org/projects/HOPE/

[wwwISLE] http://www.HLTCentral.org/projects/ISLE-HLT/

[wwwNESPOLE] http://nespole.itc.it/

[wwwORIENTEL] http://www.orientel.org/

[wwwSALTMIL] http://isl.ntftex.uni-lj.si/SALTMIL/

[wwwTIDES] http://www.darpa.mil/ipto/research/tides/

The Atlantis Observatory: Resources Available on the Internet to Serve Speakers and Learners of Minority Languages

Salvador Climent[*]
Miquel Strubell[*]
Marta Torres[*]
Glyn Williams[**]

*Universitat Oberta de Catalunya (UOC) / Internet Interdisciplinary Institute (IN3)
scliment@uoc.edu / mstrubell@uoc.edu / mtorresv@uoc.edu

**Foundation for European Research, Wales
g.williams@bangor.ac.uk

Abstract

The ATLANTIS Project (Academic Training, Languages and New Technologies in the Information Society) and its outcome, The Atlantis Observatory, are presented. The project's website (www.uoc.edu/in3/atlantis) brings together totally updated information on digital tools and resources available for Lesser-Used Languages of the European Union in a searchable database. The structure and classification of the database is explained and some preliminary results are also offered.

1. Introduction

Globalisation and the development and spread of digital technology in the Information Society provide excellent opportunities for creating spaces and tools for the use of many smaller languages. But the degrees of enterprise and know-how on which to draw from within the linguistic community vary, largely as a function of the size of that community. So, unless special support is given to such communities, there is a real danger that networks will develop only in larger languages, and particularly the hegemonic languages in the respective States, in rapidly growing areas such as the Internet. Thus the smaller linguistic communities, and especially those whose language is not that of the State, need to have at their disposal both products that can satisfy new demands, and platforms which will allow them to share initiatives with partners whose languages face a similar challenge.

In this framework, The ATLANTIS Project was aimed to create a virtual network that facilitates regular contact among individuals from all European Union lesser-used languages (LUL) to share knowledge on digital tools and resources available for such linguistic communities.

In the following section, the background and the main goals of the project are presented; then in section 3 we acknowledge the languages that are the subject of study. Sections 4 and 5 are devoted to describing the main areas to be analysed and their structuring and presentation in the database. Section 6 presents some preliminary results and reports. The paper ends with some concluding remarks.

2. The ATLANTIS Project. Baseline and objectives

The ATLANTIS Project, *Academic Training, Languages and New Technologies in the Information Society*, (funded by the EU under the terms of contract nº 2001 – 0265 / 001 – 001 EDU – MLCME) has been carried conjunctly by the Internet Interdisciplinary Institute (IN3) of the Universitat Oberta de Catalunya (Open University of Catalonia, UOC), the Foundation for the European Research University of Wales and the Onderzoeks Centruum voor Meertaligheid (Multilingualism Research Centre) of the Katholieke Universiteit Brussel (Dutch Language Catholic University of Brussels).

It leads on naturally from the Euromosaic report (Euromosaic, 1996), a study of the minority language groups of the European Union (EU) in order to ascertain their current situation by reference to their potential for production and reproduction, and the difficulties which they encounter in doing so. The Euromosaic report highlighted the shift in thinking about the value of diversity for economic deployment and European integration. It argued that language is a central component of diversity, and that if diversity is the cornerstone of innovative development, then attention must be given to sustaining the existing pool of diversity within the EU.

Now focusing on one of the various social and institutional aspects whereby a language group produces and reproduces itself –digital technology in the IS–, The ATLANTIS Project was designed to accomplish the following main objectives:

a. Bring together totally updated information on digital tools and resources available for Lesser-Used Languages.

b. Place the results on a new website –The Atlantis Observatory: www.uoc.edu/in3/atlantis/ – that will consist of a searchable database of the resources detected and thus duly classified.

c. Draw up a final report that will underline areas, projects and technology which, in the view of the participants, offer greatest potential for multiplying effects from one language group to another.

It must be noticed that these aims go along to a great extent with the general aims of the SALTMIL SIG (Special Interest Group on Speech and language Technology for Minority Languages) –promotion of research, development and education in the area of Human Language Technologies for less prevalent languages. Nadeu et al. (2001) specifically point that *"the vision of the SALTMIL SIG is that sharing of information and the forming of a network of researchers is important to begin with. It is hoped that this networking will form the seed-bed out of which more substantial projects will grow"*.

3. Languages targeted

The languages included in this study are all the autochthonous languages in the European Union which are not one of the eleven official EU languages –therefore, those minority languages which are EU official on account of being the official language in a neighbouring State are not included. In a few cases (such as Albanian or Slovene), though the language is official in a neighbouring State, it has been included because that State has not yet joined the enlarged Union.

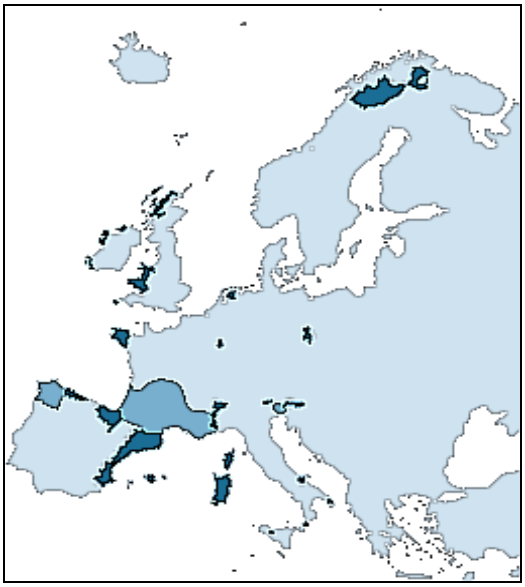


Fig. 1: Location of the languages targeted by the project

Therefore, languages targeted are the following (see Fig. 1):

1. Albanian (as spoken in Italy)
2. Asturian (Spain)
3. Basque (Spain, France)
4. Breton (France)
5. Catalan (Spain, France, Italy)
6. Cornish (UK)
7. Corsican (France)
8. Franco-provençal (Italy)
9. Frisian (Netherlands)
10. Friulian (Italy)
11. Gaelic (UK)
12. Galician (Spain)
13. Irish (UK, Ireland)
14. Ladin (Italy)
15. Luxembourgish (Luxembourg)
16. Occitan (France)
17. Sami (Finland, Sweden)
18. Sardinian (Italy)
19. Slovene (Austria, Italy)
20. Sorbian (Germany)
21. Welsh (UK)

4. Work package categories

Information from all EU LUL has been gathered in six parallel work packages:

1. Learning Platforms in LUL
2. Human Language Technology Developments
3. Information and Communication Technology: Regional Plans, computer software and Internet tools
4. Cultural Digital Resources and Linguistic Diversity
5. Convergence and LUL Broadcasting
6. Electronic Publishing and LUL

In order to do that, each partner took charge of a group of linguistic communities and distributed a comprehensive questionnaire to as many researchers, professionals and academic specialists they could contact. Those informants were also requested to circulate the questionnaire among other specialists in their fields. For those few linguistic communities where feedback resulted to be scarce, the partner in charge committed itself to gather information.

Each of the six work package categories is now described in more detail.

4.1. Learning Platforms in Lesser-used Languages

On-line learning offers cost-saving contexts for small dispersed populations and can thus be of considerable value for numerous language groups. In this section, information has been gathered on the extent to which LUL groups are incorporated into on-line learning platforms being developed in each of the European regions studied which have a LUL group. All levels of educational delivery have been studied, as well as the various associated training programmes. The information on the selected sites and products will allow potential users to see

how knowledge resources are being made available in the LUL.

4.2. Human Language Technology Developments

Human language technology for lesser-used languages is the basis for much further development. The goals of e-mail, web page translation or discussion group translation require the appropriate technology for the language pairs that involve the LUL and the state language. Before this is possible, however, the basic requirements of such development have to be available: electronic corpora, dictionaries, spell checkers, grammars etc. These developments are expected to focus in on-line learning, administration and electronic publishing.

4.3. Information and Communication Technology (ICT): Regional Plans, computer software and Internet tools

Information and Communication Technologies are advancing fast. The extent to which Regional Authorities are addressing the issue, the importance they attach to the availability of tools in the relevant language, and the range of existing computer software and internet tools in each language, are the subjects of the category.

4.4. Cultural Digital Resources and Linguistic Diversity

The development, storage and accessing of digital resources in the context of the emerging Digital Economy requires the creation of Media Asset Management Systems. The extent to which this is proceeding within each region is an object of study. The development of appropriate resource locators allow such materials to be available not merely for industrial development based on the New Media sector, but also for on-line learning developments which, increasingly, will rely on digital resources. The EU's e-Content initiative is highly relevant to these developments.

4.5. Digital Convergence and Lesser-used Language Broadcasting

Many lesser-used language groups have their own audiovisual broadcasting media. The transition from solely analogue broadcasting, to the inclusion of digital systems, which a limited number of minority language communities have already embarked upon, opens up the potential of convergence. More and more audiovisual products are being made, and even shot, in a digital format. This is relevant to some learning developments and user-friendly platforms that encourage interactivity and can increase the potential of digital democracy.

4.6. Electronic Publishing and Lesser-used Languages

Electronic publishing in most LUL is already underway, if only, as happens in some cases, only through LUL web sites. The scope for low cost newspaper and journal publication has greatly expanded thanks to the web. Data has been gathered about the progress of such developments for all the language groups.

5. Structure of the database interface

The database has been organized according to the categories described above and several corresponding subcategories in a way that users can perform searches by language, by (sub)category or by any possible cross grouping of languages and/or (sub)categories.

The first category, *Learning Platforms in LUL*, is arranged for products and resources around two main axis: level (primary, secondary, tertiary and adult education) and area (language, science, mathematics and arts & social science). Moreover, users can search for online educational projects organised in two categories: (i) for learning and information purposes, and (ii) leisure oriented (games, etc.).

Due to its complexity, the *Human Language Technologies* (HLT) package is the one that has undergone a richer and stricter organisation. It has been tailored according to Sarasola (2000) levels and categories, which acknowledge the phases a minority language should follow to incrementally develop its HLT capabilities.

Sarasola's five phases have been simplified within ATLANTIS to the following three: (i) Foundations, (ii) Tools and Resources for Application Development, and (iii) Advanced Tools and Applications. Each one of such level-categories is divided in several field subcategories – such as Lexicon, Speech, Corpus, etc. These, at their turn, subdivide in types of tools, resources or applications – such as Database, Parser, Integrated System and the so.

Foundations is detached in three subcategories: Corpus (raw text), Lexicon and Morphology (raw lists, description of phenomena, different kinds of machine-readable dictionaries) and Speech (collections of recordings, descriptions).

The *Tools and Resources* category is in turn organised around five standard levels (Corpus, Lexicon and Morphology, Syntax, Semantics and Speech) each one including several tool subcategories (such as different kinds of parsers and knowledge bases), plus an Integration of Tools and Resources level.

Last, in *Advanced Tools and Applications* the following subcategories apply: Authoring Aids (spell, grammar and style checkers), Translation (Machine Translation and integrated Computer Assisted Translation environments), Information Retrieval and Extraction systems and advanced tools, Speech (synthesis, recognition, dialog systems) and Language Learning environments.

The third main category of the database, *Information and Communication Technology–Regional Plans, computer software and Internet tools*, is searchable by two subcategories: Regional Plans, and Software and Internet Tools.

The fourth, *Cultural Digital Resources and Linguistic Diversity*, is organised around seven kinds of media or resources: TV stations, Radio stations, Libraries, Museums, Music, Voice recordings and Other.

Last, the two remaining main categories, *Convergence and Broadcasting* (in fact, Radio or TV digitised) and *Electronic Publishing*, are not subdivided.

Every search in the database returns as output the list of matching items with the following information:

- Name of the product, a link to the URL of the product, name of the organization which has developed or is the owner of the tool, resource or application.

- A record with basic information about the tool, resource or application and the set of ATLANTIS categories under which it has been classified.

6. Preliminary results and reports.

At the moment we are writing this paper most of the data are still being gathered, studied and classified in order to produce six final per-category reports and the final overall report of the project. Nevertheless, we can already offer preliminary summary reports for the following languages: Breton, Friulian, Irish, Scots Gaelic, Slovene and Welsh (**§6.1** below); and Asturian, Basque, Catalan, Corsican, Galician, Occitan and Sardinian (**§6.2**). Such groupings simply correspond to work packages as distributed to the Atlantis Project research centres.

With respect to data figures, we can only show now as being reliable the total number of entries for the languages of the second group.

6.1. Breton, Friulian, Irish, Scots Gaelic, Slovene, and Welsh

For such languages, one can state the points that are detailed below.

6.1.1. Learning Platforms in Lesser-used Languages

All states are developing connectivity and establishing ICT (Information & Communication Technology) as a basis for its educational system. Those states that do acknowledge the relevance of minority languages for learning do not necessarily develop the tools and materials required for this to operate. However, this does not guarantee development. In Italy, the frontier agreement with Slovenia means that many of the developments for the Slovene language group await developments in Slovenia. In Austria on the other hand, the same language group does have the advantage of a concerted effort to develop supporting materials for the limited amount of teaching in Slovene. The main problem here is the tendency to interpret the legal requirement liberally, which means that the service is not very effective. In Scotland, connectivity is available but the developing of materials and the use of ICT is left to each individual learning enterprise and there is little central support. The situation is similar in Ireland by reference to Irish. In Brittany on the other hand the state makes virtually no provision for Breton medium education and therefore the limited amount of on-line learning that is available is the consequence of private initiative. The best situation appears to be in Wales where institutions responsible for developing on-line learning in Welsh match connectivity. This supported by the fact that the local authorities as learning providers are obliged to have their language plans confirmed by the Welsh Language Board. Also, the National Assembly for Wales, which has the sole responsibility for education in Wales, is devoted to developing a bilingual nation. Friulian lacks any support of this nature.

6.1.2. Human Language Technology Developments

Again, the situation is highly variable. In Wales, there have been certain developments but these have yet to developing machine translation and voice recognition capacity even with Welsh/English language pair. This is partly because the issue is driven by the translation agenda, which has become a powerful lobby rather than by economic needs. In both Austria and Italy the developments depends entirely on Slovenia, which is one of the few states in Europe that has not developed full capacity. In Ireland the picture is broadly similar to that in Wales whereas in Scots Gaelic has a limited presence even though dictionaries, corpora and grammars have been developed. In Brittany much of the initiative is the result of private efforts and is limited to on-line dictionaries, grammar checkers, etc. It is clear that this area requires considerable investment, usually by private commercial enterprises. Friulian also lacks any development other than limited private initiatives.

6.1.3. Information and Communication Technology: Regional Plans, computer software and Internet tools

Not all regions have such plans. Thus in Ireland there is little such coherent development even though the new initiatives in the West are developing plans which, between them, can be said to constitute regional technology plans. However, things are in their infancy and the failure of large companies to extend broadband to these areas is holding things back. Little is happening by reference to language in these areas but the awareness of the need to do so is high. In Scotland, such plans are in the hands of the Scottish Parliament and the Highlands & Islands Enterprise. The latter has responsibility for Gaelic but its plans make little reference to ICT and Gaelic. In Friulian and the Slovene border areas regional development is limited to European Regional Development Fund initiatives and there is little reference to language in such plans. The same can be said of Carinthia (in Austria) where the plans which are developed are relatively sophisticated but have little of relevance for the Slovene language group. Wales was one of the first to develop a Regional Technology Plan under the RISI programme of the EU. This has been superseded

by the Cymru ar Lein' initiative. While there is a strong awareness of the need to incorporate Welsh development awaits the ability to incorporate the language into economic development writ large. In Brittany, the technological features of regional development make no reference to Breton.

6.1.4. Cultural Digital Resources and Linguistic Diversity

We must realise that we are in the beginning of any development of the Digital Value Chain. Thus far, it is unlikely that there is a regional DVC anywhere. Nonetheless, there are early developments. The Cymru'n Creu project in Wales is developing at least one end of the DVC. The exploitation end is emerging but is not articulated with the content end. The content end requires considerable investment whereas the production end does not. It is likely that Ireland will eventually develop one but is moving slowly in this direction at present. Scotland is in the same situation as Wales with SCRAN being an important innovative venture. SCRAN (Scottish Cultural Resources Access Network) was set up by museums, libraries and archives to create multimedia, manage digital IPR and provide educational access.

It is less likely that the other regions will be DVC regions and may well emerge as either content regions or production regions, more likely the later. This is largely because regional resources are housed in the capital region of the state so that initiatives will derive from that location on a state-wide basis. This does not preclude the emergence of regional eContent economies but it is less likely than in the historic regions with strong political autonomy. Carinthia is digitising some resources and there are multimedia companies capable of exploiting these but it is very limited. It is even less so among the Slovenes in Italy and also in Friulian. Brittany is in a similar situation. Whether the DVC regions focus on minority language digital resources depends on two things:

i. The extent to which they appreciate that diversity is driver of the Digital Economy and markets will be structured by language and not by states.

ii. The specific drive to incorporate minority languages into the New Economy.

6.1.5 Convergence and Lesser-used Language Broadcasting

This is also a matter of regional and central policy. The two language groups in north-eastern Italy will be hampered by the limited amount of exposure to media for the languages and the centralized nature of the broadcasting framework. However as costs plummet and deregulation takes hold, it will be possible to develop private initiatives. Carinthia also has a limited regional broadcasting presence and less so for Slovene. The Slovene language groups will, in all likelihood, benefit from deregulation and the entry of Slovenia into the EU, which will create a more integrated digital broadcasting region. Ireland is hampered by the size of its population and the dependence on terrestrial cabling which tends to be expensive. The main providers have recently pulled out and the state system is being partly privatised. Thus, development is hindered. Its minority language service will involve transformation of existing analogue services. Brittany has started developing a strong regional broadcasting capacity in the minority language and this will benefit from digitisation and the opportunities afforded by convergence.

6.1.6. Electronic Publishing and Lesser-used Languages

Electronic publishing is easier to conceive of partly because orthodox publishing in the minority language already exists and partly because of the relatively low cost. In all likelihood, this will be a parallel venture involving both orthodox publishing and electronic publishing existing side by side. The interesting developments involve exploiting convergence. This is already happening in Wales using Welsh where the main newspaper and the BBC are cooperating and also by reference to the community newspapers which are linked to the BBC's web service for the Welsh diaspora. As costs fall regional broadcasting and publishing will converge and will become far more localized. The publishing houses in Carinthia are also developing electronic Slovene language services. The Slovene newspaper in Italy is also available on-line but further developments are limited. Friulian has a limited development, as does the Gaelic language group in Scotland. Ireland's developments are also in a rudimentary state.

6.2. Asturian, Basque, Catalan, Corsican, Galician, Occitan and Sardinian.

For such languages we have collected and processed the following number of entries:

Asturian	50
Basque	408
Catalan	400
Corsican	50
Galician	225
Occitan	100
Sardinian	50

Some preliminary conclusions for this group are detailed below.

6.2.1 Learning Platforms in Lesser-used Languages

In this field we find a number of resources for the teaching/learning of languages on-line at different levels and for different target groups. Some are multilingual in nature, and a number are simple websites for adults, such as World Language Resources (which caters for Basque, Catalan, Galician, Sardinian, Corsican), Tandem Agency, etc. Monolingual language courses, grammars and lexicons are often offered by private individuals keen on

disseminating their language on the net. For instance, an on-line Occitan course at

http://occitanet.free.fr/cors/intro.htm. Institutional support for developing and/or disseminating language educational products can be observed in most of these languages, such as *A Palabra Herdada. Curso de Galego*, promoted by the Dirección Xeral de Política Lingüística of Galicia.

Other educational projects are not for teaching the language, but rather use it as the medium of instruction. "*Recursos educativos para ciencias naturais*" in Galician (http://www.galego21.org/ciberlingua/recur.htm) is a good example of well-sorted links to available resources of this nature, but there are not many. Another is aimed at primary school education: *CD ikastola.net*. Nearly all material aimed at primary education is for Catalan or Basque: nothing has been detected in Occitan, Sardinian or Corsican, and a few tools are in Galician or Asturian. The same can also be said about secondary education, though in this case more Galician products have been found. At university level most Catalan universities offer on-line language courses both for non-native learners and for native speakers improving their literacy skills. Other tertiary level sites offer information on literature, e.g. Biblioteca d'Autores Asturianos at http://www.araz.net/escritores/ or philosophy resources in Galician, http://filosofia.00go.com/.

These are nearly all single products unrelated to digital educational platforms as such. Others cover leisure products which range from digital games such as *Trivial Pursuit euskaraz eta on line!* in Basque, at http://www.argia.com/tribiala.htm, to distribution lists and newsgroups in Occitan (soc.culture.Occitan, or the forum at http://www.oest-gasconha.com/listadif.php3).

There are, however a number of digital learning platforms. These are to be found in the virtual campuses of many universities such as the Universitat Oberta de Catalunya, which offers a range of degree courses, both undergraduate and postgraduate, in Catalan (http://www.uoc.edu/). The universities involved are virtually all Catalan (including, of course, Valencian universities) or Basque.

6.2.2. Human Language Technology Developments

Much of the digital work in the "Foundations" section has been done on Basque, by a wide range of organisations, many of which publicly supported. But all the languages studied do have at least some work in this area. Projects include untagged corpora, speech recordings and mono- or bi-lingual dictionaries. One example of an oral archive is the *Archivu Oral de la Llingua Asturiana*, http://www.asturies.org/asturianu/archoral/. In the "Tools and Resources for Application" section, Basque, Catalan and Galician seem to be the most productive. As regards taggers and tagged corpora, most of the work appears to have been done on Catalan and, to a lesser extent, Galician. Lexical and speech databases can be found in and for Catalan, Basque, Galician and for Corsican: www.ac-corse.fr/expos_autres/webdlc2/webdlc/Acceuil.html. No such developments have been found for Asturian, Occitan or Sardinian. Several terminology research centres offer resources on the Internet including Termcat (for Catalan) and UZEI (for Basque). In the field of Lexical-Semantic knowledge bases, WordNets have been developed for Catalan and Basque, and a Galician version is being developed.

Moving now to "Advanced tools and applications" we have found Authoring aids (spelling correctors and, just for Catalan, a grammar and style checker) for the following languages: Catalan, Galician and Basque. An Asturian product is about to be launched. Bilingual machine-translation systems have been developed for Basque, Galician and Catalan, such as the Basque-Spanish tool developed by the Basque government: http://www1.euskadi.net/hizt_3000/. Speech tools include those developed by Telefónica for speech recognition and synthesis for Catalan, Basque and Galician (alongside Spanish). Philips has developed the continuous-speech recognition tool *Free Speech* for Catalan only. No other languages in our group seem to have similar tools. Linguistic information retrieval and extraction tools have been located for Catalan and Galician. Web crawlers have been developed to manage Basque, Catalan and Galician.

6.2.3 Information and Communication Technology: Regional Plans, computer software and Internet tools

Regional ICT plans of greatly varying scope and objectives have been found at least for the following: Catalan (Catalonia, Balearic Islands), Basque and Sardinia. They also vary in the importance they attach to language in the plan. The Regional Development Plan (http://dursi.gencat.es/ca/de/pla_estrategic.htm) developed by the government of Catalonia does include specific projects related to the Catalan language.

As to software developed for the internet, languages such as Basque, Catalan and Galician have developed versions of the most widely used tools, such as several operating systems, the music file manager WinAmp (available also in Asturian) and web crawlers such as Netscape. Softcatalà, Softkat (for Basque) and Proxecto Xis-Galego 21 are three organizations devoted primarily to this work, as well as to developing new software. Very little has been found for Corsican, Occitan and Sardinian.

6.2.4. Cultural Digital Resources and Linguistic Diversity

In this section the digital treatment of libraries is interesting. Several cases have national libraries with limited digital services. All languages studied have at least websites reproducing literary and/or academic texts. Some libraries are fully digital: *Biblioteca Joan-Lluís Vives* is a good example of a resource containing digital versions of important Catalan literary texts. *InterRomania* has literary texts in Sardinian, Catalan and Corsican.

The small subsection on Museums is devoted strictly to those which offer digital resources related to the area, or from its own stocks, on the Internet.

The "voice" subsection contains a heterogeneous collection of resources relating to Basque, Galician, Sardinian and Corsican, from recited toponyms to full-scale digital archives.

The musical resources are plentiful. Significantly, each language studied has at least one website offering such recordings, with MP3, given the ease with which digitisation is possible. Thus the range of resources is enormous: from versions of original recordings on record. Some are sung, others are instrumental, and they range from traditional music to rock.

The section also includes a wide variety of other digital resources related to each culture: from photography to the visual arts, cartoons, catalogues of films made and/or dubbed (in Catalan). A high quality Galician multimedia resource centre is housed at http://www.culturagalega.org

6.2.5. Convergence and Lesser-used Language Broadcasting

The situation for radio and for television is somewhat different. Radio stations are available on the Internet in most of the languages, except Asturian. Al least 11 Catalan radio stations broadcast (live or stored) on the Internet, as do several of Basque stations. As regards television, digital satellite television is available in Catalan, Basque and Galician, whereas the picture for Internet TV is different: Catalan is not available, whereas Galician and Basque, and even Occitan, broadcasts are available. France 3 Corse has uploaded some of its programmes, a few of which (local news programmes) are in Corsican: http://www.france3.fr/semiStatic/382-1250-NIL-NIL.html. France 3 promises the same for Occitan and Catalan.

6.2.6 Electronic Publishing and Lesser-used Languages

The section is very rich in quantity and variety. The "Academia de la Llingua Asturiana" has fully four digital journals of a cultural and literary nature. Indeed, every language has similar journals. There are also other journals of a non-cultural nature, such as the Basque cooperative movement Eroski's journal *Consumer*, which is published in Galician, Catalan (http://revista.consumer.es/web/ca/) and Basque as well as Spanish. http://codigocero.com/ is a Galician journal designed as a portal which offers information and news about new technologies.

Other linguistic products which have been regarded as electronic publications include multimedia encyclopædias, dictionaries (including interesting combinations such as Occitan-Basque), vocabularies and grammars. A CD-Rom on lesser-used languages, Lingua+, can also be viewed via the Internet. Publishers and/or sellers of electronic books include Basque houses, and the Catalan http://www.llibres.com, most of whose sales are still printed books. Electronic short stories in Corsican are sold through http://www.ac-corse.fr/fole2/fole.htm, where a demo can be viewed.

In dealing with daily newspapers a distinction has to be made between printed newspapers which also have an electronic edition, and strictly electronic dailies. In both cases a considerable investment is needed. Among the former there are many examples. Including http://www.egunkaria.com/ in Basque, and http://www.avui.com and http://www.diaridebalears.com/, among others, in Catalan. Among the latter we find Vieiros-Hoxe http://www.vieiros.com/ in Galician, and http://www.diaridebarcelona.com, which is run by the Barcelona city council, in Catalan.

Several languages have regular news services. Good examples are http://www.vilaweb.com, which operates in Catalan and uses mostly links with other electronic dailies, http://www.asturies.com/ in Asturian.

7. Concluding remarks

The Observatory has concluded its development phase, and has fulfilled one of its main aims: to bring together into its database a wide variety of initiatives relating to the new digital age. It is to be hoped that many of these will spur others into similar initiatives, hopefully working synergistically: as we stated at the outset, it is hoped that it will facilitate "regular contact among individuals from all European Union lesser-used languages to share knowledge on digital tools and resources available for such linguistic communities".

What remains to be done, as we write this paper, is to determine to what extent each of these communities is well placed to enter the Digital Economy. Perhaps it is too ambitious, or pretentious, to imagine that we can pinpoint, for each community, which obstacles may appear in its drive towards the Digital Economy, as lack of basic tools, weak levels of networking, etc. Were this to be feasible, the Observatory could act as a useful reference point for planners.
At a less ambitious level, we are confident that users of the database will point out the inaccuracies and help us to continuously update the information it contains.

8. References

Euromosaic (1996) Euromosaic: The production and reproduction of the minority language groups in the European Union. www.uoc.edu/euromosaic.

Nadeu C., D. Ó Cróinín, B. Petek, K. Sarasola, B. Williams (2001) ISCA SALTMIL SIG: Speech and Language Technology for Minority Languages. In *Proceedings of EUROSPEECH 2001*. Alborg, Denmark

Sarasola K. (2000). Strategic priorities for the development of language technology in minority languages. In *Proceedings of Workshop on "Developing language resources for minority languages: re-usability and strategic priorities"*. Second International Conference on Language Resources and Evaluation (LREC 2000) Athens, Greece

Towards the definition of a basic toolkit for HLT

Agirre E., Aldezabal I., Alegria I., Arregi X., Arriola J.M., Artola X., Díaz de Ilarraza A., Ezeiza N., Gojenola K., Sarasola K., Soroa A

IXA Group
Dept. of Computer Languages and Systems
University of the Basque Country, 649 P. K.,
E-20080 Donostia, Basque Country

KSarasola@si.ehu.es

Abstract

This paper intends to be an initial proposal to promote research and development in language independent tools. The definition of a basic HLT toolkit is vital to allow the development of lesser-used languages. Which kind of public HLT products could be integrated, at the moment, in a basic toolkit portable for any language? We try to answer this question by examining the fifty items registered in the Natural Language Software Registry as language independent tools. We propose a toolkit having standard representation of data and develop a strategy for the integration, in a common framework, of the NLP tools.

1. Introduction

SALTMIL, the ISCA SIG (International Speech Communication Association Special Interest Group) on Speech and Language Technology for Minority Languages, has the overall aim of promoting research and development in the field of speech and language technology for lesser-used languages. Actually, its main activity is providing a channel of communication between researchers by means of workshops and the discussion list. The members of SALTMIL, we often wonder how to promote research and development in a more active way. In this paper we would like to propose a medium term project to accomplish that goal: the definition of a basic toolkit for HLT. Of course, this toolkit should be designed following the basic principles of reusability and portability[1]. So, the adoption of common standards and procedures will help to minimise costs and workload in research. This way will be beneficial for any kind of language (and vital for lesser-used languages), and would define a new collaboration-space for researchers working with different languages.

The real challenge is, however, how to define a basic toolkit for HLT? In this paper we will not resolve this problem, but we want to lay some foundations to address it. First, we will try to collect an initial list of present tools and applications that are portable (usable) for different languages:

- How many of the present HLT tools and applications are portable?
- How many of them are free for academic and public uses?
- Is there any tool for any of main basic applications? or… Is there any application with no accessible tool?

In this way, by recognizing which are the most basic tools, we propose four phases as a general strategy to follow in the processing of any language. Therefore, tools considered in the first phase will be taken as more basic than the later ones.

The paper is organized as follows: Section 2 proposes a strategy to develop language technology for language, grouping linguistic resources, tools and applications in four different phases. Section 3 examines the programs registered by the Natural Language Software Registry (NLSR) in order to determine the present proportion between portable and not-portable HLT products. Section 4 proposes a standard representation of linguistic data; it is a method we use in IXA Group in order to allow the integration between different tools in the same HLT framework; the standard representation would be fundamental for any possible basic toolkit. Finally, some concluding remarks are included.

2. Recognizing basic tools and their preference

We present here an open proposal for making progress in Human Language Technology. This proposal is based on the fifteen years experience of the IXA Group with the automatic processing of Basque. Anyway, the steps here proposed do not correspond exactly with those observed in the history of the processing of English, it is due to the high capacity and computational power of present computers allows arranging problems in a different way. We must remark that our work has been centered on the processing of written language and that we do not have any reliable experience on spoken language. However, in this proposal some general steps on speech technology have included.

Language foundations and research are essential to create any tool or application; but in the same way tools and applications will be very helpful in research and improving language foundations. Therefore, these three levels (language foundations, tools and applications) have

[1] Main themes chosen for the last two ISCA SALTMIL SIG workshops were "*Re-usability and strategic priorities*" (Athens 2000) and "*Portability Issues in Human Language Technologies*" (Gran Canaria 2002).

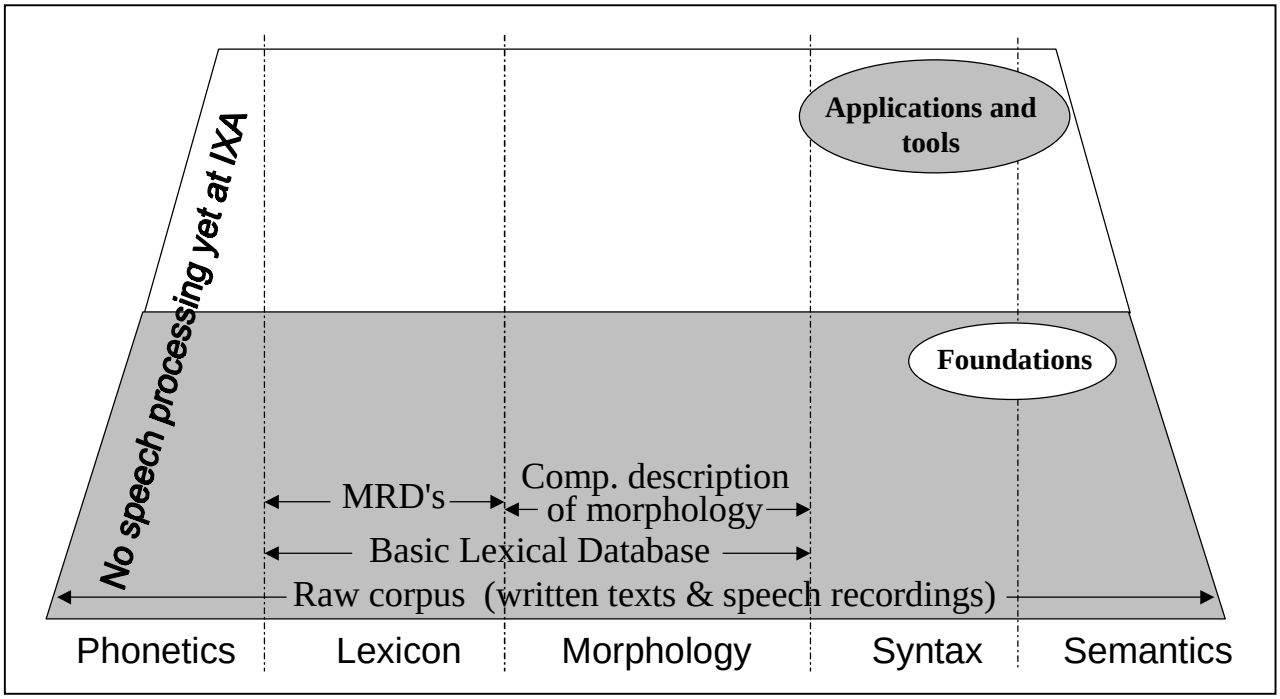


Figure 1. First phase: Foundations.

to be incrementally developed in a parallel and coordinated way in order to get the best benefit from them. Taking this into account, we propose four phases as a general strategy to follow in the processing of the language.

Initial phase: Foundations (see Figure 1).

- Corpus I. Collection of raw text without any tagging mark.
- Lexical database I. The first version could be simply a list of lemmas and affixes.
- Machine-readable dictionaries.
- Morphological description.
- Speech corpus I.
- Description of phonemes.

Second phase: Basic tools and applications.

- Statistical tools for the treatment of corpus.
- Morphological analyzer/generator.
- Lemmatizer/tagger.
- Spelling checker and corrector (although in morphologically simple languages a word list could be enough).
- Speech processing at word level.
- Corpus II. Word-forms are tagged with their part of speech and lemma.

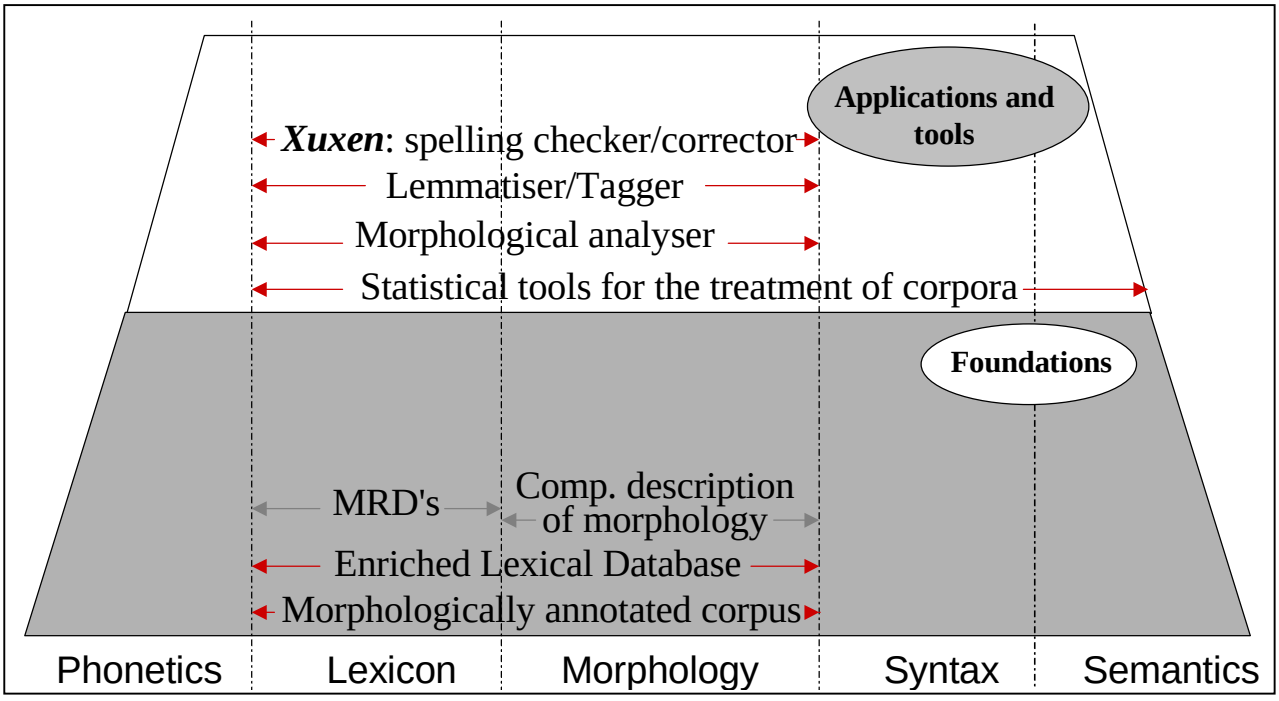


Figure 2. Second phase: Basic tools and application.

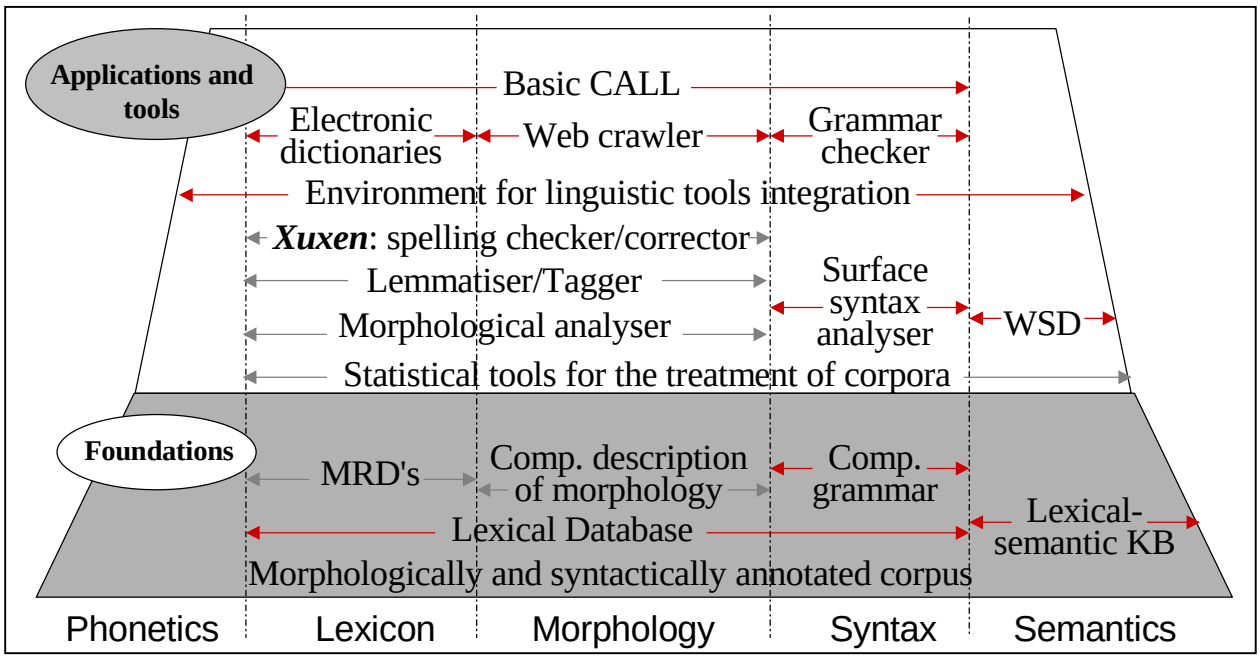


Figure 3. Third phase: advanced tools and applications.

- Lexical database II. Lexical support for the construction of general applications, including part of speech and morphological information.

Third phase: Advanced tools and applications.

- An environment for tool integration. For example, following the lines defined by TEI using XML. Section 4 describes this proposal.
- Web crawler. A traditional search machine that integrates lemmatization and language identification.
- Surface syntax.
- Corpus III. Syntactically tagged text.
- Grammar and style checkers.
- Structured versions of dictionaries. They allow enhanced functionality not available for printed or raw electronic versions.
- Lexical database III. The previous version is enriched with multiword lexical units.
- Integration of dictionaries in text editors.
- Lexical-semantic knowledge base. Creation of a concept taxonomy (e.g.: Wordnet).
- Word-sense disambiguation.
- Speech processing at sentence level.
- Basic Computer Aided Language Learning (CALL) systems.

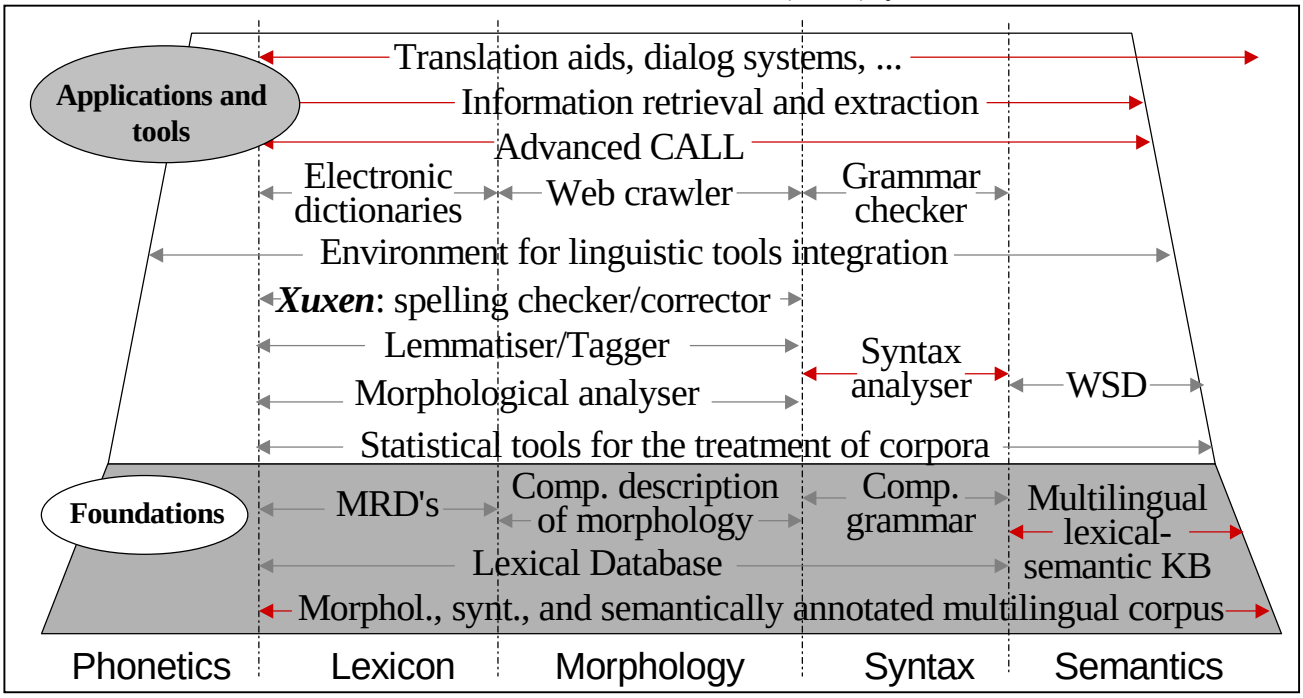
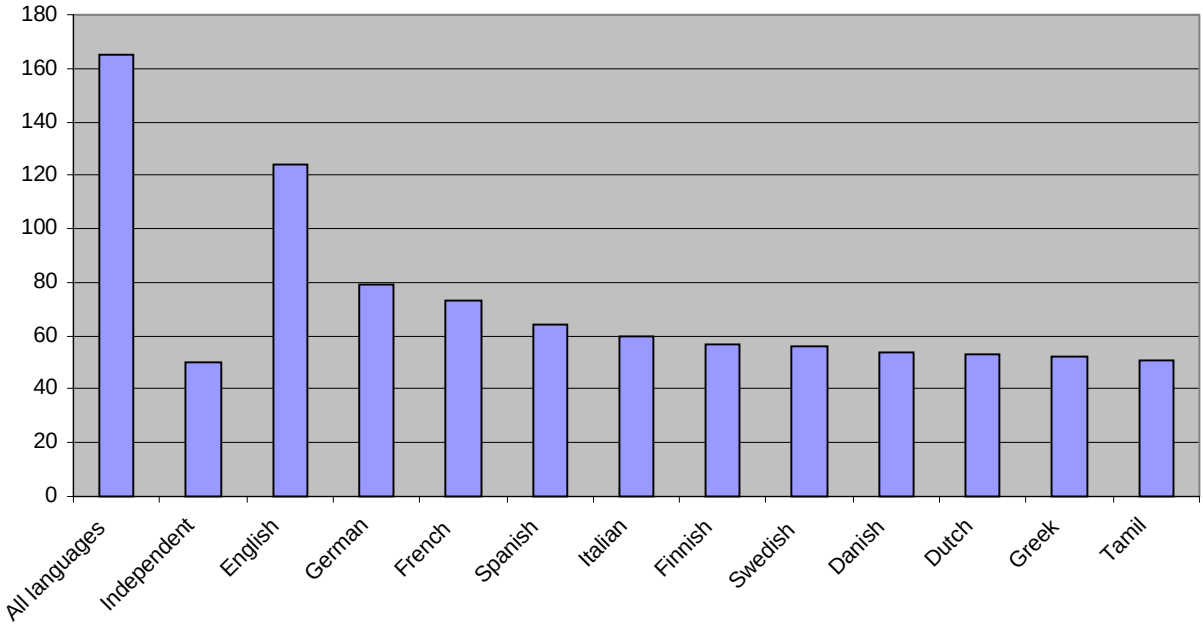


Figure 4. Fourth phase: Multilingualism and general applications..



Fourth phase: Multilingualism and general applications.

- Information retrieval and extraction.
- Translation aids. Integrated use of multiple on-line dictionaries, translation of noun phrases and simple sentences.
- Corpus IV. Semantically tagged text after word-sense disambiguation.
- Knowledge base on multilingual lexico-semantic relations and its applications.
- Dialog systems.

Now that we have started working on the fourth phase, every foundation, tool and application developed in the previous phases is of great importance to face new problems.

3. Present portable HLT products

Which is the start point at the present? Which kind of public HLT products could be integrated, at the moment, in a basic toolkit portable for any language?

With the aim of looking for data to answer to those questions, we examined the programs registered in the Natural Language Software Registry[2] (NLSR), an initiative of the Computational Linguistics Association (CL) and hosted at DFKI in Saarbrücken. The NLSR concentrates on listing HLT software, but it does not exclude the listing of linguistic resources (corpus, monolingual and multilingual lexicon). Other institutions, such as ELRA/ELDA or the Linguistic Data Consortium, provide listings of such resources. However, looking for portable products, to be precise, looking for products usable for multiple languages, the NLSR result sufficient because, actually, all linguistic resources are related to particular languages and so, they are not significant in this search. Of course, there are other HLT tools that have not been submitted to the NLSR, but we think that examine this database is a good start point.

3.1. Present proportion between portable and not-portable HLT products

First of all, we looked for how many of the present HLT tools and applications support different languages. This task was not very difficult because the system allows queries with a particular value for the slot named *Supported language(s)*. Figure 5 shows that a) the all amount of programs registered is 167; b) 50 of them (30%) has been declared to be language independent; c) of course, English is the language that support most of the programs. 125 support English (75%), that means that only 42 systems have been defined for the remaining 24 languages defined in NLRS; d) German, French, Spanish and Italian are the next languages an they are supported only by 79, 73, 64 and 60 respectively; and e) other languages are supported by those fifty defined as language independent and, occasionally, by a few other programs, for example 51 hits for Tamil. Those data reveals evident the significance of portability in Natural Language Software.

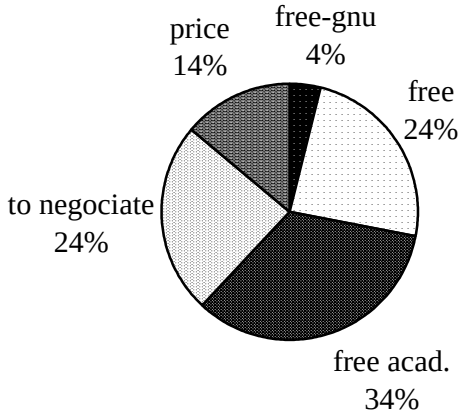


Figure 6: Price of portable HLT products

[2] http://registry.dfki.de

3.2. Price of portable HLT products

How many of the portable HLT products are free for academic and commercial uses? Among the fifty products they are 14 programs that free for any use (two of them, Zdatr and the speech synthesizer MBROLA, are distributed under the GNU Public Public License). Other 17 systems are free for academic uses. The price of 12 systems is defined as "to negotiate" even for academic uses. And finally 7 systems has a fixed price stated from $129 to $799; their average price is $546.

3.3. Distribution of portable products between HLT sections

Is there any portable tool for all the main basic sections in HLT? Or… is there any application with no accessible tools? Table 1 shows the distribution by sections of language independent software in NLSR. Similar data is shown for products that support English. We remark the following points: a) the number of products for the last four sections is not enough to be considered: b) the distribution of language independent products is similar to that of the total amount of products; c) there is any system in every section; d) the percentage of language independent products is considerable higher in Spoken Language and in NLP Development Aid.

Section	Total	Indep.	% indep.	Eng.	% Eng.
Total	167	50	0,30	125	0,75
Annotation	15	4	0,27	13	0,87
Written lang.	122	28	0,23	90	0,74
Spoken language	31	15	0,48	23	0,74
NLP development Aid	41	16	0,39	31	0,76
Lang. Resources	23	6	0,26	18	0,78
Multimedia	2	1	0,50	1	0,50
Multimodality	5	1	0,20	4	0,80
Evaluation	4	3	0,75	4	1,00

Table 1: Distribution of software by HLT sections

And now let's consider the distribution of NSLR products taking into account the kind of linguistic knowledge they manage. The kinds of knowledge to be considered are those referred in the previous section plus special points for NLP frameworks than includes facilities for lexical, morphology, syntax or speech. There is not any program to deal with dictionaries (creation of structured versions of dictionaries or integration of them in other applications), nor for semantics.

3.3.1. Corpus

Product	Description	Price
Alembic Workbench	a multi-lingual corpus annotation development tool	free
Bigram Statistics Package	Bigram analysis software	free
emdros	text database engine for linguistic analysis and research	free
PWA	Word Aligner	free acad.
SRILM -- SRI Language Modeling Toolkit	Statistical language modeling toolkit	free acad.
Entropizer 1.1	A toolbox for sequential analysis	to negotiate

Table 2: NLSR language independent products for corpus

3.3.2. Morphology

Product	Description	Price
PC-KIMMO	Two-level morphological analyzer	free acad.
TnT - Statistical Part-of-Speech Tagging	a statistical part-of-speech tagging for german, english and languages that delimit words with space	free acad.

Table 3: NLSR language independent product for morphology

3.3.3. Lexical databases

Product	Description	Price
DATR	A formalism for lexical knowledge representation	free
Xerox TermOnLine	Xerox TermOnLine is a terminology database sharing tool	to negotiate
Xerox TermOrganizer	Xerox TermOrganizer is a terminology database management system.	to negotiate

Table 4: NLSR language independent product for lexical databases

3.3.4. Speech

Product	Description	Price
IVANS: The Interactive Voice ANalysis System	Voice analysis, voice quality rating, voice/client data management	$749
CSRE - Computerized Speech Research Environment	speech analysis, editing, synthesis and processing system	$750
The OroNasal System	Nasalance measurement, analysis of oral and nasal airflow/energy in speech	$799
CSLU Toolkit	a comprehensive suite of tools to enable exploration, learning, and research into speech and human-computer interaction	free acad.
CSL -- Computerized Speech Lab	speech acquisition, analysis and playback	to negotiate
Signalyze(tm)	Interactive program for speech/signal analysis (runs only on Macintosh)	$350
TFR: The Time-Frequency Representation System	a comprehensive speech/signal analysis, editing and processing system	$599
Multi-Speech	a comprehensive speech recording, analysis, feedback, and measurement software program	to negotiate
WinPitch, WinPitch II	Speech analysis and annotation	to negotiate
ProTrain	speech analysis and speech production training system	$349
Praat	a research, publication, and productivity tool for phoneticians	free acad.
MBROLA	a speech synthesizer based on the concatenation of diphones	free-GNU
EULER	a freely available, easy-to-use, and easy-to-extend, generic multilingual TTS	to negotiate

Table 5: NLSR language independent product for speech

3.3.5. Syntax

Product	Description	Price
ASDParser and ASDEditor	Parser and editor for Augmented Syntax Diagram grammars, implemented in Java.	free
XLFG	Syntactic analysis using the LFG formalism	free
AGFL Grammar Work Lab	Formalism and tools for context free grammars	free acad.
CUF	constraint-based grammar formalism	free acad.
GULP -- Graph Unification Logic Programming	an extension of Prolog for unification-based grammar	free acad.
LexGram	development and processing of categorial grammars	free acad.

Table 6: NLSR language independent product for syntax

3.3.6. NLP framework

Product	Description	Price
Alembic	an end-to-end multi-lingual natural language processing system	free
The Quipu Grok Library	a library of Java components for performing many different NLP tasks	free
PAGE: A Platfrom for Advanced Grammar Engineering.	System for linguistic analysis and test of linguistic theory (HPSG, FUG, PATR-II). Can be used as part of a deep NLP system or as part of a speech system (a special version is used in Verbmobil).	to negotiate
TDL---Type Description Language	System for linguistic analysis and test of linguistic theory (HPSG, FUG, PATR-II). Can be used as part of a deep NLP system or as part of a speech system (a special version is used in Verbmobil).	to negotiate
QDATR	An implementation of the DATR formalism	free acad.
Kura	Kura is a system for the analysis and presentation of linguistic data such as interlinear texts.	free
Zdatr	Zdatr is a standardised DATR implementation in ANSI C	free-GNU

Table 7: NLSR language independent product for NLP frameworks

3.3.7. Applications

Product	Description	Price
BETSY - Bayesian Essay Test Scoring sYstem	Free Windows based text classifier/essay scorer	free acad.
Flag	Terminology, style and language checking	to negotiate
Universal Translator Deluxe	An omni-directional translation system	$129
Onix	High performance information retrieval engine	to negotiate
Brevity	Document summarization toolkit	to negotiate

Table 8: NLSR language independent product for applications

4. A standard representation for linguistic data using TEI conformant feature structures

The standard representation of linguistic data in order to allow the integration between different tools in the same HLT framework will be fundamental for any possible basic toolkit. In this section we present as a proposal the strategy used for the integration, in a common framework, of the NLP tools developed for Basque during the last twelve years (Artola et al.; 2000). The documents used as input and output of the different tools contain TEI-conformant feature structures (FS) coded in SGML[3]. These FSs describe the linguistic information that is exchanged among the integrated analysis tools.

The tools integrated until now are a lexical database, a tokenizer, a wide-coverage morphosyntactic analyzer, a general purpose tagger/lemmatizer, and a syntactic parser.

Due to the complexity of the information to be exchanged among the different tools, FSs are used to represent it. Feature structures are coded following the TEI's DTD for FSs, and Feature System Declaration (FSD) descriptions have been thoroughly defined.

The use of SGML for encoding the I/O streams flowing between programs forces us to formally describe the mark-up, and provides software to check that this mark-up holds invariantly in an annotated corpus.

A library of Abstract Data Types representing the objects needed for the communication between the tools has been designed and implemented. It offers the necessary operations to get the information from an SGML document containing FSs, and to produce the corresponding output according to a well-defined FSD.

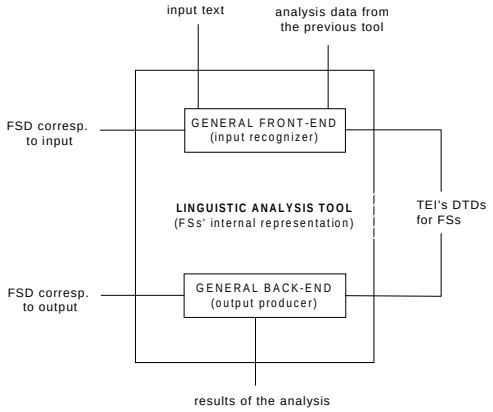


Figure 7. Schematic view of a linguistic analysis tool with its general front-end and back-end.

The use of SGML as an I/O stream format between programs has, in our opinion, the following advantages:

a) It is a well-defined standard for the representation of structured texts that provides a formal framework for the internal processing.
b) It provides widely recognized facilities for the exchange of data: given the DTD, it is easy to process any conformant document.
c) It forces us to formally define the input and the output of the tools used for the linguistic analysis of the text.
d) It facilitates the future integration of new tools into the analysis chain.
e) Pieces of software are available for checking the syntactic correctness of the documents, information

[3] All the references to SGML in this section could be replaced by references to XML.

retrieval, modifications, filtering, and so on. It makes it easy to generate the information in different formats (for processing, printing, screen-displaying, publishing in the web, or translating into other languages).

f) Finally, it allows us to store different analysis sets (segmentations, complete morphosyntactic analyses, lemmatization results, and so on) linked to a tokenized piece of text, in which any particular analysis FS will not have to be repeated.

.

5. Conclusions

If we want HLT to be of help for more than 6000 languages in the world, and not a new source of discrimination between them, the portability of HLT software is a crucial feature. Looking for language independent software in the Natural Software Registry, we saw that only 30% of the tools has been so declared; that 62% of those language independent programs are at least academic free and that they are quite homogeneously distributed among the different sections of HLT and among the kinds of knowledge they manage.

As many problems would arise when trying to coordinate several of those language independent programs, we present as a proposal the strategy used for the integration, in a common framework, of the NLP tools developed for Basque. Feature structures are used to represent linguistic information, and feature structures are coded following the TEI's DTD for FSs, and Feature System Declaration descriptions (FSD) have been thoroughly defined.

Worldwide international organizations that work for the development of culture and education should promote the definition and creation of a basic toolkit for HLT available for as many languages as possible. ISCA SALTMIL SIG should coordinate researchers and those organisations to initiate such project.

References

Artola X., Díaz de Ilarraza A., Ezeiza N., Gojenola K., Maritxalar A., Soroa A.. A Proposal for the Integration of NLP Tools using SGML-tagged documents. In *Proc. of the Second Int. Conf. on Language Resources and Evaluation*. Athens (Greece). 2000

Petek B. "Funding for research into human language technologies for less prevalent languages" Second International Conference on Language Resources and Evaluation (LREC 2000). Athens, Greece.

Sarasola K. "Strategic priorities for the development of language technology in minority languages". Proceedings of Workshop on "Developing language resources for minority languages: re-useability and strategic priorities". Second International Conference on Language Resources and Evaluation (LREC 2000). Athens, Greece.

Ubiquitous multilingual corpus management in computational fieldwork

Dafydd Gibbon

Fakultät für Linguistik und Literaturwissenschaft
Universität Bielefeld
Postfach 100131
D–33501 Bielefeld
Germany
gibbon@spectrum.uni-bielefeld.de

Abstract

The present application addresses the issue of portability in the context of linguistic fieldwork, both in the sense of platform interoperability and in the sense of ultra-mobility. A three-level networked architecture, the UbiCorpus model, for information gathering in the field is described: (1) a Resource Archive server layer, (2) a Data Processing application layer, and (3) a new Corpus Pilot layer designed to support specific fieldwork sessions under adverse conditions, for on-site questionnaire presentation and metadata editing.

1. Goals

In linguistic fieldwork,[1] conceptually the initial stage in any language documentation procedure, the issue of portability is important in two senses: first, the sense of platform interoperability and second, in the sense of ultra-mobility. This issue is addressed by the present application. A three-level networked architecture, the UbiCorpus model, for information gathering in the field is described: (1) a Resource Archive server layer, typically non-mobile, and distributed; (2) a Data Processing application layer, typically a local laptop or desktop; and (3) a new Corpus Pilot layer, designed to support specific fieldwork sessions under adverse conditions with questionnaire presentation and metadata editing, and typically, it is suggested, implemented on a handheld PDA. The UbiCorpus model is based on extensive fieldwork experience, mainly in West Africa. The Corpus Pilot layer is described in detail.

Owing to severe financial and platform resource limitations in practical linguistic fieldwork situations, the general development strategy is to use available freeware or open source components as far as possible, and to augment these with custom applications which are distributed as freeware for initial testing, and subsequently published as as open source software.

Figure 1: Questionnaire-based interview on Anyi syntax with Kouamé Ama Bié by Sophie Salffner & Sandrine Adouakou in Adaou, Ivory Coast (equipment: field laryngograph, DAT, Palm, pen & paper).

2. Requirements specification

Relatively recently, issues of corpus standards and resources as developed in the field of speech technology (Gibbon et al., 1997; Gibbon et al., 2000; Bird and Liberman, 2001) have been extended to fieldwork corpora in linguistics, ethnography, and related sciences, and specific issues such as the role of metadata in resource archiving and reusability have come to the fore, adding to the complexity of the documentation task facing the fieldworker. The present application area is computational support for this fieldwork documentation task within an integrated fieldwork resource environment. This concern is on the one hand more comprehensive than the currently popular issues of annotation-based data enhancement and web-based resource dissemination, and on the other hand orthogonal to these expensive technologies in that an effective but inexpensive practical new "low end high tech" technique for grass roots applications in geographically inaccessible areas is introduced.

From the perspective of field linguistics, language documentation traditionally consists in the main of field notes, an outline of the situation of the language, transcriptions, and generally including a sketch grammar consisting of basic phonology, morphology, and grammar, together with a lexicon containing glosses and examples and perhaps a thesaurus. The prompt materials for eliciting this kind of documentation are mainly systematic linguistic and ethnographic questionnaires, and the media for production of the documentation are generally office-oriented software such

[1]Grateful acknowledgements to Sandrine Adouakou, Firmin Ahoua, Doris Bleiching, Bruce Connell, Eddi Gbery, Ulrike Gut, Ben Hell, Sophie Salffner, Thorsten Trippel and Eno-Abasi Urua for discussion of problems addressed in this contribution.

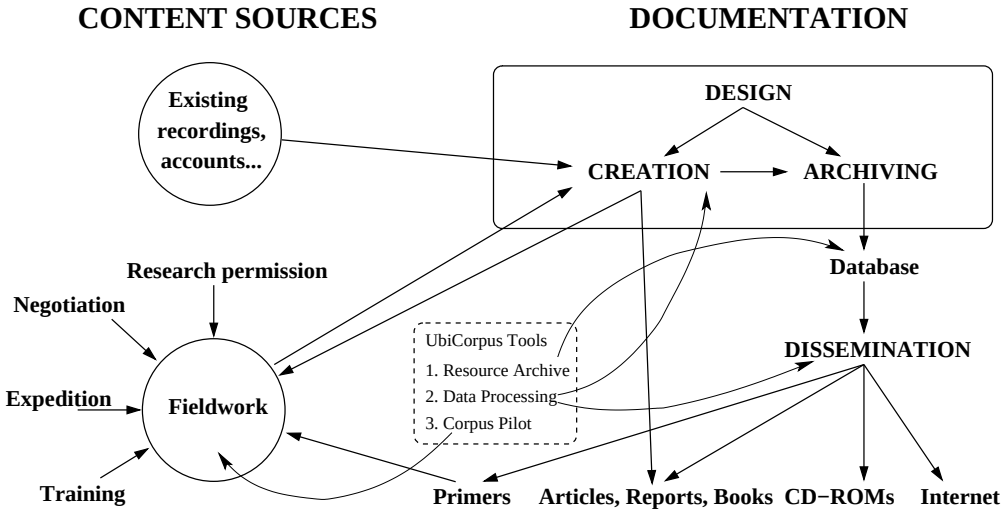


Figure 2: Language documentation logistics model.

as word processors (MS-Word etc.), DBMS (Access, FilemakerPro etc.), and spreadsheets (Excel, etc., also used for database entry). The guiding objectives of this concept of documentation are applications in the production of translations, terminologies, and alphabetisation materials.

The UbiCorpus model is designed to support this kind of fieldwork in the following main respects:

1. questionnaire presentation (either by database or in free format, as a plain text editor or with special formatting and rendering, for example by means of an IPA font),

2. transcription (either plain ASCII such as X-SAMPA, or in an IPA font),

3. metadata input.

One of the main advantages of the model is that when implemented on a modern palmtop device it provides a convenient, efficient and — important for many applications — inconspicuous method for the frequently neglected task of systematic on-site metadata logging.

However, the scope of the model is more general, and supports both the documentation of spoken language corpora in general, and further corpus processing in the form of the development of structured computational lexica (van Eynde and Gibbon, 2000) and computationally supported grammar testing. The UbiCorpus model is embedded in a comprehensive documentation model which covers not only the fieldwork activity itself, but the environment of preparation, archiving and application in which fieldwork is embedded (cf. Figure 2).

The first general operational requirement for the UbiCorpus model is portability. In the present context the term is systematically ambiguous:

- interoperability of applications on different OS and hardware platforms,

- compatibility of data formats through import and export filters for functionally equivalent or interfaced applications,

- ubiquity, i.e. time and place independent mobile deployment.

In the present context, the primary focus is on ubiquity, with interoperability and compatibility seen from this perspective.

Computational support for certain aspects of linguistic fieldwork has been available for many years, both for laptop-based data entry and initial analysis on the move or in isolated areas, and for desktop-based detailed descriptive work and document production (with increasing overlap between laptop and desktop functionalities). Software applications have been characteristically in the following areas:

- Lexical databases, either using general office DBMS such as FileMakerPro and MS-Access, or custom lexicon project software such as SIL's Shoebox; the latter also includes lexical support for textual glossing.

- Publication support such as DB export functions, fonts.

- Phonetic software, for signal analysis (e.g. general signal editors such as CoolEdit, or SIL's CECIL and signal analysis packages, or Praat) and for the symbol-signal time alignment (labelling) of digital recordings (e.g. Praat, Transcriber).

- Computational linguistic software for basic phonological, morphological and syntactic processing.

Some of this functionality (lexical databases, document production, computational linguistic processing) overlaps with the new Corpus Pilot layer, but this layer has the following characteristic additional fieldwork corpus acquisition functionality (Gibbon et al., 1997; Gibbon et al., 2000):

Pre-recording phase: planning of the overall corpus structure and contens, in particular design of corpus recording sessions, including the preparation of scenario descriptions, interview strategies, questionnaires, data prompts (for instance with prompt randomisation),

Recording phase: conduct of corpus recording sessions, including session management with the logging of metadata in a metadata editor and database, questionnaire consultation and data prompt presentation;

Post-recording phase: provision of recorded and logged data for archiving and processing, including metadata export, transcription, lexicon development, systematic sketch grammar support and document production.

3. Design: modules, interfaces

The language documentation model within which the UbiCorpus model is deployed is visualised in Figure 2; the documentation model was developed for project work in West Africa. The two components of the model with which the UbiCorpus tools are concerned are the *Creation* and *Archiving* component, and the *Fieldwork* information source. The latter is directly associated with the Corpus Pilot layer described below. The UbiCorpus model itself is visualised in Figure 3.

The three layers of the UbiCorpus model are characterised as follows:

Resource Archive (RA) layer

The bottom layer represents the archive database and the access and media dissemination functions associated with it. On the declarative side, a number of current language resource and documentation proposals may be assigned to the Resource Archive layer: a single resource database such as a corpus or a lexicon, a multiple resource database such as a browsable corpus or concordance system, a web portal constituting a large and systematic resource world, or an entire dissemination agency. On the procedural side, the Resource Archive layer provides search functions of various kinds, from standard browsing strategies to intelligent search and concordance construction, with token renderings of resources in any suitable media, whether entire corpora or lexica.

Data Processing (DP) layer

This is the layer which is familiar to the "ordinary working linguist". The data include paper fieldwork logbooks, transcriptions, sketch grammars and card index lexica; word processor and database versions of these; analog and digital audio and video recordings; time aligned digital annotations of recordings, and concordance or browsing software based on annotations; metadata catalogues for all of these Data Processing layer data types. Procedurally, the platforms and applications used at the Data Processing layer are very varied, though there is a tendency to go for platform independence and standardised data interchange formats. By using modern laptops, both the Resource Archive and Data Processing layers can be integrated into a single mobile environment.

Corpus Pilot (CP) layer

The top layer of the model represents the functionality which needs to be available in an actual fieldwork situation. This functionality can be very varied, and much — especially free format interviews and film recording — lies outside the range of systematic computational support. However, the following on-site support features can easily be covered:

1. metadata editor and database,
2. participant database for interviewee, interviewer etc.,
3. structured or free format questionnaire presentation.

Interfaces

The interfaces between these three layers, and modules within these layers, are defined mainly on the basis of generic ASCII formats, including XML annotated text, CSV database tables, and RTF formatted documents (including IPA font information). For the interface between a palmtop implementation of the Corpus Pilot layer and the Data Processing layer, conversion scripts are provided as required, in order to export palmtop database and text formats into the generic ASCII formats. Data transfer at the implementation level is via the usual synchronisation functions provided with handheld devices, or via scp, http, and ftp procotols for laptops, desktops and server.

4. Implementation: hybrid applications

Resource Archive (RA) layer

The server archive provides web portal access for the local and global linguistic communities, CD-ROM access for the local linguistic community, and analogue selections (in general, tape cassette, print media) for practical applications in the local user community. Currently, the leading models for the Resource Archive level are provided by the LDC and ELRA dissemination agencies; the E-MELD project is developing a general model for best practice in resource collation, and a meta–portal for flexible access to language resources. The local server currently used for initial database collation contains a number of specific search functionalities for corpus analysis, in particular an audio concordance (Gibbon and Trippel, 2002).

Data Processing (DP) layer

The classical environment for fieldwork data processing is a laptop, often a Mac, but also very frequently an Intel based device configured alternatively with Linux or MS based portable standard software. The kinds of application typically used are for basic corpus processing: Transcriber and Praat for transcription and annotation; Shoebox for lexical database development; MS Office or StarOffice for word processor, database and spreadsheet applications. These may be augmented with custom applications in Java (cf. the TASX engine (Milde and Gut, 2001)) and Perl (PAX audio concordance).

Corpus Pilot (CP) layer

The Corpus Pilot layer is implemented as custom-developed Palm compatible PDA applications. The rationale behind the use of the PalmOS based handhelds, as opposed to the use of a laptop, is based on the following considerations:

1. extremely inexpensive (in relation to other computational equipment),

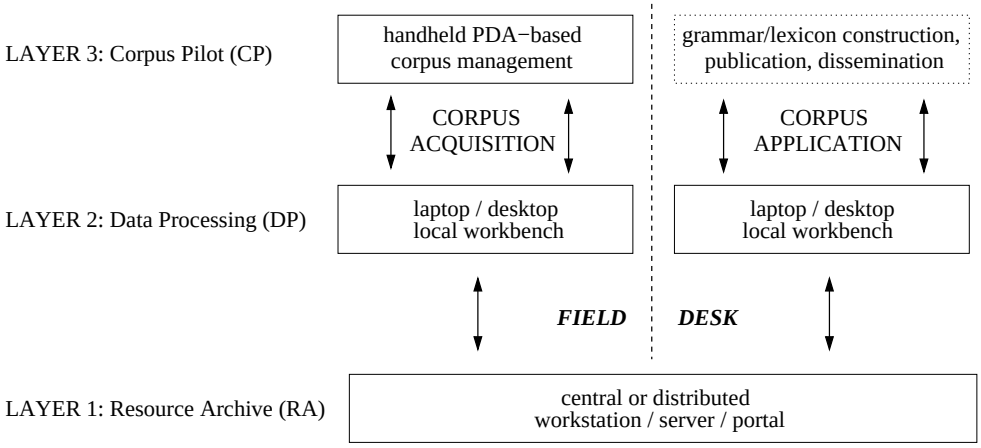


Figure 3: The three layer UbiCorpus model.

2. ultra-lightweight (lighter than other standard portable fieldwork equipment such as field laryngograph, DAT recorder),

3. long operating cycle (with normal use, around 3 weeks on 2 AAA batteries or one charge), depending on model,

4. fast and highly ergonomic in use,

5. small and unobtrusive in the interview situation,

6. an integrated environment with other PDA functionalities such as calendar, diary, address and other databases, other custom applications in C and Scheme.

Networking

The three levels are networked by standard techniques: server-to-applications in general via TCP/IP-based protocols and mobile or landline telephone. The applications-to-acquisition via dedicated sychronisation software of the kind typically used to link handheld PDAs to desktop installations.

Use in the field

The satisfaction of these criteria points towards a high level of suitability for use in extreme fieldwork situations without power supplies, for instance in isolated outdoor locations (forest, village, etc.).

The functionality which has been included in the Corpus Pilot layer so far covers the following:

- Metadata editor and database for audio/video recordings, photos, paper notes, artefact cataloguing. This application is based on a widely used PalmOS DBMS application, HanDBase, which provides a wide range of input support facilities (popups, date picker, free format notes, etc.), as well as cross-table linking.

- Questionnaire administration. In general, free text format has been used for questionnaire administration, and responses have been recorded for later out-of-field processing. For some questionnaire types (e.g. demographic information), the HanDBase DBMS is used.

- Lexicon development tools. Three applications are used for lexical database input (excluding freely formatted notes):

  1. an Excel-compatible spreadsheet, QuickSheet, which permits export in either CSV or Excel format (Excel is widely used in field linguistics as a convenient input tool for lexical databases, because of the ease with which databases may be constructed and restructed, and because it has many database-like functions, as well as built-in arithmetic functions if required for corpus work),
  2. the HanDBase DBMS which is also used for the metadata editor database,
  3. an implementation of the DATR lexicon knowledge representation language in LispMe, a Scheme implementation for the PalmOS platform (this application is a more Data Processing layer oriented tool, but is included in the Corpus Pilot layer implementation suite for convenience).

- Transcription support. In general, transcription in X-SAMPA (Gibbon et al., 2000) is used, but if required, IPA fonts may be used with the WordSmith word processor for PalmOS devices; RTF import and export facilities are available.

- Statistics package for initial evaluations. This is also a more Data Processing layer application, but integrated into the Corpus Pilot layer; functions include all the measures used in basic experimental and corpus work (including random sorting, mean, median, standard deviation, standard error, as well as standard pairwise comparison measures).

- Context-free parser package for basic grammar development. This is another Data Processing layer application, which is integrated into the Corpus Pilot layer because of the convenience of the LispMe Scheme application in which the parser suite is implemented.

The metadata application has been selected for detailed description, because it is most immediately relevant to the issue of language resources.

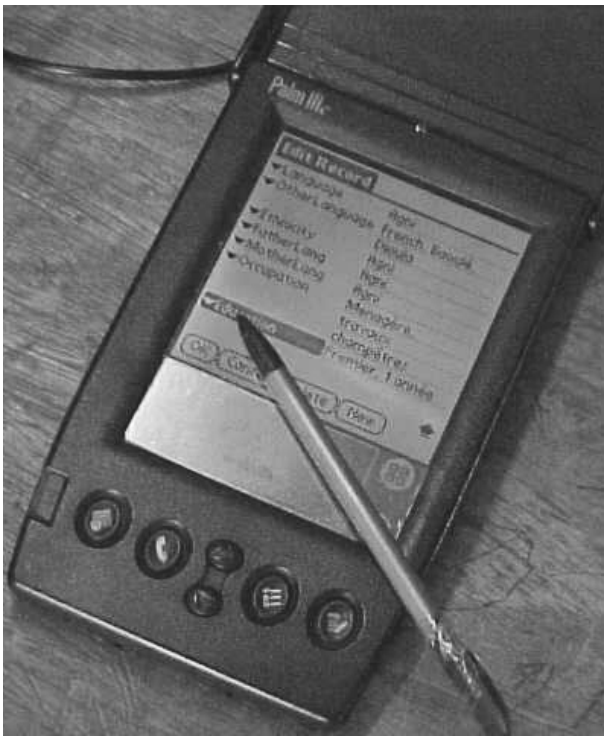


Figure 4: Palmtop metadata editor.

5. Metadata editor and database application

A metadata editor for audio/video recordings, photos, paper notes, artefact cataloguing was designed, based on a standard PalmOS relational database shell (HanDBase). The metadata editor provides a fast and inconspicuous input method for structured metadata for recordings and other field documentation, based on current work on metadata in the ISLE, E-MELD projects, and in the pilot phase of the DOBES project.

For the work in hand, standardised metadata specifications, such as the Dublin Core and IMDI sets, were taken into account. However, new resource types such as those which are characteristic of linguistic fieldwork demonstrate that the standards are still very much under development, since some of the standard metadata types are not relevant for the fieldwork data, and the fieldwork data types contain information not usually specified in metadata sets, but which are common in the characterisation of spoken language resource databases (Gibbon et al., 1997). In respect of the fieldwork resource type, it appears that it cannot be expected that a truly universal — or at least consensual — set of corpus metadata specifications will be developed in the near future, or perhaps at all, at a significant level of granularity. It may be possible to constrain the attribute list, though the existence of many different fieldwork questionnaire types belies this. However, the values of the attributes are in general unpredictable, entailing not only free string types but possibly unpredictable rendering types (e.g. different alphabets; scanned signatures of approval).

Indeed, it may be noted in passing that the expectation of fully standardising the entire metadata specification tends to reveal singularly little awareness of the potential of machine learning and text mining procedures for handling generalisation tasks of this kind. It may be predicted that such procedures will be applied in future not only to extensive resource data sets but also to increasingly extensive sets of metadata.

Table 1: Fieldwork metadata specifications.

Attribute	**Type**
RecordID:	string
LANGname(s):	popup: Agni,Agni; Ega
SILcode:	popup: ANY; DIE
Affiliation:	string
Lect:	string
Country:	popup: Côte d'Ivoire
ISO:	popup: CI
Continent:	popup: Africa; AmericaCentral; AmericaNorth; AmericaSouth; Asia; Australasia; Europe
LangNote:	longstring
SESSION:	popup: FieldIndoor; FieldOutdoor; Interview; Laboratory
SessionDate:	pick
SessionTime:	pick
SessionLocale:	string
Domain:	popup: Phonetics; Phonology; Morphology; Lexicon; Syntax; Text; Discourse; Gesture; Music; Situation
Genre:	Artefacts; Ceremony; Dialogue; Experiment-Perception; ExperimentProduction; History; Interview; Joke/riddle; Narrative; Questionnaire; Task
Part/Sex/Age:	string
Interviewers:	string
Recordist:	string
Media:	popup: Airflow; AnalogAudio; AnalogAV; AnalogStill; AnalogVideo; DigitalVideo; DigitalAudio; DigitalAV; DigitalStill; DigitalVideo; Laryngograph; Memory; Paper
Equipment:	longstring
SessionNote:	longstring

In consequence, the metadata specifications used in the UbiCorpus applications are deliberately opportunistic, in the sense that they are task-specific and freely extensible. A selection of attributes and values for the current fieldwork application are shown in Table 1. Metadata attributes concerned with the Resource Archive layer of archiving and property rights are omitted.

For current purposes, databases are exported in the attribute-value format shown below and converted into the TASX reference XML format (Milde and Gut, 2001). A specific example of the application of the metadata editor in the fieldwork session pictured in Figure 1 is shown in the exported record shown in Table 2.

The metadata editor and database application has been tested extensively in fieldwork on West African languages, and has proved to be an indispensable productivity tool, especially in difficult situations where very limited time is available.

6. Conclusion

Architectures using the first two levels, e.g. a server configuration and a laptop for use in the field, are very com-

Table 2: Fieldwork metadata example.

Attribute	Value
RecordID:	Agni2002a
LANGname(s):	Agni, Anyi
SILcode:	ANY
Affiliation:	Kwa/Tano
Lect:	Indni
Country:	Côte d'Ivoire
ISO:	CI
Continent:	Africa
LangNote:	
SESSION:	FieldIndoor
SessionDate:	11.3.02
SessionTime:	8:57
SessionLocale:	Adaou
Domain:	Syntax
Genre:	Questionnaire
Part/Sex/Age:	Kouamé Ama Bié f 35
Interviewers:	Adouakou
Recordist:	Salffner, Gibbon
Media:	Laryngograph
Equipment:	1) Audio: 2 channels, l laryngograph, r Sennheiser studio mike 2) Stills: Sony digital 3) Video: Panasonic digital (illustration of techniques)
SessionNote:	Adouakou phrases repeat

mon. However, in many situations the laptop concept is unsuitable because of heavy power requirements which are not available in many fieldwork locations. For these applications, the PalmOS based family constitutes the platform of choice because of minimal size and power requirements, permitting several weeks use on one charge or small battery. Although the PalmOS platform is obviously unsuitable for signal processing applications (such as time-aligned annotation) it is well-suited for logging, transcription and reference purposes.

The power of PDA miniature computing platforms as useful components of laboratory and office environments is often underestimated, and we demonstrate that a number of applications for which even a laptop is clumsy or unsuited for the developing field of computational ethnolinguistic fieldwork may be elegantly provided on the Palm PDA platform. The addition of a foldable keyboard further enhances the text handling capacity of the devices.

In the medium term, it will be possible to integrate the hybrid applications at the Corpus Pilot, Data Processing and Resource Archive levels into a corpus management environment which not only permits seamless dataflow and workflow, a goal already achieved, but also into a non-technical user-friendly prototype which may serve as the basis of a fieldwork management product implementation.

The UbiCorpus architecture has been used as the basic specification for different kinds of language documentation work in a variety of different projects. The Resource Archive layer was originally designed and implemented for web–based lexical database development in the VerbMobil project (Wahlster, 2000), funded by the German Federal Ministry of Education and Research (BMBF). The concept has been further developed theoretically and practically in connection with the projects *Theorie und Design multimodaler Lexika* funded by the German Research Council (DFG), *Enzyklopädie der Sprachen der Elfenbeinküste* funded by the German Academic Exchange Service (DAAD) and *Ega: a documentation model for an endangered Ivorian language* in the pilot phase of the DOBES funding programme of the Volkswagen Foundation.

In its local implementation, the current Resource Archive layer version also includes support for telecooperation and web-teaching. The Data Processing layer includes numerous applications which cannot be specified here. The Corpus Pilot layer as described in the present contribution has been informally but extensively field tested at a number of fieldwork locations, most recently in the framework of DAAD funded doctoral thesis work. It is planned to apply the field testing criteria defined in (Gibbon et al., 2000) to an extended implementation of the components of UbiCorpus model.

7. References

Steven Bird and Mark Liberman. 2001. A formal framework for linguistic annotation. *Speech Communication*, (33 (1,2)):23–60.

Dafydd Gibbon and Thorsten Trippel. 2002. Annotation driven concordancing: the pax toolkit. In *Proceedings of LREC 2002*. LREC.

Dafydd Gibbon, Roger Moore, and Richard Winski, editors. 1997. *Handbook of Standards and Resources for Spoken Language Systems*. Mouton de Gruyter, Berlin.

Dafydd Gibbon, Inge Mertins, and Roger K. Moore, editors. 2000. *Handbook of Multimodal and Spoken Dialogue Systems, Resources, Terminology and Product Evaluation*. Kluwer Academic Publishers, Boston/Dordrecht/London.

Jan-Torsten Milde and Ulrike Gut. 2001. The TASX-engine: an XML-based corpus database for time aligned language data. In *Proceedings of the IRCS Workshop on Linguistic Databases*, Philadelphia. University of Pennsylvania.

Frank van Eynde and Dafydd Gibbon. 2000. *Lexicon Development for Speech and Language Processing*. Kluwer Academic Publishers, Dordrecht.

Wolfgang Wahlster, editor. 2000. *Verbmobil: Foundations of Speech-to-Speech Translation*. Springer Verlag.

A Theory of Portability

Hyo-Kyung Lee*

*Department of Computer Science
1304 W. Springfield Ave., Urbana, IL 61801, USA
h-lee5@cs.uiuc.edu

Abstract

To discuss the portability of human natural language technology, it is necessary to define the portability precisely first. If one claims that his or her language technology works for other languages, how can we verify such claim when every language has a different set of features, i.e. speech or text tagging system? This paper presents a view of protabilty as a function of a common representable set of features and argues that the development of such representation is critical in discussing portability issues.

1. Introduction

If you try a sentence boundary identification program[1] developed for English, you will easily notice that it does not work for other language such as Korean. However, the developers never mention that the program will not work for other languages. It is a very common practice among developers to ignore the portability issues in human language related technology because it is often targeted for only one language and assumed to work for that language. Yet, such ignorance is missing too many opportunities for the future success of the technology. If a technology that works for one language can be extended to another language with a minimal modification, such technology can be regarded as the most valuable technology in its potential considering the fact that there exists more than 6,000 languages in the world.

The main difficulty of applying a technology that works for one language into another is obviously due to the set of features that are unique to one specific language. The more the technology resorts to those features, the less it will succeed on other languages. In this regard, it is necessary to separate out those features and to concentrate on the common features that every language shares for maximal portability. Finding the common grounds for all languages is not an easy task but can be achieved by abstracting levels of language processing into hierarchy. In other words, there are different levels of sources that hinders the portability during human language processing and the portability problem should be discussed in as high level as possible.

This leads to the central question of this paper: if one claims that one's language technology works for other languages, how can we verify such claims when every language has a different set of features, i.e. speech or text tagging system and quantify its portability? The only way to determine it is to test how many features are translatable into the common feature sets which are similar to the interlingua in machine translation. This paper presents a view of portability as a function of a common representable set of features and argues that the development of such representation is critical in discussing portability issues.

[1] http://l2r.cs.uiuc.edu/~cogcomp/cc-software.html

2. Sources of Portability Problem

To identify the sources of portability problem in human language technology, it seems wise to illustrate it with the actual examples that might occur in statistical machine translation, namely sentence boundary identification for aligning sentences and word sense disambiguation for word selection. Here, we identify two categories in portability problems and formalize it. For the rest of the paper, we use a term *program* to denote a particular instance of human language technology.

2.1. Representational Problem

Almost every language has its own unique set of features. At the same time, some languages share many common features. For example, semantic or syntactic features like the notion of person's name and noun phrases are quite universal. On the other hand, honorifics used in Korean or Japanese language is hard to find in European languages. Such uniqueness of features is the major obstacle for the portability of human language processing program. For example, if you want to create a sentence-aligned corpus for statistical machine translation, the first step is to identify sentence boundaries. If one program uses the notion of capitalized word to determine whether the period is used for abbreviation or not, it won't work for a language, like Korean, that doesn't have any notion of the capital word in its writing system. For a program to be fully portable, it should avoid using such features.

Clearly, we can distinguish between two different feature sets which we will call *soft* and *hard* features. If some features are common in two or more languages, we call them soft features; otherwise hard features. Soft features are ubiquitous in the same families of languages and they are all functionally equivalent. One key observation is that features are independent from the surface forms of one particular language. For example, the same parsing program can be used to parse two different languages although grammatical notations of the languages are different as long as they can be mutually translated into the equivalent representation.

2.2. Functional Problem

Although two languages share the same soft features, not all functions consistently generate the desired outputs based on them. Let's assume that we want to disambiguate

senses of English words based on the local context feature such as n-grams to find the corresponding Korean words in statistical machine translation(Ng and Lee, 1996). If you could achieve 90% of accuracy on the task with such method in English, it does not guarantee the same accuracy in Korean. The reason for such discrepancy can be attributed to the previous representational problem but the key issue that we want to emphasize here is only the performance aspect of the program. This is a separate dimension of portability problem which is related to the performance-wise consistency issue. A good portable program should perform well with the minimal variance and high accuracy across several languages. For example, if a machine translation system that performs very well on English-Korean translation fails on English-Japanese translation with the same soft set of features, we can say that such system has a functional problem in portability.

3. Theory of Portability

In this section, we present a functional view of portability in a more formal way by providing definitions and examples first and then theorems derived from them.

3.1. Definition of Portability

Definition 1 Features *are any properties of language that are used in a program P as inputs X and outputs Y. A* program *P is a collection of* functions *f.*

Example 1 *Period, question mark, and exclamation mark are features used in English for sentence boundaries.*

Definition 2 *Let's denote a set of all features for language L as $\chi(L)$.* Soft features *Z for a set of languages $L_1, \ldots, L_n$ are features s.t. $\forall z \in Z, z \in \bigcap_i^n \chi(L_i)$. All other features that are not soft features are* hard features *$Z' = \bigcup_i^n \chi(L_i) - Z$.*

Example 2 *Let $n = 2$ and L_1 is Korean and L_2 is English. Period, question mark, and exclamation marks are soft features used in both L_1 and $L2$ for sentence boundaries. The capitalization of words is a hard feature unique in L_2.*

Definition 3 *A family of languages are called* σ-similar *if $\frac{|Z|}{|Z|+|Z'|} = \sigma$ w.r.t. P.*

Example 3 *Let's assume that a sentence boundary identification program P uses only four features: period, question mark, exclamation mark, and a test value $\{0, 1\}$ for capitalization for the word that ends with a period. Again, if $n = 2$ and L_1 is Korean and L_2 is English, two languages are $\sigma(= 0.75)$-similar since $|Z| = 3$ and $|Z'| = 1$.*

Definition 4 *A function $f: X \rightarrow Y$ is called* σ-portable *for n σ-similar languages $L_1, \ldots, L_n$, if $X \subseteq Z$ and $Y \subseteq Z$ for a soft feature set Z.*

Example 4 *Let f be a classification function that uses the previous four features in Example 3 as input X and the boolean truth value $\{0, 1\}$ (to indicate a sentence boundary) as output Y. Then, f is $\sigma(= 0.80)$-portable.*

Definition 5 *A program P is* σ-portable *iff $\exists \sigma > 0$ among n languages and all f are σ-portable over $\sigma \cdot \sum_{i=1}^{n} |\chi(L_i)|$ soft features.*

Example 5 *Let P be a sentence identification program that has two functions f_1 and f_2. Let f_1 be the classification function in the previous example and f_2 be a boolean function that test the capitalization of words. Then, P is* not *σ-portable for English and Korean because f_2 uses hard feature. If P has only one function f_1, we can claim that P is σ-portable.*

Definition 6 *A function $f: X \rightarrow Y$ is called* ϵ-portable *over n languages $L_1, \ldots, L_n$, if $\forall i, j (1 \leq i, j \leq n), Pr(f_i \neq f_j) < \epsilon$*

Example 6 *Let f be a classification function in Example 4. Since Korean sentences do not use periods for abbreviation purposes, it is easy to see that $Pr(f_k = 1) > Pr(f_e = 1)$ when equal number of examples are represented with soft features. If the difference $Pr(f_k = 1) - Pr(f_e = 1)$ in such empirical performance of f over two languages is less than the predefined bound 0.02, we can say that f is $\epsilon(= 0.02)$-portable.*

Definition 7 *A program P is ϵ-portable if all functions $f \in P$ are ϵ-portable and generates the coherent output over n languages with the confidence at least $1 - \delta$ for some small ϵ, δ.*

Example 7 *Let P is a program that has only one f in Example 6. If f is tested on Japanese and also produced the result of $Pr(f_k = 1) - Pr(f_j = 1) < \epsilon$ and $Pr(f_e = 1) - Pr(f_j = 1) < \epsilon$ with at least $1 - \delta$ accuracy over many examples, we can say that P is $\epsilon(= 0.02)$-portable.*

3.2. Theory of Portability

Here, we introduce the notion of portability similar to the learnability notion in the learning theory (Valiant, 1984). The first theory is related to the representational problem.

Theorem 1 *Soft features are harder to obtain as the number of languages n increases.*

$$|Z_n| \geq |Z_{n+1}|$$

Corollary 1 *Hard features are easier to obtain as the number of languages n increases.*

$$|Z'_n| \leq |Z'_{n+1}|$$

Proof This is obvious from the definition of soft features. Since soft sets are extracted from the common feature sets, there are less features than previous n-th soft feature set unless all features are soft features in $n+1$-th language. On the other hand, hard features are obtained from the union of the feature sets and the size of them grows over n. ∎

Lemma 1 *σ is monotonically decreasing over the number of languages n increases.*

Now, let's look at the hardness of portability issue which is the main topic of this paper by combining two parameters — σ and ϵ. First, let's define probably approximately correct(PAC) portability as follows:

Definition 8 *A program P is* PAC-portable *if P is both σ-portable and ϵ-portable.*

If we apply the same technique used in (Haussler, 1988), we get the *PAC-portability bound* which is similar to the PAC-learnability bound as shown in Equation (1). It is adapted from (Mitchell, 1997) for illustration purpose.

$$m \geq \frac{1}{\epsilon}(\ln |H| + \ln(1/\delta)) \quad (1)$$

We can replace m with $\binom{n}{2} = \frac{n(n+1)}{2}$ and $|H|$ with $2^{\sigma k}$ where $k = \sum_{i=1}^{n} |\chi(L_i)|$ by assuming that the number of different functions are only dependent on the size of soft set $|Z| = \sigma k$.

Theorem 2 *If a program P is PAC-portable with some small σ, ϵ and δ, the total number of portable languages n is bounded by:*

$$\frac{n(n+1)}{2} \geq \frac{1}{\epsilon}(\sigma k \ln 2 + \ln(1/\delta)) \quad (2)$$

The following proof is essentially same as what Haussler (1988) showed.

Proof Let F be all the functions that use soft features in n languages. Clearly, there are at most $|F| = 2^{\sigma k}$ possible functions in P over σk soft features and $k = \sum_{i=1}^{n} |\chi(L_i)|$ is a constant for n languages. Let $f^1, f^2, \ldots, f^l$ be all functions in P such that each pair of functions (f_i^l, f_j^l) over two languages i, j have true error greater than ϵ. We need to consider the all pairs of functions over n languages and there exists $\binom{n}{2} = \frac{n(n+1)}{2}$ pairs for each f^l. We fail in ϵ-portability if and only if at least one of these f^l pairs fails. The probability that at least one of these will be consistent with all n languages is at most

$$l(1-\epsilon)^{\frac{n(n+1)}{2}}$$

And since $l \leq |F|$, this is at most $|F|(1-\epsilon)^{\frac{n(n+1)}{2}}$. Finally, we use an inequality that if $0 \leq x \leq 1$ then $(1-x) \leq e^{-x}$. Thus,

$$l(1-\epsilon)^{\frac{n(n+1)}{2}} \leq |F|(1-\epsilon)^{\frac{n(n+1)}{2}} \leq |F|e^{\epsilon\frac{n(n+1)}{2}}$$

upper bound holds.

We can use the above result to determine the number of languages required to reduce the portability failure below some δ.

$$|F|e^{\epsilon\frac{n(n+1)}{2}} \leq \delta$$

which means that we need

$$\frac{n(n+1)}{2} \geq \frac{1}{\epsilon}(\ln |F| + \ln(1/\delta))$$

By substituting $|F| = 2^{\sigma k}$, we get the

$$\frac{n(n+1)}{2} \geq \frac{1}{\epsilon}(\sigma k \ln 2 + \ln(1/\delta))$$

which is identical to Equation (2) ∎

This is correct because a program P can be regarded as portable for n languages as long as at least one function in F survives the portability test bounded by ϵ and δ. Consequently, it is easy to see that a program P that has many functions needs more languages to ensure portability.

4. Discussion

Although there has been a significant amount of computational linguistic research for major languages such as English for more than fifty years, the portability issue of natural language technology based on such research has not been studied until recently.

However, portability of technology is neither cheap to obtain nor trivial to implement according to our theory. From the functional perspective of language technology, the efforts of linguists can be described as finding good theories or rules that can generate both universal and local features for various languages. Likewise, one of the main reasons in the recent success of statistical natural language processing techniques(Manning and Schutze, 1999) can be found in its portability. Statistical approaches, unlike traditional symbolic approaches, are less dependent on language specific features. Our definition of portability demonstrates that as the number of soft features increases, the same technology is portable for more languages. If σ is fully dependent on n and decreases linearly, the technology is not portable. If one can find a good features that are not affected by n and a robust technology that depends only those features, then such technology can enjoy its maximum portability.

To claim portability of a technology, empirical justifications of its performance guarantee are also required over many languages and this is reflected in the parameter ϵ. What it suggests is that even if the same statistical method that uses the common features in many languages, the distribution of features could be dependent on each language and thus significantly different from others. Our theory clearly demonstrates that reducing portability error ϵ requires $O(\sqrt{n})$ languages to be verified with.

5. Conclusion

We presented a formal PAC framework for the functional view of portability. Although it is still a sketchy work, the main contribution of this work is to define portability in a formal way and show the relation among features and performance measures. Therefore, the development of good theories and rules that can work for as many languages as possible and the empirical application of them is critical in discussing the portability issues.

6. References

D. Haussler. 1988. Quantifying inductive bias: Ai learning algorithms and valiant's learning framework. *Artificial Intelligence*, (36):177–221.

C. Manning and H. Schutze. 1999. *Foundations of Statistical Natural Langauge Processing*. MIT Press.

T. Mitchell. 1997. *Machine Learning*. McGraw Hill.

H. T. Ng and H. B. Lee. 1996. Integrating multiple knowledge sources to disambiguate word sense: An exemplar-based approach. In *Proc. of 34th Conference of the ACL*.

L. Valiant. 1984. A theory of the learnable. In *ACM Symposium on Theory of Computing*, pages 436–445.

A requirement analysis for an open set of human language technology tasks

Fredrik Olsson

Swedish Institute of Computer Science
Box 1263, SE-164 29 Kista, Sweden
fredrik.olsson@sics.se

Abstract

This work presents a requirement analysis and a design proposal for a general architecture for a specified, yet open set of human language technology (HLT) tasks — the set chosen is dubbed *information refinement*. Apart from using information refinement as a means to focus the requirement analysis and accompanying design proposal, the analysis and proposal are based on a survey of a number of projects that have had great impact on the realisation of today's HLT architectures, as well as on the experiences gained from a long-term case study aiming at composing a general purpose tool-kit for Swedish. The analysis and design are currently used in an ongoing effort at SICS to implement an open and general architecture for information refinement.

1. Introduction

During the last few years, the need for general, reusable software for computational linguistics and human language technology (HLT) has become widely acknowledged by the research community as well as by the industry. Usually, the overall motivation of striving for reusable software is to shorten the way from the origin of an idea to its implementation in a prototype system. Utilising reusable software also means that the effort spent in building an HLT system is reduced, and thus, that personal labour can be focused on more important issues.

The aim of this paper is to present a requirement analysis and design proposal for a specified, yet open set of human language technology tasks — *information refinement* is introduced as constituting a set of related tasks intended to serve as a target for developing a general and open architecture, Kaba. The requirement analysis and design proposal presented in sections 7. and 8. are based on three parts: the notion of information refinement (Section 2.); a survey of a number of projects and software that have had great impact on how HLT software is constructed today (TIPSTER, CLE, ALEP, GATE, DARPA Communicator, and ATLAS presented in Section 3.); and on the experience gained from a case study on constructing a language processing tool-set for Swedish in a national project called SVENSK (Section 4.). See (Olsson, 2002) for an elaboration on the requirements specification and design of an open architecture for information refinement.

2. The notion of information refinement

By the term information refinement, the *process* is referred to in which text is handled with the aim of *accessing* the pieces of content that are relevant from a certain *perspective* (Olsson et al., 2001).

Information access is about providing people with different tools and methods for granting reliable and simple access to the information they need, ideally with awareness of task and context of the access situation. A system for information access is intended to see to an expressed information need. Such a need is not always static — the *process* of searching for information is a dynamic one in which the information need, sources of information, characteristics of the task, and the type of text involved may change during a search session.

Since different readers have different information needs, prerequisites, and attitudes, they also have different *perspectives* when reading one and the same text. When considering that there are different perspectives, it is natural to think of information access and refinement systems as something that should not (only) deliver texts in their entirety, but rather in some sense understand the contents of the text and tailor the information according to the reader's perspective.

Information exctraction, information retrieval and automatic summarisation are all examples of human language techniques that fall under the information refinement category. Current work concerning information refinement at SICS include protein name tagging (Eriksson et al., 2002), information access using mobile services (Hulth et al., 2001), and support of professionals in information seeking (Hansen and Järvelin, 2000).

3. Some important HLT projects

This section introduces some of the software and projects that have, or have had, impacts on the ways today's software for HLT is designed and implemented. The survey of the literature in the area is not exhaustive, but merely provides an overview of the aspects and features of some important projects.

3.1. The TIPSTER architecture

The TIPSTER project (Grishman et al., 1997) was a joint effort between a number of U.S. government agencies led by DARPA and funded by CIA, DARPA, and DoD in collaboration with NIST and SPAWAR. The project started in 1991 and ended due to lack of funding in 1998.

The main focus of TIPSTER was to improve document processing efficiency and cost effectiveness, and in doing that, technologies such as information retrieval, information extraction, and automatic text summarisation were of great interest. There were two primary goals of the TIPSTER project, the first of which was to provide developers and users with an architecture that allowed for information retrieval in several gigabytes of texts, and the second goal was to provide an environment for research in document

detection and data extraction. However, by the time the project was discontinued, no fully implemented version of the TIPSTER architecture was produced.

3.2. CLE

SRI International's Cambridge Research Centre and Cambridge University's Computer Laboratory in 1985 suggested a UK-internal project developing a Core Language Engine (CLE), a domain independent system for translating English sentences into formal representations (Moore and Jones, 1985; Alshawi et al., 1992).

SRI's CLE built on a modular-staged design in which explicit intermediate levels of linguistic representation were used as an interface between successive phases of analysis. The CLE has been applied to a range of tasks, including machine translation and interfacing to a reasoning engine. Smith (1992) gives two examples of such systems; the LF-Prolog Query Evaluator and the Order Processing Exemplar (OPEX). The modular design also proved well suited for porting to other languages and the implementation was quite efficient. Thus, the project proved its purpose. However, even though the CLE system received considerable attention, it failed to spread in the community, the main reason being that it simply was too expensive to obtain it.

3.3. ALEP

The origin of the Advanced Language Engineering Platform (ALEP), the work on which started in 1991 and ended in 1995, was the issue of the lack of a general platform for research and development of large scale natural language processing systems (Simpkins, 1995; Bredenkamp et al., 1997). ALEP was an initiative of the Commission of the European Community (CEC) based on the experiences from the Eurotra and CLE projects.

ALEP was intended to function as a catalyst for speeding up the process of going from a research prototype of a system to a ready-to-ship product. The kind of users that ALEP first and foremost was targeted at were advanced experts, i.e., researchers in computational linguistics, possibly in conjunction with application developers. Simpkins (1995) expected that the openness of ALEP would attract users for research and development. Later, it turned out that this was not the case and ALEP never became widely spread.

3.4. GATE

Since the mid 90's, the General Architecture for Text Engineering (GATE) platform as reported on by, e.g., Cunningham (2000) is being developed at the University of Sheffield and funded by the U.K. Engineering and Physical Sciences Research Council (EPSRC). GATE provides a communication and control infrastructure for linking together language engineering software. It does not adhere to a particular linguistic theory, but is rather an architecture and a development environment designed to fit the needs of researchers and application developers. GATE, currently available as version 2.0, is free for non-commercial and research purposes.

GATE supports reuse of resources, data as well as algorithms, since it provides for well-defined application programmers interfaces (APIs). Once a module has been integrated in the system, it is very easy to combine it with already existing modules to form new systems. Each component integrated into GATE has a standard I/O interface, which conforms to a subset of the TIPSTER annotation model. The infrastructure of GATE provides several levels of integration, reflecting how closely a new module should be connected to the core system.

3.5. The DARPA Communicator

Currently, the MITRE Corporation is (under DARPA funding) developing the DARPA Communicator. The goal of the DARPA Communicator is to set the scene for the next generation of conversational, multi-modal, interfaces to distributed information to be used in, e.g., travel planning, that require information from different sources to be combined.

The reference DARPA Communicator architecture builds on MIT's Galaxy-II system (Polifroni and Seneff, 2000; Seneff et al., 1999; Seneff et al., 1998). Among its key features, the authors list the ability to control system integration using a scripting language: each script includes information about the active servers, a set of operations supported by the server, as well as a set of programs. An in-depth explanation of the program control is given by Seneff et al. (1999). Essentially, the Galaxy-II system builds on a central process, the Hub, which mediates information between a number of different servers. The Galaxy-II system supports a wide range of component types, e.g., language understanding and generation, speech recognition and synthesis, dialogue management, and context tracking (Goldschen and Loehr, 1999).

There is a freely available, public version of the core DARPA Communicator.

3.6. ATLAS

The Architecture and Tools for Linguistic Analysis Systems (ATLAS) project is conducted by NIST, MITRE and LDC (Bird et al., 2000). The main goal is to develop a general architecture for annotation of linguistic data, including a formal/logical data format, a set of APIs, a tool-set, and persistent storage.

Within the ATLAS project, the participants are mainly interested in creating a formal framework for constructing, maintaining, and searching in linguistic annotations. In some aspects, the ATLAS annotation set model seems very similar to the TIPSTER annotation scheme. Bird and Liberman (2000) say that there are several ways of translating a TIPSTER-style annotation to a corresponding ATLAS one. In the end, the ATLAS working group concludes that TIPSTER-like annotations are not appropriate for audio transcriptions, except for "cases where such transcriptions are immutable in principle", (Bird and Liberman, 2000).

4. A case study — SVENSK

The SVENSK project was a national effort funded by the former Swedish National Board for Industrial and Techni-

cal Development (Nutek) and SICS addressing the problem of reusing language engineering software, see e.g., (Eriksson and Gambäck, 1997; Gambäck and Olsson, 2000). The SVENSK project was divided into three phases, spanning the spring of 1996 to the end of 1999. The aim has been to develop a multi-purpose language processing system for Swedish based, where possible, on existing components. Rather than building a monolithic system attempting to meet the needs of both academia and industry, the project has created a general tool-box of reusable language processing components and resources, primarily targeted at teaching and research.

The re-usability of the language processing components in SVENSK system arises from having each component integrated into GATE.

Collecting and distributing algorithmic resources and making different programs inter-operate present a wide range of challenges, along several different dimensions outlined next.

4.1. Diplomatic challenges

Making language processing resources freely available and, in particular, re-usability of resources is really a very uncommon concept in the computational linguistic community. Possibly this also reflects another uncommon concept, that of experiment reproducibility. In most research areas the possibility for other researchers to reproduce an experiment is taken for granted. It is even considered as the very core of what is accepted as good research at all. Strangely enough, this is seldom the case in computer science in general and even more rare within computational linguistics, perhaps because of tradition or lack of interest.

4.2. Technical challenges

From the technical point of view, one major conclusion is that the difficulties of integrating language processing software never can be over-estimated. Even when using a liberal architecture such as GATE it is hard work making different pieces of software from different sources and built according to different programming traditions meet any kind of interface standard.

In a way, it is understandable that academia does not always put much effort in packaging and documenting their software, since their main purpose is not to sell and widely distribute it. More surprising and discouraging, however, is that some of the actors on the commercial scene do not document their systems in a proper manner, either. Far too often this has resulted in inconsistencies with the input and output of other modules.

4.3. Linguistic challenges

Of course, language engineering components differ with respect to such things as language coverage, processing accuracy and the types of tasks addressed. It is also the case that tasks can be carried out at various levels of proficiency. The trouble is that there is no quality control available neither to the tool-box developer nor to the end-user. If a large number of language processing components are to be integrated, they should first be categorised so that components with a great difference in, say, lexical coverage are not combined.

A familiar problem for all builders of language processing systems relates to the adaptation to new domains. When reusing resources built by others this becomes even more accentuated, especially if a language engineering resource is available only in the black-box form (and thus relates to the issues of the previous subsection).

5. General observations and experiences

Below are some broad conclusions — focal points — drawn from the previous and present chapters, of what should be considered when creating a general HLT architecture:

1. **An architecture should be general with respect to a class of tasks, not to an entire field of research** The issue of *how* general an architecture should be needs to be considered since a too general one tends to be hard to handle.

2. **Keep the software open** There are various dimensions along which software could be considered open: distributing and licensing it; keeping its source open and inviting other people to participate in developing it; and to achieve software that are easily adaptable to new domains and types of information.

3. **Allow for use of existing programs as well as for the creation of system-specific ones** The potential drawback in using existing, externally produced software concerns issues such as, e.g., maintenance, fixing bugs, and extending/updating resources such as lexica and ontologies. All these things rely on the external program being supported by its producer.

4. **Support maintenance of systems and the components making them up** Develop tools and methods to support maintenance of components and systems, both on the linguistic level, e.g., integrated machine learning methods for lexical acquisition and grammar induction, and on the software level, e.g., new file formats and operating systems.

6. Motivation for a new architecture

The motivation for building a new architecture is primarily due to the fact that when information refinement emerged as a research area at SICS, there was no single architecture which fulfilled the demands that SICS's projects made at the time. In particular, no one of the existing platforms granted us full access to the source code and full distributional rights of the code, something which would be of great interest to us since we wanted to be able to distribute the source code of future information refinement systems freely, and since the functionality of the tools used for information refinement will have to be tuned to each new information refinement task. The latter may include changes to, e.g., the way the tools interact with each other and with the user, as well as the kind of data they produce — such changes may be difficult to achieve unless the software architecture hosting the tools is accessible at the source code level.

The work on a new HLT architecture called Kaba is an ongoing effort which was initiated in 1999 by Kristofer Franzén and Jussi Karlgren at SICS. At first, Kaba was intended to constitute an information extraction system for Swedish. An attempt at porting an existing information extraction system from English to Swedish turned out to be cumbersome (Franzén, 1999). Along the above lines, the conclusion was reached that future research in information refinement at SICS would benefit from a research vehicle having been built on site. Since 1999, the research focus has shifted slightly from information extraction to the more general goal of information refinement, which makes the need for an open and general architecture even clearer.

7. Requirement analysis

Deciding on what requirements are relevant for a given project tends to be a top-down process, going from broad issues such as, e.g., that the software under development should be portable to new operating systems, to splitting the portability into more specific sub-requirements. Requirements analysis always asks the *what*-questions regarding the software, e.g.: *what* equipment constraints exist, and *what* functions are to be incorporated. The *how*-questions are issued in the design phase described in Section 8..

Kaba is intended to function as a tool for developers of information refinement systems, first and foremost for research systems, but also for prototypes for testing ideas within information refinement. Kaba will *not* be a fixed set of tools for creating ready-to-ship products.

A typical Kaba user is a computational linguist with programming skills. This person's role is to use Kaba for the creation of information refinement systems to be used further in research and prototyping.

7.1. Project constraints and external factors

To accomplish the portability of Kaba on the software level, a widely supported programming language, such as Java, has to be used throughout the development process to implement all parts of the architecture. Further, Kaba will require (and presuppose) a linguistic processor that performs basic linguistic analysis of the texts to be processed, e.g., part-of-speech tagging and some fundamental grammatical analysis. Most likely the processor will be the Swedish and the English Functional Dependency Grammars (FDG) from Conexor Oy, Helsinki, Finland (Tapanainen and Järvinen, 1997).

Kaba must be implemented using a technology and an environment that facilitates easy integration of in-house or third party software for linguistic analysis as well as basic computational facilities, e.g., for reading and writing various file formats.

7.2. The scope of the work

Figure 1 shows three different ways that an information refinement system based on Kaba can interface with its environment, and thus gives some notion of what a developer of such a system has to deal with. What differs between the three constellations is the kind of user the system is intended for. In Figure 1 A, the system interacts with an information provider of some sort, e.g., a web site, a database,

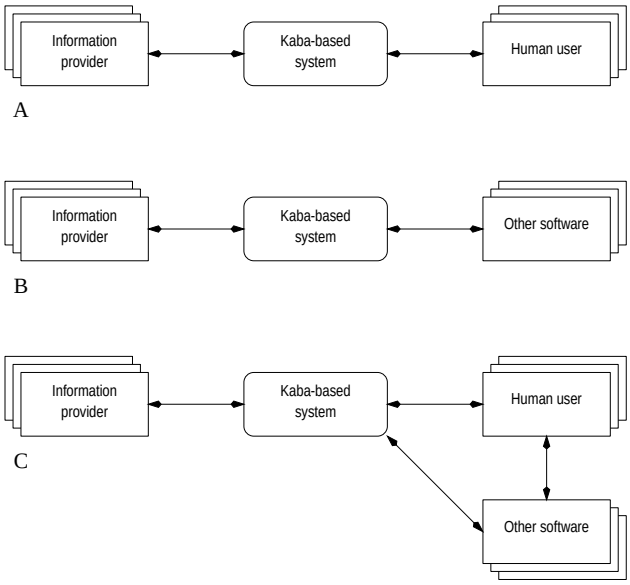


Figure 1: Characteristics of the environment of a Kaba-based system.

or a mobile service, on the one hand, and a human user on the other.

In Figure 1 B, the Kaba-based system communicates with the same kind of information provider as in Figure 1 A, but with another machine as counterpart instead of a human. The setup illustrates the case when a Kaba-based system is part of a larger system.

Finally, Figure 1 C, shows a configuration in which the system interacts with a human user as well as another machine.

7.3. The scope of the architecture

When starting to look at what a user may want to do with Kaba, it seems as a good idea to structure the requirements into what is commonly known as use cases (UC). Cockburn (1997) gives an overview of a method that deals with the identification and structuring of UCs. He defines a use case as being what happens when *actors* interact with a system to achieve a desired goal. An actor is an external entity (human or other software) that uses the system. In effect, UCs hold the functional requirements of a system in an easy-to-read format, and they represent the goal of an interaction between an actor and the system.

In total, 30 use cases have been identified for Kaba and seven of these constitute the top level of the use case hierarchy (Olsson, 2002):

UC 1: Develop an information refinement research and development prototype system.

UC 2: Evaluate an information refinement research and development prototype system.

UC 3: Port an existing system to a new domain or language.

UC 4: Document system.

UC 5: Maintain system.

UC 6: Create learning material or tutorial.

UC 7: Manage LR and PR components.

Use cases 1 and 7 each have several sub-goals which are illustrated in Figure 2 and Figure 3, respectively.

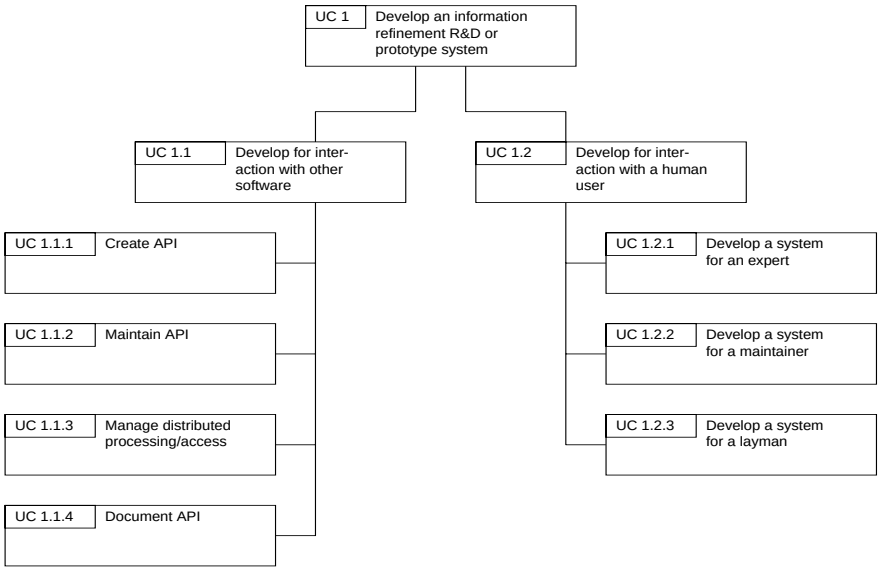


Figure 2: Schematic view of use case 1 and its sub-goals.

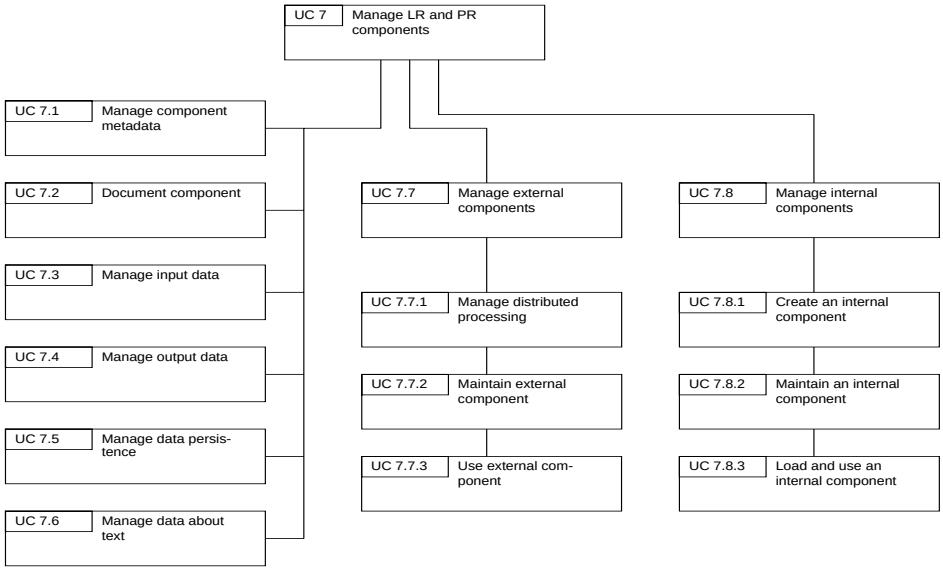


Figure 3: Schematic view of use case 7 and its sub-goals.

8. Design proposal

The design proposal is intended to give a hint as to how the requirement analysis could be realised.

8.1. Component metadata

This section covers use case 7.1 (*Manage component metadata*). Metadata about both language resources (LR) and processing resources (PR) is needed for several reasons, the first of which is to allow the developer (and the future users of the system) to browse a collection of components to see what components there are in order to build an information refinement system utilising existing components. In the same manner, metadata can be used to identify shortcomings of existing components and act as a basis for requirements analysis and specification when new language processing components need to be constructed or when new language resources need to be developed.

There are several means by which metadata can be expressed, and it seems natural to convey such data in the same format as the components themselves are annotated or produce annotations about text. Thus, the system internal format of metadata should correspond to the internal format of the data about text as described in Section 8.3., while the external format of metadata should agree with the format for data persistence described in Section 8.4..

8.2. Input and output

This section deals with use cases 7.3 (*Manage input data*) and 7.4 (*Manage output data*). The Kaba information refinement development platform presupposes that some sort of linguistic analysis has been performed on the text to be processed by a Kaba-based system. Currently, the FDG for English and Swedish are intended to be used, but it should also be possible to use any TIPSTER compliant linguistic processing component.

On the output side, a Kaba-based system should be able

to generate representations of the text it has processed in a format suitable to the user, regardless of whether the user is another computer program or a human.

8.3. System internal representation of annotated text

This section covers use case 7.6 (*Manage data about text*). Data about text can be expressed in various ways and the crucial point in all data representation is that it should facilitate rapid access to arbitrary pieces of information about the text. The representation formalism should allow for scaling up without causing the system's performance to drop.

While the format of the external and persistent data is like XML (see Section 8.4.), the internal representation is based on the TIPSTER annotation scheme. Although the two schemes are conceptually different the conversion between TIPSTER-style annotations and XML-based representations is quite straightforward.

8.4. Data persistence

This section deals with use case 7.5 (*Manage data persistence*). Data persistence is needed in order to provide Kaba with multiple-session capabilities, that is, to allow a user to work with the same source of information during several sessions and, in each session, having access to the results from the previous ones. The need for working in multiple sessions may occur, e.g., due to a system crash, for saving intermediate results, or simply because the user needs to interrupt the refinement process for other reasons.

The most suitable format is likely to be some instance of XML, partially because of the fact that it is becoming increasingly widespread in language engineering applications, and partially because there exist tools for manipulating and converting between different instantiations of XML.

8.5. Interacting with others

There are several aspects of interaction which have to be taken into account when designing an information refinement architecture like Kaba: (1) when a Kaba-based system is used by other software as a part of a larger system, (2) when a Kaba-based system utilises external components, both processing and data, as a part of an information refinement system, and (3) when a Kaba-based system needs to interact with human users.

Case (1) is reflected in use case 1.1.1 (*Create API*). In effect, what is required for a Kaba-based system to function in the context of a larger system, is a means for the developer of the larger system to have access to a restricted and well-defined set of the functionality in the Kaba-based system. Such access can be provided by means of a Java API.

Case (2) is addressed in use case 7.7.3 (*Use external component*) which concerns how to allow a Kaba-based system to use external components, i.e., components not primarily implemented for use within Kaba such as, for instance, part-of-speech taggers and ontologies. To allow Kaba to interact with external components, it is important that the components all look the same from Kaba's point of view. This means that the APIs that Kaba has to use to achieve this interaction have to be well defined and consistent.

Case (3) is addressed in use cases 1.2.1 (*Develop a system for an expert*), 1.2.2 (*Develop a system for a maintainer*), 1.2.3 (*Develop a system for layman*), all of which aim at facilitating interaction between different kinds of end-users and a Kaba-based system. Case (3) boils down to creating a connection between a tool or library for constructing GUIs, such as the Java Swing Classes (Topley, 1998), and Kaba.

8.6. Distributed processing

This section addresses use cases 1.1.3 (*Manage distributed processing/access*) and 7.7.1 (*Manage distributed processing*). In various settings, the parts making up a Kaba-based system need to be situated on different machines, connected by a network. One such setting occurs when some component, for example the one providing the initial linguistic analysis of input text, is available only for a particular operating system, while the rest of the system runs on another machine in the network. The different parts of the system then have to communicate using some protocol, e.g., SOAP.

8.7. Documentation and tutorials

This section addresses use cases 1.1.4 (*Document API*), 4 (*Document system*), 6 (*Create learning material or tutorial*), and 7.2 (*Document component*).

Kaba should come with incentives for developers, both of the Kaba architecture itself and of Kaba-based systems, to document their efforts. Such stimulus should be in the form of guide-lines and examples. There is a range of possible formats for documenting software systems, e.g., HTML and plain ASCII. It is also important that the guidelines are tied as little as possible to the chosen format. As for documenting the source code, existing tools such as Javadoc should be used.

Examples and tutorials should be encouraged by providing templates, example examples and tutorials to Kaba users and system developers.

8.8. Creating internal components

This section deals with use case 7.8.1 (*Create an internal component*). In Kaba, an internal component is one that is under the control of the developer in that it provides him with a more elaborate API than external components do. Typically, an internal component is created explicitly for use within a Kaba-based system.

A variant of the Common Pattern Specification Language (CPSL) called Kaba Pattern Specification Language (KPSL) will form the base formalism in which the functionality of the internal components will be expressed. CPSL is an effort by the TIPSTER working group that, unfortunately, has not been officially released. However, Appelt (1999) as well as Cunningham et al. (2000) present implementations of annotation engines based on CPSL. Essentially, a CPSL rule describes a finite state transducer for TIPSTER annotations.

It should be possible to construct internal components in several ways, for instance by hand-crafting rules using

a graphical rule editor, or by breeding them using machine learning methods.

8.9. Loading and using internal components

This section deals with use case 7.8.3 (*Load and use an internal component*). Once the KPSL rules making up a component have been developed, they are turned into Java code by a KPSL rule compiler. Along with the compiler come Java classes that facilitate dynamic loading of compiled sets of rules. Thus, as long as the KPSL rules have been compiled to Java and the Kaba-based system knows where to find the components, there are means by which they can be dynamically loaded into the system at run time.

8.10. Maintenance

The fundamental question when it comes to maintenance of any software is *When is maintenance necessary for this piece of software in this particular setting?* and, in the context of information refinement systems, this calls for well-defined criteria that can be used to probe the system's performance with respect to the task it is supposed to accomplish, or the system's affordance with respect to the users' expectations as to what the system is really supposed to do.

8.10.1. Maintenance of external components

This section addresses use cases 1.1.2 (*Maintain API*) and 7.7.2 (*Maintain external component*). The cases are closely related in that communication between a Kaba-based system and other software will always take place via some kind of API. Thus, maintaining an external component is in many cases the same as maintaining the API that Kaba uses for communicating with that component.

8.10.2. Maintenance of internal components

This section deals with use case 7.8.2 (*Maintain an internal component*). Maintenance of internal components should be facilitated by a graphical interface for inspecting, editing, loading, executing, and evaluating KPSL rules with respect to some success criteria set up for the component. It should be possible to do all this using the same, or a similar, graphical interface as when creating internal components.

8.10.3. Maintenance of systems

This section deals with use case 5 (*Maintain system*) which involves all other kind of maintenance mentioned previously in this section, i.e., maintenance of component APIs and external components (Section 8.10.1.), as well as of internal components (Section 8.10.2.). In addition, maintenance of systems also involves taking care of the whole formed by the pieces, e.g., seeing to it that the documentation is up to date, installing new software when needed, and monitoring the system's performance on a regular basis. This should be supported in the same way as maintenance of external components is, e.g., by giving guidelines for how to integrate the documentation of the parts into a central repository, and collect information about availability of new components.

8.11. Providing support for porting systems to new domains

This section deals with use case 3 (*Port an existing system to a new domain or language*). While maintenance may accommodate correction of minor changes to a system, there will also be occasions when the shift of domain or information need is so different from that captured by an existing system that maintenance of the system or one of its components is not enough to compensate for it. In these cases, the question of whether to use an existing system or to create a new one from scratch arises. One of the issues of providing support for porting systems to new domains and needs should be to supply the developer with clues for deciding the answer to that question. If the answer is that an existing system could probably be altered (ported) to meet the new needs, then the follow-up question should be: *What parts of the existing system can be re-used, and to what degree do they need to be modified?* Again, Kaba should provide methods that makes answering this question easier.

8.12. Providing support for evaluation

This section addresses use case 2 (*Evaluate an information refinement R&D or prototype system*). Evaluation of information refinement systems is a crucial issue in several aspects. The basic support for evaluation of information refinement systems can be of two kinds: by providing linguistically annotated data that act as a key to the questions for which a system is to be evaluated, or by providing tools that presuppose the presence of an answer-key for comparing data structures and calculating measurements of performance. Both kinds of support are necessary. In the former case, machine learning methods are often used as an aid in obtaining the correctly annotated corpora constituting the answer-key. In the latter case, the comparison of data structures should yield values in an appropriate metric, e.g., precision and recall, depending on the features that are evaluated.

9. Conclusions

When developing a general tool or architecture, it is possible to focus the technical and linguistic efforts in several ways. The most obvious one is to formulate and maintain an explicit goal regarding the kind of tasks that programs developed within the general architecture at hand should cope with. By obtaining and focusing on the goal at an early stage in the development of the open architecture, one can avoid ending up with a definition and design of a far too general system: when it comes to generality for language engineering, it should be with respect to a class of tasks, rather than to the field as such.

Acknowledgements

Lars Borin, Björn Gambäck, Kristofer Franzén, Jussi Karlgren, Preben Hansen, Gunnar Eriksson, Mikael Eriksson, Anette Hulth, Mark Tierney, Anna Jonsson.

10. References

Hiyan Alshawi, David Carter, Jan van Eijck, Björn Gambäck, Robert C. Moore, Douglas B. Moran, Fernando

C. N. Pereira, Stephen G. Pulman, Manny Rayner, and Arnold G. Smith. 1992. *The Core Language Engine*. MIT Press, Cambridge, Massachusetts, March.

Douglas E. Appelt, 1999. *The Complete TextPro Reference Manual*, June.

Steven Bird and Mark Liberman. 2000. A Formal Framework for Linguistic Annotation. *Speech Communication*, 33(1,2):23–60.

Steven Bird, David Day, John Garofolo, John Henderson, Christophe Laprun, and Mark Liberman. 2000. ATLAS: A flexible and Extensible Architecture for Linguistic Annotation. In *Proceedings of the Second International Conference on Language Resources and Evaluation*, pages 1699–1706, Athens, Greece, June.

Andrew Bredenkamp, Thierry Declerck, Frederik Fouvry, Bradley Music, and Axel Theofilidis. 1997. Linguistic Engineering using ALEP. In R. Mitkov and N. Nicolov, editors, *Proceedings of the 2nd International Conference on Recent Advances in Natural Language Processing*, pages 92–97, Tzigov Chark, Bulgaria, September.

Alistair Cockburn. 1997. Structuring Use Cases with Goals. *Journal of Object-Oriented Programming*, Sep-Oct and Nov-Dec.

Hamish Cunningham, Diana Maynard, and Valentin Tablan. 2000. JAPE: A Java Annotation Patterns Engine. Technical Report CS–00–10, University of Sheffield, Department of Computer Science, Sheffield, UK. Second Edition.

Hamish Cunningham. 2000. *Software Architecture for Language Engineering*. Ph.D. thesis, University of Sheffield, UK.

Mikael Eriksson and Björn Gambäck. 1997. SVENSK: A Toolbox of Swedish Language Processing Resources. In R. Mitkov and N. Nicolov, editors, *Proceedings of the 2nd International Conference on Recent Advances in Natural Language Processing*, pages 336–341, Tzigov Chark, Bulgaria, September.

Gunnar Eriksson, Kristofer Franzén, Fredrik Olsson, Lars Asker, and Per Lidén. 2002. Exploiting Syntax when Detecting Protein Names in Text. In *Proceedings of Workshop on Natural Language Processing in Biomedical Applications*, Nicosia, Cyprus, March.

Kristofer Franzén. 1999. Adapting an English Information Extraction System to Swedish. In *Proceedings of the 12th Nordic Conference of Computational Linguistics*, pages 57–65, Norwegian University of Science and Technology, Trondheim, Norway, December.

Björn Gambäck and Fredrik Olsson. 2000. Experiences of Language Engineering Algorithm Reuse. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation*, volume 1, pages 161–166, Athens, Greece, May. ELRA.

Alan Goldschen and Dan Loehr. 1999. The role of the DARPA Communicator Architecture as a Human Computer Interface for Distributed Simulations. In *Spring Simulation Interoperability Workshop*, Orlando, Florida, USA, March. Simulation Interoperability Standards Organization (SISO).

Ralph Grishman, Ted Dunning, Jamie Callan, Bill Caid, Jim Cowie, Louise Guthrie, Jerry Hobbs, Paul Jacobs, Matt Mettler, Bill Ogden, Bev Schwartz, Ira Sider, and Ralph Weischedel, 1997. *TIPSTER Text Phase II Architecture Design. Version 2.3*. New York, New York, January.

Preben Hansen and Kalervo Järvelin. 2000. The Information Seeking and Retrieval Process at the Swedish Patent and Registration Office. Moving from Lab-based to Real Life Work-task Environment. In *Proceedings of the ACM-SIGIR 2000 Workshop on Patent Retrieval*, pages pp. 43–53, Athens, Greece, July 28.

Anette Hulth, Fredrik Olsson, and Mark Tierney. 2001. Exploring Key Phrases for Browsing an Online News Feed in a Mobile Context. In *Proceedings of Management of uncertainty and imprecision in multimedia information systems*, Toulouse, France, September. A workshop held in conjunction with the Sixth European Conference on Symbolic and Quantitative Approaches to Reasoning with Uncertainty (ECSQARU-2001).

Robert C. Moore and Karen Sparck Jones. 1985. A research programme in natural language processing. CRC technical report, SRI International, Cambridge, England.

Fredrik Olsson, Preben Hansen, Kristofer Franzén, and Jussi Karlgren. 2001. Information Access and Refinement — A Research Theme. *ERCIM News*, 46, July.

Fredrik Olsson. 2002. *Requirements and Design Considerations for an Open and General Architecture for Information Refinement*. Licentiate of philosophy thesis, Department of Linguistics, Uppsala University, Uppsala, March. Available at `http://www.sics.se/~fredriko/lic`.

Joseph Polifroni and Stephanie Seneff. 2000. Galaxy-II as an Architecture for Spoken Dialogue Evaluation. In *Proceedings of the Second International Conference on Language Resources and Evaluation*, Athens, Greece, May. ELRA.

Stephanie Seneff, Ed Hurley, Raymond Lau, Christine Pao, Philipp Schmid, and Victor Zue. 1998. Galaxy–II: A Reference Architecture for Conversational System Development. In *Proceedings of the 5th International Conference on Spoken Language Processing*, volume 3, pages 931–934, Sydney, Australia, December.

Stephanie Seneff, Raymond Lau, and Joseph Polifroni. 1999. Organization, Communication, and Control in the Galaxy-II Conversational System. In *Proceedings of Eurospeech 99*, Budapest, Hungary, September.

Neil K. Simpkins. 1995. ALEP — An Open Architecture for Language Engineering. Technical report, Cray Systems, 151 rue des Muguets, L-2167 Luxembourg.

Arnold Smith. 1992. The CLE in Application Development. In Hiyan Alshawi, editor, *The Core Language Engine*, chapter 12, pages 235–250. MIT Press, Cambridge, Massachusetts, USA, March.

Pasi Tapanainen and Timo Järvinen. 1997. A Non-Projective Dependency Parser. In *Proceedings of the 5th Conference of Applied Natural Language Processing*, Washington, D.C. USA, April. ACL.

Kim Topley. 1998. *Core — Java Foundation Classes*. Prentice Hall PTR Core Series. Prentice Hall.

Taking Advantage of Spanish Speech Resources to Improve Catalan Acoustic HMMs

Jaume Padrell and José B. Mariño

TALP Research Center
Universitat Politècnica de Catalunya, Barcelona 08034, Spain
{jaume,canton}@talp.upc.es

Abstract

At TALP, we are working on speech recognition of official languages in Catalonia, i.e. Spanish and Catalan. These two languages share approximately 80 % of their allophones. The speech databases that we have available to train HMMs in Catalan have a smaller size than the Spanish databases. This difference of size of training databases results in poorer phonetic unit models for Catalan than for Spanish. The Catalan database size is not enough to allow correct training of more complex models like triphones. The aim of this work is to find segments in Spanish databases that, used in conjunction to the Catalan utterances to train the HMM models, get an improvement of the speech recognition rate for the Catalan language. To make this selection, the following information is used: the distance between the HMM which are trained separately in Spanish and Catalan, and the phonetic attributes of every allophone. A contextual acoustic unit, the demiphone, and a state tying approach are used. This tying is done by tree clustering, using the phonetic attributes of the units and the distances between the HMM states. Different tests have been carried out by using different percentage of tied states in training simultaneously in Catalan and Spanish. In this way, Catalan models are obtained that give generally better results than the models trained only with the Catalan utterances. However, we observe from one of the tests that, when the number of gaussians is increased, that improvement becomes a loss of performance. Currently, we are working on the inclusion of additional labels to avoid that tree clustering puts in the same pool phoneme realizations that are too much different.

1. Introduction

It is not strange to develop in a same laboratory speech recognition systems for different languages. In the TALP we work in official languages of Catalonia, the Spanish and the Catalan. These two languages share approximately 80% of their allophones (both come from the Latin and share geographic space).

The available speech databases to train the HMMs in Catalan have an smaller size that the available databases in Spanish.

This difference in the dimensions of the training databases is one of the causes by which poorer estimations of the Catalan phonetic units are obtained than in Spanish. Thus, whereas in Spanish the units with what we obtain higher recognition rates are triphonemes or demiphones, in Catalan the best results are obtained modeling allophones (Pachès, 1999), 3 states CDHMM with 32 Gaussian for state, since the database size is not sufficient to allow correct train of more complex models like triphones.

In other works (Mariño et al., 2000b), for bilingual recognition systems with these two languages that should work with limited resources (memory, time, etc.), a set of bilingual HMMs has been created (modeling demiphones) that share some models for both languages. These models, trained with utterances from both languages, obtain a lower recognition rate than their respective monolingual models, but the degradation is not significant.

These last recognition results suggests the possibility of a carefully selection of some utterances from the Spanish databases might the Catalan acoustic HMMs.

(Bonaventura et al., 1997; Wheatley et al., 1994) already suggested the idea to train phoneme models for a language using other languages and implemented a number of different metrics for measuring similarities among cross-language phonetic models. (Bub et al., 1997) considered this task as a question of model adaptation and (Imperl and Horvat, 1999) already used context-depending phonetic units (triphones) in multilingual models.

All these works discus the difficult to select the utterances segments to train the shared models. As a framework to do this selection, in this paper we present some preliminary results using demiphones (Mariño et al., 2000a) as context-depending phonetic units and clustering algorithms that are usually employed to train units that appear little in the training corpus. So, the aim is to use these clustering algorithms to relate contextual units from different languages.

The paper is organized as follows. Section 2. describes the work methodology, section 3. gives some preliminary results and section 4. presents conclusions and future work.

2. Procedure

The procedure that we followed has been: 1) to choose an acoustic unit inventory for both languages, 2) to choose algorithms to select and to tie acoustic units and, 3) to train and to evaluate units for the Catalan with a different percentages of units also trained with the Spanish utterances.

2.1. Spanish and Catalan Allophones Inventory

The allophone transcription is made by different softwares (Saga for the Spanish and Segre for the Catalan) developed in TALP research center. These programs use rules to turn the orthographic text to strings of allophone coded in SAMPA notation.

The transcriptor Saga uses the rules described in (Llisterri and Mariño, 1993) to obtain the phonetic transcription. The program Segre uses extern rules developed in the UAB.

The program Segre transcribes the Catalan sentences using 34 different allophones. In table 1 are shown these

allophones with the attributes[1] that we associated to them. They are used to indicate common characteristics between the units for tree-clustering. These attributes can have phonetic meaning (voiced, manner and point of articulation) or not (for example speaker gender), however, in the present case all attributes have a phonetic meaning.

Al	Attributes	L
a	vowels, open, central, voiced	C,ES
e	vowels, mid_close, front, voiced	C,ES
E	vowels, mid_open, front, voiced	C
i	vowels, close, front, voiced	C,ES
o	vowels, mid_close, back, voiced, rounded	C,ES
O	vowels, mid_open, back, voiced, rounded	C
u	vowels, close, back, voiced, rounded	C,ES
@	vowels, schwa, central, voiced, unrounded	C
j	glides, palatal, semivowel, voiced, close, front	C,ES
w	glides, labial_velar, approximant, voiced, close, back	C,ES
uw	glides, voiced, close, back, rounded	C
p	consonants, bilabial, plosive, voiceless, stop	C,ES
t	consonants, dental, plosive, voiceless, stop	C,ES
k	consonants, velar, plosive, voiceless, stop	C,ES
b	consonants, bilabial, plosive, voiced	C,ES
d	consonants, dental, plosive, voiced	C,ES
g	consonants, velar, plosive, voiced	C,ES
B	consonants, bilabial, approximant, voiced	C,ES
D	consonants, dental, approximant, voiced	C,ES
G	consonants, velar, approximant, voiced	C,ES
f	consonants, labiodental, fricative, voiceless	C,ES
s	consonants, alveolar, fricative, voiceless	C,ES
z	consonants, alveolar, fricative, voiced	C,ES
x	consonants, velar, fricative, voiceless	ES
jj	consonants, palatal, approximant, voiced	ES
T	consonants, interdental, fricative, voiceless	ES
tS	consonants, palatal, affricate, voiceless, mid_palatal	ES
S	consonants, palatal, fricative, voiceless, mid_palatal	C
Z	consonants, palatal, fricative, voiced, mid_palatal	C
y	consonants, palatal, approximant, voiced	C
l	consonants, alveolar, lateral, voiced, liquid, back	C,ES
L	consonants, palatal, lateral, voiced	C,ES
m	consonants, bilabial, nasal, voiced	C,ES
n	consonants, alveolar, nasal, voiced	C,ES
N	consonants, velar, nasal, voiced	C,ES
J	consonants, palatal, nasal, voiced	C,ES
r	consonants, alveolar, tap, voiced, rothics, liquid	C,ES
rr	consonants, alveolar, trill, voiced, rothics, vibrate	C,ES
R	Alveolar, Voiced, Rothics, vibrate	ES

Table 1: Allophone list (All.) that the program Segre (C) and Saga (ES) uses (in SAMPA notation) and attributes that are assigned to each unit.

The program Saga provides the 32 allophones for Spanish language. They are also shown in table 1. These inventory were used in the Spanish SpeechDat database (Moreno, 1997) design.

Between the 32 selected allophones to represent Spanish (ES) and the 34 to represent the Catalan (C), there are 27 allophones (C,ES) that share the same SAMPA notation, for example the vowels /a/, /e/, /i/, /o/ and /u/.

2.2. Shared Training and HMM Distance Measure

In an initial step, we trained units separately for each language. In a second step, we re-estimated the Catalan units using also utterances in Spanish. To do this, we tied the Catalan HMM states and the Spanish ones by tree-clustering (Young et al., 1999) with the separately trained HMMs values and the allophones attributes from table 1.

We sort the HMM states that have been tied in both languages by distances between their values. This will later help us to decide which units are finally shared in the experiments, when we only leave tied a ρ percentage of these HMM states.

The distance measure between two HMMs that we use is described in (Young et al., 1999). This measure is based on the sum of the probabilities that the averages that characterize HMM_1 belong to HMM_2 and vice versa. The probability is evaluated logarithmically so, in the case that HMM_1 and HMM_2 are the same model, we obtain a distance zero among them.

So, we re-estimate the HMMs of both languages jointly. The Catalan HMMs with the Catalan utterances and the Spanish HMMs with Spanish utterances. However, there is a certain ρ percentage of HMMs states shared (or tied) between both languages and therefore, that are trained simultaneously in Catalan and Spanish.

3. Experiments and Results

HTK (Young et al., 1999) software is used to train HMMs and to carry out the speech recognition experiments. A classic parametrization of four characteristics has been used, three of dimension 12 and one of dimension 2, trained respectively with a Mel-Cepstrum with mean subtraction, its first differential, its second differential and, in joint form, second and third energy differential. CDHMM are used to model the acoustic units.

3.1. Spanish and Catalan Speech Databases

3.1.1. Training Data

In order to estimate the HMMs two sets of utterances are used (in both cases the automatic transcription is done without considering coarticulation between words):

- Catalan corpus is formed by 3,981 sentences from the SpeechDat Catalan database (Hernando and Nadeu, 1999), with 639 different speakers (Catalan Eastern dialect) and 171,443 allophones according to the Segre transcription (6.5 hours of speech for training).
- Spanish corpus is formed by 4,951 sentences from 976 different speakers (Spanish speakers from Catalonia) from the SpeechDat Spanish database (Moreno, 1997). This sentences set is formed by 242,813 allophones according to the program Saga (also more than 6 hours of speech for training). This corpus represents a fourth database size than it is available for Spanish.

[1] They were designed, in addition to the TALP members, by the Laboratory of Phonetics from the UAB and by Sílvia Llach from the UG.

3	@ ESa a ESo o O ESe e E											ESi i		ESu u
4	@ ESa a ESo o O						ESe e E					ESi i		ESu u
5	@ ESa a			ESe e E			ESi i		ESo o O					ESu u
8	@	ESa a		ESe e		E	ESi i		ESo o		O			ESu u
13	@	ESa	a	ESe	e	E	ESi	i	ESo	o	O	ESu	u	

Table 2: Vowels clustering by smaller between models distances according to the final maximum number of groups (first column). The Spanish allophones are distinguished from the Catalans by the prefix ES added to its SAMPA representation.

In this paper, we focus our experiments on the ρ percentage of HMMs states tied between both languages. Future research will be addressed to the size database ratios.

Between Catalan and Spanish models we notice 27 allophones that have the same (SAMPA) representation. These are the 94.43% total number of allophones in our Spanish training corpus and the 76.95% in the Catalan one (the difference is mainly due to the allophone schwa /@/ that does not exist in Spanish and has 16.90% frequency of apparition in our Catalan database). On the other hand, the most frequent allophone in our Spanish database (/a/ with 13.04% frequency) is also in Catalan database.

3.1.2. Evaluation Data

The evaluation tests have been carried out with a database with locality names (2,633 sentences with 232 different names with length from one to five words by sentence) and with a people names database (2,956 sentences with 510 different names). All these sentences come from the Vocatel database (Nadeu et al., 1997).

3.2. Allophone Clustering

Allophone models have been trained separately for each language (CDHMM of 3 states with 4 Gaussian for state). These models objective is twofold: first, to do a preliminary analysis of distances that there are between models from both languages and, second, like a departure point for demiphone units training.

To study the correlation between SAMPA representation and HMM distance we clustered the 13 models that represent the vowels set for both languages into 8 models, matching the models that are at smaller distance. In table 2 is shown the clusters that are obtained according to their final maximum number (3, 4, 5 and 8) that are requested to the clustering algorithm. We obtain that each vowel joins with whom shares symbol SAMPA.

The table 3 shows similar experiment but clustering HMM states independently. The clusters are ordered from less distance between HMM states to more. It can be seen that the similarity between models depends on the HMM state.

We created models simultaneously training the allophones that share its SAMPA representation. The experiments gave recognition rates poorer than with the allophones trained only with Catalan utterances. Probably this is due to many shared allophones occur in different allophone contexts in both languages.

In above mentioned work (Mariño et al., 2000b), where demiphones were used for a bilingual recognition system, the degradation was not significant. In order to approach the different allophone context problem we also use the demiphone as acoustic unit, so that the context tied can be better controlled.

Cl. Order	Left State	Middle State	Right State
1	a ESa	a ESa	a ESa
2	i ESi	e ESe	e ESe
3	e E	e ESe E	o ESo
4	@ a ESa	O o	@ a ESa
5	e E ESe	i ESi	i ESi

Table 3: Clustering order depending on distance between HMM states followed by Catalan and Spanish (with prefix ES) vowels.

3.3. Demiphone Clustering

These demiphones which were trained simultaneously in both languages were chosen by tree clustering, using the allophone attributes (table 1) and the distance between the HMMs (we use the tree clustering described in (Young et al., 1999)).

In Catalan, after tree-clustering, are used 1,092 demiphones modeled by CDHMM of 2 states with 1 Gaussian for state. We also use a model for silence and one for the speaker noise, both of 3 states and 1 Gaussian for state. Following the same procedure, in Spanish 852 CDHMM for demiphones are obtained, plus one for silence and one for the speaker noise.

The analysis of the clusters that are obtained tying by trees is complex. First, we obtained different clusters depending on if we tried simultaneously to cluster all states that form a model or make clusters by state. Second, some of the clusters had that we would name phonetic explanations, but others were inexplicable from this point of view.

It is difficult to evaluate which tying improve the recognition in Catalan and which not. Preliminary experiments with our databases seems indicate that tying between some vowels (for example, /e/ /E/ and /ESe/, or /o/ /O/ and /ESo/) worsen the speech recognition in Catalan language.

Several tests have been done operating only on the tying ρ percentage allowed between demiphones pre-tied by tree-clustering with both languages.

In order to have baseline models for the evaluation demiphones CDHMM with only the Catalan utterances have been trained (it is the case of $\rho = 0.00\%$). In the table 4 are shown the different recognition rates that were obtained. In the first column it is indicated ρ, the states percentage for a total HMM sates set of $(2 * 1,092)$ states that

were tied in the training and which, therefore, were trained simultaneously both languages.

ρ(%)	Corr. Names (%)	Corr. Localities (%)
1 Gaussian for state		
0.00	71.28 [69.76,72.80]	85.26 [84.02,86.50]
12.34	71.96 [70.44,73.48]	86.75 [85.51,88.00]
24.95	71.96 [70.44,73.48]	86.06 [84.82,87.30]
34.01	72.63 [71.11,74.15]	85.72 [84.48,87.00]
4 Gaussian for state		
0.00	75.51 [74.00,77.03]	89.21 [88.14,90.28]
34.01	76.52 [75.00,78.04]	87.20 [85.96,88.44]

Table 4: Recognition rates for Catalan sentences depending on the ρ percentage of states trained simultaneously. Between parenthesis there are the probabilities margin with a level of significance of 95 %.

One of the main causes for database people names had worse recognition rate than the site names is that many names are only different by last allophone (due to the gender; for example Francesc for male and Francesc**a** for female) and, in addition, are shorter.

In the results of the table 4 Catalan models with one gaussian for state are obtained that give generally better results using a percentage of bilingual states than the models trained only with the Catalan utterances. However, when we increased the number of Gaussians the recognition improvement becomes a loss of performance for localities database experiment.

4. Conclusion and Future Work

In this paper, we described a method to take advantage of Spanish language speech resources to improve Catalan language acoustic HMMs to speech recognition. We used language as an attribute in the clustering algorithm and CDHMM modeling demiphones. They allowed a better control over the tied allophone context between languages. Further research is need to improve the phonetic transcription and the attributes of these units, for example distinguishing units that at the moment have same symbol SAMPA and, to experiment other types of distances between pdfs, for example the Hellinger distance (Settimi et al., 1999). Our next step will be to carry out experiments increasing the size of the Spanish speech databases and to carry out recognition tests with other tasks, observing the amount of used Spanish material in the training and the test, not only the shared states percentage. Once developed this tying procedure it will be interesting to extend it to other languages that have poor speech databases resources. In our center similar works between dialects of Spanish are being made (Nogueiras et al., 2002).

5. Acknowledgments

This research was supported by CICYT of Spanish government under contract TIC2000-1005-C03-01. The authors would like to thank Climent Nadeu due to his help and encouragement. The authors also wish to acknowledge the access to the VOCATEL database allowed by Telefónica I+D.

6. References

P. Bonaventura, F. Gallocchio, and G. Micca. 1997. Multilingual Speech Recognition for Flexible Vocabularies. In *European Conference on Speech Communication and Technology*, pages 355–358, Rhodes.

U. Bub, J. Köhler, and B. Imperl. 1997. In-Service Adaptation of Multilingual Hidden-Markov-Models . In *International Conference on Acoustics, Speech, and Signal Processing*, pages 1451–54, Münich.

J. Hernando and C. Nadeu. 1999. SpeechDat. Catalan Database for the Fixed Telephone Network. Technical report TALP center, Universitat Politècnica de Catalunya.

B. Imperl and B. Horvat. 1999. The Clustering Algorithm for the Definition of Multilingual Set of Context Dependent Speech Models. In *European Conference on Speech Communication and Technology*, pages 887–90, Budapest.

J. Llisterri and J. B. Mariño. 1993. Spanish Adaptation of SAMPA and Automatic Phonetic Transcription. Technical report, ESPRIT Project 6819, SAM-A/UPC/001/V1, London.

J.B. Mariño, A. Nogueiras, P. Pachès, and A. Bonafonte. 2000a. The Demiphone: an Efficient Contextual Subword Unit for Continuous Speech Recognition. *Speech Communication*, 32(3):187–197.

J.B. Mariño, J. Padrell, A. Moreno, and C. Nadeu. 2000b. Monolingual and Bilingual Spanish-Catalan Speech Recognizers Developed from Speechdat Databases. In *Workshop on Developing Language Resources for Minority Languages: Reusability and Strategic Priorities, LREC*, pages 57–61, Athens.

A. Moreno. 1997. SpeechDat Spanish Database for Fixed Telephone Network. Technical report, SpeechDat Project LE2-4001.

C. Nadeu, J. Padrell, and A. Febrer. 1997. Diseño de la Base de Datos Vestel y Preparación de la Captura. Technical report, Projecte VOCATEL (Telefónica I+D), Universitat Politècnica de Catalunya.

A. Nogueiras, M. Caballero, and A. Moreno. 2002. Multi-Dialectal Spanish Speech Recognition. In *International Conference on Acoustics, Speech, and Signal Processing*, Orlando.

P. Pachès. 1999. *Improved modelling for robust speech recognition*. Ph.D. thesis, Universitat Politècnica de Catalunya.

Raffaella Settimi, J.Q. Smith, and Ali S. Gargoum. 1999. Approximate Learning in Complex Dynamic Bayesian Networks. In *Fifteenth Conference on Uncertainty in Artificial Intelligence*, Stockholm.

B. Wheatley, K. Kondo, W. Anderson, and Y. Muthusamy. 1994. An Evaluation of Cross-Language Adaptation for Rapid HMM Development in a New Language. In *International Conference on Acoustics, Speech, and Signal Processing*, pages 237–40, Adelaine.

S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev, and P. Woodland. 1999. *The HTK Book, Version 2.2*. Entropic.

Portability Issues of Text Alignment Techniques

António Ribeiro*, Gabriel Lopes* and João Mexia+

Universidade Nova de Lisboa
Faculty of Sciences and Technology, Department of *Informatics/+Mathematics
Quinta da Torre, Monte da Caparica, P–2829–516 Caparica, Portugal
{ambar, gpl}@di.fct.unl.pt

Abstract

Much of the work on parallel texts alignment tries to push the boundaries of language independence as far as possible. This has been a trend since the first approaches on sentence alignment in the early 1990s. In this paper we discuss portability issues of parallel texts alignment techniques. How language independent can they be? We examine several alignment techniques proposed by previous authors, discuss how far they went with language independent methodologies, why some authors decided to add linguistic knowledge to their systems and what improvements they attained by doing it. We will also discuss some methodologies and the problems faced by systems which aim at extracting Translation Equivalents from aligned parallel texts.

1. Introduction

Text alignment techniques aim at identifying automatically correspondences between *parallel texts*, i.e. either correspondences between text segments, words or even sequences of characters. Parallel texts are sets of texts which are translations of each other in different languages, like the proceedings of the European Parliament, which is published in the eleven official languages[1] of the European Union – the Official Journal of the European Communities –, or the proceedings of the Canadian Parliament which is published in both English and French – the Canadian Hansards.

Much of the work on parallel texts alignment tries to push the boundaries of language independence as far as possible, i.e. by not using language specific knowledge for the alignment process. This has been a trend since the first approaches on sentence alignment in the early 1990s (Kay and Röscheisen, 1993; Brown *et al*., 1991; Gale and Church, 1991). Still, some authors have resorted to adding some linguistic knowledge in order to improve the alignment results, either by adding short bilingual dictionaries to bootstrap the alignment process (Wu, 1994; Melamed, 1999) or by using word similarity measures to find similar words automatically (Simard *et al*., 1992; Melamed, 1999).

Many authors have tried not to feed linguistic knowledge to their alignment systems, in particular, short bilingual dictionaries, since this makes them easily language dependent and, consequently, hardly portable to other languages. Also, those dictionaries may be incomplete and outdated. In addition, they usually do not provide all the possible word variants due to possible language inflection. Moreover, linguistic knowledge may be expensive to get, may require much time to compile, may be hard to get especially for minority languages or languages for which there are not much linguistic resources available (as in African languages).

In this paper we discuss portability issues of parallel texts alignment techniques. How language independent can they be? We examine several alignment techniques proposed by previous authors, discuss how far these authors have gone with language independent methodologies, why some authors decided to add linguistic knowledge to their systems and what improvements they attained by doing it. We will also discuss some methodologies and the problems faced by systems which aim at extracting Translation Equivalents from aligned parallel texts.

This paper is organised as follows: the next section gives a brief overview of what parallel texts alignment is. Section 3 provides some evidence on the amount of lexical cues available in European languages. Section 4 describes previous work on alignment techniques developed by some authors, both on sentence and word level, and discusses the strategies they used regarding language independence. Section 5 describes how language independent the extraction of Translation Equivalents can be. Finally, section 6 presents the conclusions and some future work.

2. Parallel Texts Alignment

Today, it has become quite common to find parallel texts virtually everywhere from translations of books in bookshops, to consumer products information in supermarkets, instructions manuals in the industry, multilingual portals in the Internet, and it has even become trendy to find parallel versions of songs in English and Spanish. In all these parallel texts, one can notice a continuum in the 'degree of non-parallelness' from legislative texts and instructions manuals, which tend to be very faithful to the originals, to translations of books or lyrics of songs, which leave more freedom and creativity to the translator.

Text alignment techniques aim at identifying automatically correspondences between those parallel texts. Once they are aligned it is possible to start using them for various purposes. For example, an immediate application is the production of bilingual concordances. Bilingual concordances are particularly useful for the preparation of commercial bilingual dictionaries, for translators and even for foreign language learners. They allow the examination of the way specific words or terms are translated into another language, providing simultaneously part of the context in which they appear.

Furthermore, they can also be used to build Bilingual Dictionaries, Bilingual Terminology Databanks, Translation Memories, to name but a few immediate applications.

[1] Danish (da), Dutch (nl), English (en), Finnish (fi), French (fr), German (de), Greek (el), Italian (it), Portuguese (pt), Spanish (es) and Swedish (sv).

This data can then be included in Machine Translation systems, Computer Assisted Translation tools, Cross-Language Information Retrieval systems or Lexicographers workbenches.

The lexical cues found in parallel texts have been quite used. They can be identical tokens in two texts (numbers, proper names, punctuation marks), similar words (cognates, like *Comissão* and *Commission*, in Portuguese and English, respectively) or known translations (like *data* and *fecha*, in Portuguese and Spanish, respectively). These tokens, called *anchors* (from Kay and Röscheisen, 1993, p. 128), allow correspondences between the texts, and help the alignment system to keep track of the evolution of text and to avoid straying away from the correct alignment.

3. Sharing Words

Several authors have used lexical cues as potential anchors for alignment. In fact, the number of identical tokens available in parallel texts should not be underestimated.

According to the results reported in Ribeiro *et al.* (2000), almost 15% of the 'vocabulary' (different tokens) found in their texts from the Official Journal of the European Communities was found to be the same in its various official languages with respect to the Portuguese text (this number also includes names, numbers and punctuation). They used a sample of parallel texts from three sources: records of the Written Questions to the European Commission, records of Debates in the European Parliament and Judgements of The Court of Justice of the European Communities. Table 1 gives an overview of the equal vocabulary size across the ten language pairs (see footnote 1 for the abbreviations):

	Sub-Corpus			
Pair	Written Questions	Debates	Judgements	Average
pt-da	1.2k (17%)	2.0k (10%)	0.2k (19%)	1.9k (11%)
pt-de	1.0k (15%)	1.9k (10%)	0.2k (19%)	1.8k (10%)
pt-el	1.0k (15%)	1.5k (8%)	0.1k (18%)	1.5k (9%)
pt-en	1.3k (19%)	2.2k (11%)	0.2k (20%)	2.1k (12%)
pt-es	2.5k (38%)	6.5k (32%)	0.3k (36%)	6.0k (33%)
pt-fi	---	---	0.2k (19%)	0.2k (19%)
pt-fr	1.3k (19%)	2.3k (11%)	0.2k (22%)	2.1k (12%)
pt-it	1.4k (22%)	3.0k (15%)	0.2k (25%)	2.8k (16%)
pt-nl	1.2k (17%)	2.0k (10%)	0.1k (19%)	1.9k (11%)
pt-sv	---	---	0.2k (19%)	0.2k (19%)
Average	1.3k (20%)	2.7k (13%)	0.2k (22%)	2.5k (14%)

Table 1: Average size of common vocabulary per pair of parallel texts in thousands.

Table 1 also shows the average percentages with respect to the size of the vocabulary found in Portuguese parallel texts are in brackets.

For example, an average of 2500 tokens were found to be exactly the same for the Written Questions parallel texts in Portuguese and Spanish (pt-es). This corresponds to an average of 38%, i.e. 38% of the vocabulary found in the Portuguese Written Questions parallel texts was equal to the Spanish vocabulary.

In the case of close languages such as Portuguese and Spanish, the average rate rises to more than 30%; for the opposite reason, it drops to about 10% for the pair Portuguese–German. Furthermore, the number of occurrences of these shared vocabulary tokens in the parallel texts (see Table 2) reaches an average of almost 50% in parallel texts in Portuguese and Spanish. For Portuguese and German parallel texts, this number is about 20% on average.

	Sub-Corpus			
Pair	Written Questions	Debates	Judgements	Average
pt-da	18.3k (32%)	103.6k (25%)	1.5k (33%)	92.5k (26%)
pt-de	15.0k (27%)	80.7k (19%)	1.4k (31%)	72.2k (20%)
pt-el	16.4k (29%)	66.7k (16%)	1.4k (31%)	60.1k (18%)
pt-en	17.8k (31%)	100.5k (24%)	1.4k (30%)	89.8k (25%)
pt-es	29.7k (52%)	192.5k (46%)	2.4k (52%)	171.4k (47%)
pt-fi	---	---	1.3k (30%)	1.3k (30%)
pt-fr	22.8k (40%)	106.3k (26%)	1.9k (41%)	95.5k (27%)
pt-it	20.3k (35%)	96.7k (23%)	1.8k (38%)	86.7k (25%)
pt-nl	19.8k (35%)	106.0k (25%)	1.6k (35%)	94.8k (26%)
pt-sv	---	---	1.3k (29%)	1.3k (29%)
Average	20.0k (35%)	106.6k (25%)	1.6k (35%)	95.4k (27%)

Table 2: Average number of common tokens per pair of parallel texts in thousands

Table 2 also shows the average percentages of common tokens with respect to the number of tokens of the Portuguese parallel text are in brackets. For example, about 1400 tokens are were found to be equal in both Greek and Portuguese for the Judgements parallel texts; this covers 31% of the total number of tokens of the Portuguese Judgements parallel texts.

This is a wealthy source of lexical cues for parallel texts alignment that should not be left unused.

Homographs, as a naive and particular form of cognates, are likely translations, which makes them potential reliable anchors. For example, *Portugal* is written like this in several European languages, which makes it a potential anchor for alignment.

These anchors end up being mainly numbers and names. Here are a few examples of anchors from a parallel text in English and Portuguese: *2002* (numbers, dates), *ASEAN* (acronyms), *Patten* (proper names), *China* (names of countries), *Manila* (names of cities), *apartheid* (foreign words), *Ltd* (abbreviations), *habitats* (Latin words), *ferry* (common words), *global* (common vocabulary).

4. Alignment Techniques

Some alignment techniques establish correspondences between sentences – *sentence alignment* – where as other techniques try to provide more fine-grained alignments by establishing correspondences between words – *word alignment*. The next section will describe some sentence alignment techniques. Section 4.2 describes word alignment techniques.

4.1. Sentence Alignment

Back in the early days of alignment, in the 1990s, sentences were set as the basic units for alignment. Each text was viewed as a sequence of sentences and alignment algorithms attempted at making correspondences between the sentences in the parallel texts.

The method proposed by Kay and Röscheisen (1993) assumed that for sentences in a translation to correspond, the words in them must also correspond. Two words were considered to have similar distributions if they tended to co-occur in the tentatively aligned sentences. In this case, if their measure of similarity was above a threshold, it would mean they were translations and, finally, sentences were aligned if the number of words associating them was greater than an empirically defined threshold.

In other alternative approaches, less knowledge based, sentences were aligned as long as they had a proportional

number of words (Brown *et al*., 1991) or characters (Gale and Church, 1991). They started from the fact that long sentences tend to have long translations and, conversely, short sentences tend to have short translations. This correlation was the basis for their statistical models. Brown *et al*. (1991, p. 175) remarked that the error rate was slightly reduced from 3.2% to 2.3% when using some linguistic knowledge like the time stamps, question numbers and author names found in the parallel texts. This confirmed that sentences could be aligned just by looking at sentence lengths measured in number of tokens and that extra linguistic knowledge did not improve the results significantly.

Although none of these algorithms depend on some word similarity measure as in later work (e.g. Simard *et al*., 1992), these algorithms tended to break down when sentence boundaries were not clearly marked.. This means full stops would have to be clearly interpreted as sentence boundaries markers. However, they are not safe markers of sentence boundaries.

Gale and Church (1991, p. 179) reported that only 53% of the full stops found in the Wall Street Journal were used to mark sentence boundaries. Full stops may be part of abbreviations (*Dr. A. Bromley*) or numbers (*1.3%*), they are not usually found in headlines (*Tyre production*), they may not even exist because they were not added, or they were either lost or were mistaken for noise in the early days when electronic versions of parallel texts were still rare and texts needed to be scanned.

Wu (1994) also aligned English–Chinese sentences with proportional lengths. He also began by applying a method similar to the one used by Gale and Church (1991) and reported results not much worse than those expected by this algorithm. Still, he claimed sentence alignment precision over 96% when the method incorporated a seed bilingual lexicon of words commonly found in the texts to be aligned (e.g. names of months, like *December* and its equivalent in Chinese 十二月). So, again Wu's work confirmed that the use of lexical cues would be beneficial for alignment.

4.2. Word Alignment

If word alignment is the main goal, alignment algorithms must be more 'careful' in order to avoid wrong word correspondences. This is a much more fine-grained alignment since it is no longer done at sentence level but at word level. In contrast with sentence alignment algorithms which permit a margin of tolerance for occasional wrong word matches, at word level, the sentence is no longer a 'safety net'. Consequently, the penalty on wrong word matches becomes much higher.

By adding some lexical information, Church (1993) showed that alignment of parallel text segments was possible by exploiting orthographic *cognates* instead of sentence delimiters. He used the rule of equal 4-grams in order to find 'cognate' (similar) sequences of characters in the parallel texts, i.e. sequences of four characters which are equal in the texts. This is a good strategy for languages which share lexical similarities like languages which share a character set.

The idea of exploiting *cognates* for alignment had been proposed one year earlier in a paper by Simard *et al*. (1992). According to the Longman Dictionary of Applied–Linguistics, a cognate is "a word in one language which is similar in form and meaning to a word in another language because both languages are related" (Richards *et al*., 1985, p. 43). For example, the words *Parliament* and *Parlement*, in English and French respectively, are cognates. They are similar in form and have the same meaning. When two words have the same or similar forms in two languages but have different meanings in each language, they are called false cognates or false friends (Richards *et al*., 1985, p. 103). For example, the English word *library* and the French word *librairie* are false cognates (Melamed, 1999, p. 114): *library* is translated as *bibliothèque* in French and, conversely, *librairie* as *bookstore* in English.

Simard *et al*. (1992) used a simple rule to check whether two words were cognates. They considered two words as cognates if their first four characters were identical (Simard *et al*., 1992, p. 71), as is the case of *Parliament* and *Parlement*. This simple heuristic proved to be quite useful, providing a greater number of lexical cues for alignment though it has some shortcomings. According to this rule, the English word *government* and the French word *gouvernement* are not cognates. Also, *conservative* and *conseil* ('council'), in English and French respectively, are wrongly considered as cognates (Melamed, 1999, p. 113). The rule is sensitive to variations in the first four letters and it does not distinguish different word endings.

In fact, both the rule proposed by Simard *et al.* (1992) and the one used by Church (1993) are two variants of Approximate String Matching Techniques. The former technique corresponds to *truncation*, where only the *n* first characters are considered. The latter technique resembles *n*-gram matching, which determines the similarity of two words by the number of common *n*-grams. A technique developed by Adamson and Boreham (1974) uses contiguous bigrams and base their word similarity score on the coefficient of Dice to compare the number of common bigrams between two words and the number of bigrams of each individual word.

McEnery and Oakes (1995) tried to improve the definition of cognates by comparing the truncation technique, the number of shared bigrams in two words with a score based on the coefficient of Dice and using dynamic programming. In experiments they performed comparing English and French vocabulary, they found that the bigram matching technique precision was 97% using a threshold of 0.9, and 81% for a similarity score between 0.8 and 0.9; the truncation technique precision was 97.5% for a length of eight characters and 68.5% for a length of six characters.

The word alignment approaches just described are not appropriate for pairs of languages for which it is not possible to find some common cues. In order to overcome this problem, Melamed (1999, p. 113) also suggests the use of *phonetic cognates* especially for languages with different alphabets. Phonetic cognates are words which are phonetically similar though written differently or in different scripts, like 'program' /pr græm/ and 'プログラム' /puroguramu/ in English and Japanese. This increases the number of cues available for alignment.

The requirement for clear sentence boundaries was dropped in Fung and Church (1994) on a case-study for English-Chinese. It was the first time alignment procedures were being tested on texts between non-Latin languages and without finding sentence boundaries. Each

parallel text was split into K pieces and word correspondences were identified by analysing their distribution across those pieces. In particular, a binary vector of occurrences with size K (hence, the K-vec) would record the occurrence of a word in each of the pieces. Should the word occur in the *i*-th piece of the text, then the *i*-th position of the vector would be set to '1'. Next, in order to find whether two words corresponded, their respective K-vecs were compared. In this way, it was possible to build a rough estimate of a bilingual lexicon. This would feed the alignment algorithm of Church (1993), where each occurrence of two translations would become a dot in the graph.

This method was extended in Fung and McKeown (1994). It was also based on the extraction of a small bilingual dictionary based on words with *similar distributions* in the parallel texts. However, instead of K-vecs, which stored the occurrences of words in each of the K pieces of a text, Fung and McKeown (1994) used vectors that stored the distances between consecutive occurrences of a word (DK-vec's). For example, if a word appeared at offsets (2380, 2390, 2463, 2565, ...), then the corresponding distances vector would be (10, 73, 102, ...). Should an English word and a Chinese word have distance vectors with a similarity above a threshold, then those two words would be used as potential anchors for alignment. Later, in Fung and McKeown (1997), rather than using only single words, the algorithm extracted terms to compile the list of reliable pairs of translations, using specific syntactic patterns. However, this made it become language dependent.

Melamed (1999) also used orthographic cognates. Moreover, he used lists of stop words to avoid matching of closed-class words (like articles and prepositions) which tended to generate much noise, which requires some linguistic knowledge to be hand-coded into the system. In order to measure word similarity, he defined the Longest Common Sub-sequence Ratio as follows:

$$Ratio(w_1, w_2) = \frac{Length(LongestCommonSub\text{-}Sequence(w_1, w_2))}{Max(Length(w_1), Length(w_2))}$$

where w_1 and w_2 are the two words to be compared (Melamed, 1999, p. 113).

This measure compares the length of the longest common sub-sequence of characters with the length of the longest token. For the previous example, the ratio is 10 (the length of *government*) over 12 (the length of *gouvernement*) whereas the ratio is just 6 over 12 for *conservative* and *conseil* (council). This measure tends to favour long sequences similar to the longest word and to penalise sequences which are too short compared to a long word. However, for this very same reason, it fails to consider *gouvernement* and *governo* in French and Portuguese as cognates because *governo* is a shorter word. Their ratio is also 6 over 12.

For alignment purposes, Melamed (1999) selects all pairs of words which have a ratio above a certain threshold, heuristically selected. However, this becomes a language dependent value. Still, this comparison measure seems to provide better results than the one first proposed by Simard *et al.* (1992) but it is not based on a statistically supported study.

Danielsson and Mühlenbock (2000) aim at aligning cognates starting from aligned sentences in two quite similar languages: Norwegian and Swedish. The 'fuzzy match' of two words is "calculated as the number of matching consonants[,] allowing for one mismatched character" (Danielsson and Mühlenbock, 2000, p. 162). For example, the Norwegian word *plutselig* (suddenly) and the Swedish word *plötsligt* would be matched through *pltslg*: all consonants match except for the 't'. However, *bakspeilet* (rear-view mirror) and *backspegeln*, in Norwegian and Swedish respectively, would not match because four consonants are not shared 'c', 'g', 'n' and 't'. This strategy resembles the technique developed by Pollock and Zamora (1984, p. 359) whereby words are coded using the first letter of the word, the remaining unique consonants in order of occurrence and, finally, the unique vowels also in order of occurrence – the *skeleton key*. For example, *plutselig* would be coded as *pltsguei* and *plötsligt* as *pltsgöi* where the sequence of consonants is equal.

Choueka *et al.* (2000) present an alignment algorithm for English and Hebrew, a highly inflected language with a different alphabet, a complex morphology and flexible word order. For example, *and since I saw him* is translated into a single Hebrew word (Choueka *et al.*, 2000, p. 74): åëùøàéúéå /ukhshereitiv/. First, texts were lemmatised, i.e. each word was reduced to its basic form as found in a dictionary entry (e.g. *saw* to *see*). This is clearly a language dependent task though it is quite difficult to solve for highly inflected languages. Also high frequency words were removed using a list of stop words. Next, parallel texts were aligned using the methodology of Fung and McKeown (1997). The lemmatisation step increases the chances of finding similar words in the aligned parallel texts in order to compile automatically a bilingual dictionary by analysing their distribution across the aligned parallel texts; otherwise, non-lemmatised words would just become too rare which would make it difficult to find trustworthy Translation Equivalents due to data sparseness.

In contrast with previous approaches, Ribeiro *et al.* (2001b) consider two words to have a high level of 'cognateness', if they share a typical sequence of characters that is common to that particular pair of

Language Pairs	pt-da	pt-de	pt-el	pt-en	pt-es	pt-fi	pt-fr	pt-it	pt-nl	pt-sv	Average
punctuation	60%	63%	66%	60%	46%	68%	59%	60%	63%	74%	62%
numbers	16%	20%	20%	21%	11%	18%	16%	12%	19%	20%	17%
names	17%	8%	7%	10%	8%	7%	6%	5%	9%	1%	8%
common words	3%	3%	6%	7%	33%	6%	14%	18%	9%	3%	10%
others	4%	6%	1%	2%	2%	2%	5%	5%	1%	1%	3%
Total	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%

Table 3: Percentages of types of tokens used for alignment in the alignment algorithm developed by Ribeiro *et al.* (2001).

languages. The typical sequences of characters can be extracted by statistical data analysis of contiguous and non-contiguous sequences of characters, based on the notion of 'textual unit' association. They were able to find typical sequences which lie in the beginning of words such as •*Comis*, for *Comissão* and *Comisión* in Portuguese and Spanish, lie in the middle of words as in *f_rma* which matches both *information* and *informação* in English and Portuguese respectively, cross word boundaries as *i_re•ci* for the Portuguese–French pair as in *livre•circulação–libre•circulation* ('free movement'), which made it quite adequate for the pairs of languages which use words written in the same character set. It is up to the alignment algorithm proper to confirm whether words are cognates depending on their position in the text.

This particular experiment used the Judgements sub-corpora, in three language pairs: Portuguese–English, Portuguese–French and Portuguese–Spanish. The size of the parallel texts for each language pair amounted to about 150k characters (about 30k tokens). Table 4 shows the number of typical sequences of characters extracted from each parallel text.

Language Pair	Number of Sequences
Portuguese–English	677
Portuguese–Spanish	1137
Portuguese–French	877

Table 4: Number of typical sequences of characters for each pair of languages.

Interestingly, Table 4 also confirms language similarity. Bearing in mind that Portuguese and Spanish are two quite close languages, it does not come as a surprise to see that this pair shares more typical sequences of characters than any of the other. French comes next for its closeness as a Romance language and English comes last confirming that Portuguese and English are more distant languages.

Table 3 presents an analysis of a sample of aligned parallel texts, using the previously mentioned methodology though just using equal tokens. The table shows that punctuation marks are indeed good cues for alignment. On average, more than 60% of the tokens used for correspondence points are punctuation marks. This confirms the success of early approaches that started by using sentences as the basic alignment unit and exploiting full stops for sentence alignment. It shows that the number of common words used as correspondence points is higher for similar pairs of languages like Portuguese–French, Portuguese–Italian and Portuguese–Spanish than for other pairs. It shows that, on average, 10% of the tokens used as correspondence points are common words and that 17% are numbers, which makes it more than a quarter of all tokens used as correspondence points.

5. Translation Equivalents

The extraction of Translation Equivalents is one of the most important tasks for building either Translation Memories or Bilingual Dictionaries. Translations databanks are useful language resources either for Machine Translation, Cross-Language Information Retrieval or even for human translators themselves.

Aligned parallel texts are ideal sources to extract Translation Equivalents for they provide the correspondences between the original text and their translations in other languages made by professional translators. They allow the examination of the way specific words or terms are translated into other languages. Aligned parallel texts can reduce the amount of effort necessary to build Translation Databanks.

The key issue in the extraction of Translation Equivalents is to find a correlation between co-occurrences of terms in parallel texts. In general, if two terms co-occur often in aligned text segments, then they are likely to be *equivalent*. The alignment of parallel texts splits the texts into small aligned text segments and reduces the number of words that must be checked for co-occurrence. In order to identify Translation Equivalents, their *distribution similarity* must be analysed in those aligned segments.

However, the larger the aligned text segments, the more difficult it gets to extract Translation Equivalents for more alternative translations become possible and, consequently, the search space becomes larger and with fewer evidences. This may be the case for distant languages where fewer cues may be available for alignment. As a result, the number of Translation Equivalents which can be more reliably extracted gets more reduced.

Nonetheless, the few Equivalents extracted can be subsequently fed back into the alignment system to improve the alignment proper, reduce the size of aligned text segments, and extract more Translation Equivalents in an iterative and unsupervised way. Even though it should only be possible to extract a small bilingual lexicon as in Fung and McKeown (1994), it can be quite helpful to bootstrap a more fine-grained alignment as Wu (1994) has shown.

6. Conclusions and Future Work

The exploitation of lexical cues for parallel text alignment is indeed quite helpful for alignment methods based on lexical information found in the texts. The more lexical information shared between a pair of languages, the more candidate correspondence points for alignment can be generated. As a result, this leads eventually to a more fine-grained alignment beyond the sentence level as in the early 1990s. Language similarity should be seen as bonus for alignment.

Language independent approaches are quite dear in multilingual regions where the possibility of using a single methodology to handle different languages increases portability and greatly reduces the amount of human effort. Ideally, an alignment algorithm should be completely language independent: character set independent; no previous linguistic knowledge, either from machine-readable bilingual dictionaries or hand coded seed bilingual translation lexicons; no lemmatised and/or tagged texts; no requirement for the detection of sentence boundaries.

However, as described in section 4, previous alignment approaches have often resorted to making use of sentence boundaries, lexical cues available in the parallel texts and even to hand-coding some linguistic knowledge through small bilingual lexicons and building list of stop words. This increases the number of potentially

reliable anchors for alignment and increases the chances of having more accurate alignments of parallel texts.

Nevertheless, it is wise to make good use of the lexical cues available in parallel texts. The larger the overlap between common lexical cues between two languages, the higher the number of potential anchors for alignment. Eventually, this means that the average size of aligned parallel texts gets smaller for non-sentence based alignment algorithms. The extraction of Translation Equivalent becomes more reliable and easier since there may be fewer alternative translations to choose from.

Consequently, when it comes to more distant languages like Portuguese and Chinese, where the number of lexical cues available is more reduced, the number of Translation Equivalents extracted is usually more reduced (Ribeiro *et al*., 2001a). However, it is still possible to extract some Translation Equivalents reliably in order to re-feed the alignment algorithm. Indeed, for distant languages, previous authors (Wu, 1994; Melamed, 1999) have resorted to building a small bilingual lexicon to bootstrap the alignment algorithm.

We believe that it is possible to extract some Translation Equivalents in a 'self-enriching' process instead of feeding an alignment system manually with either hand-coded bilingual lexical information or incomplete machine readable dictionaries.

By re-feeding the extracted Translation Equivalents back into the aligner it is possible to increase the number of candidate correspondence points for new lexical cues become available for the generation of correspondence points. The more candidate correspondence points, the more fine-grained the alignment and the better are the extracted equivalents. This means that the alignment precision may be improved, i.e. more correspondences may be established between words or phrases.

As the example of Choueka *et al*. (2000) has shown, it becomes more difficult to get cues for highly inflected languages where words can suffer major changes. Still, it would be interesting to test whether it should be possible to automatically lemmatise texts either by using a strategy similar to the one presented in Kay and Röscheisen (1993) whereby common suffixes and prefixes of words were automatically identified in a language independent fashion, or by extracting automatically character patterns using a methodology similar to the developed in Ribeiro *et al*. (2001b).

All in all, most work on alignment has been carried out on a wide range of 'popular' languages, most of them on English and French, but also including other Western European languages, Arabic, Chinese, Hebrew, Japanese and even some Korean. It would be quite interesting to test alignment algorithms on radically different languages to check for their degree of language independence.

7. References

Adamson, G. and Boreham, J. (1974). The Use of an Association Measure Based on Character Structure to Identify Semantically Related Pairs of Words and Document Titles. *Information Storage and Retrieval*, 10, 253–260.

Brown, P., Lai, J. and Mercer, R. (1991). Aligning Sentences in Parallel Corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics* (pp. 169–176), Berkeley, California, USA.

Choueka, Y., Conley, E. and Dagan, I. (2000). A Comprehensive Bilingual Word Alignment System: Application to Disparate Languages – English and Hebrew. In J. Véronis (ed.), *Parallel Text Processing: Alignment and Use of Translation Corpora* (pp. 69–96). Dordrecht, The Netherlands: Kluwer Academic Publisher.

Church, K. (1993). Char_align: A Program for Aligning Parallel Texts at the Character Level. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics* (pp. 1–8). Columbus, Ohio, USA.

Danielsson, P. and Mühlenbock, K. (2000). Small but Efficient: The Misconception of High-Frequency Words in Scandinavian Translation. In J. White (ed.), *Envisioning Machine Translation in the Information Future – Proceedings of the 4th Conference of the Association for Machine Translation in the Americas, AMTA 2000 – Lecture Notes in Artificial Intelligence*, 1934, pp. 158–168. Berlin, Germany: Springer-Verlag.

Fung, P. and Church, K. (1994). K-vec: A New Approach for Aligning Parallel Texts". In *Proceedings of the 15th International Conference on Computational Linguistics – Coling'94* (pp. 1096–1102), Kyoto, Japan.

Fung, P. and McKeown, K. (1997). A Technical Word- and Term-Translation Aid Using Noisy Parallel Corpora across Language Groups. In *Machine Translation*, 12(1–2), 53–87. Dordrecht, The Netherlands: Kluwer Academic Publisher.

Fung, P. and McKeown, K. (1994). Aligning Noisy Parallel Corpora across Language Groups: Word Pair Feature Matching by Dynamic Time Warping. In *Technology Partnerships for Crossing the Language Barrier: Proceedings of the First Conference of the Association for Machine Translation in the Americas* (pp. 81–88). Columbia, Maryland, USA.

Gale, W. and Church, K. (1991). A Program for Aligning Sentences in Bilingual Corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics* (pp. 177–184), Berkeley, California, USA (short version). Also (1993) *Computational Linguistics*, 19 (1), 75–102 (long version).

Kay, M. and Röscheisen, M. (1993). Text-Translation Alignment. *Computational Linguistics*, 19 (1), 121–142.

Melamed, I. (1999). Bitext Maps and Alignment via Pattern Recognition. *Computational Linguistics*, 25 (1), 107–130.

McEnery, A. and Oakes, M. (1995). Sentence and Word Alignment in the CRATER Project: Methods and Assessment. In S. Warwick-Armstrong (ed.), *Proceedings of the SIGDAT Workshop "From Texts to Tags: Issues in Multilingual Language Analysis"* (pp. 77–86). Dublin, Ireland.

Pollock, J. and Zamora, A. (1984). Automatic Spelling Correction in Scientific and Scholarly Text. *Communications of the Association for Computing Machinery (ACM)*, 27 (4), 358–368. ACM Press.

Simard, M., Foster, G. and Isabelle, P. (1992). Using Cognates to Align Sentences in Bilingual Corpora. In *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation TMI-92* (pp. 67–81), Montréal, Canada.

Simard, M. and Plamondon, P. (1998). Bilingual Sentence Alignment: Balancing Robustness and Accuracy. *Machine Translation*, 13 (1), 59–80. Dordrecht, The Netherlands: Kluwer Academic Publisher.

Ribeiro, A., Lopes, G. and Mexia, J. (2000). Using Confidence Bands for Parallel Texts Alignment. In *Proceedings of the 38th Conference of the Association for Computational Linguistics* (pp. 432–439). Hong Kong, China.

Ribeiro, A., Lopes, G. and Mexia, J. (2001a). Extracting Translation Equivalents from Portuguese-Chinese Parallel Texts. In *Proceedings of Asialex 2001 – The Second International Congress of the Asian Association for Lexicography (Asialex)* (pp. 225–230). Seoul, South Korea.

Ribeiro, A., Dias, G., Lopes, G. and Mexia, J. (2001b). Cognates Alignment. In B. Maegaard (ed.), *Proceedings of the Machine Translation Summit VIII – MT Summit VIII – Machine Translation in the Information Age* (pp. 287–292). Santiago de Compostela, Spain.

Richards, J., Platt, J. and Weber, H. (1985). *Longman Dictionary of Applied Linguistics*. London, United Kingdom: Longman.

Wu, D. (1994). Aligning a Parallel English–Chinese Corpus Statistically with Lexical Criteria. In *Proceedings of the 32nd Annual Conference of the Association for Computational Linguistics* (pp. 80–87), Las Cruces, New Mexico, USA.

SPE based selection of context dependent units for speech recognition

Matjaž Rodman♦, Bojan Petek and Tom Brøndsted*

Interactive System Laboratory
Faculty of Natural Sciences and Engineering
University of Ljubljana
Snežniška 5, 1000 Ljubljana, Slovenia
matjaz.rodman@ntftex.uni-lj.si, bojan.petek@uni-lj.si

*Center for PersonKommunikation (CPK)
Institute of Electronic Systems
Aalborg University
Niels Jernes Vej 12, 9220 Aalborg, Denmark
tb@cpk.auc.dk

Abstract

Decision tree-based approach is a well known and frequently used method for tying states of the context dependent phone models since it is able to provide good models for contexts not encountered in the training data. In contrast to the other approaches, this method allows us to include expert linguistic knowledge into the system. Our research focused on the inclusion of standard generative theory by Chomsky & Halle (1968), called the SPE theory (the Sound Pattern of English), into the decision tree building process as expert linguistic knowledge. Our attempt was to "merge" the SpeechDat2 SAMPA label set, used for English and Slovenian languages, with the SPE. We created all possible natural groups of phones (SAMPA segments defined by a set of binary phonological features) for both languages and included them into a set of questions used in the process of creating the decision trees. Based on the decision tree constructed this way, we created an English and Slovenian speech recognition systems and tested both of them. Compared with the reference speech recognition system (Lindberg et al., 2000; Johansen et al., 2000) we got some promising results that encouraged us to continue this work and to perform further testing.

1. Introduction

Much of the phonetic variation in natural speech is due to contextual effects. In order to be able to accurately model variations in natural speech a careful choice of the units represented by each model is required. In large-vocabulary speech recognition systems, modelling of vocabulary words by subword units (phonemes or units derived from phonemes) is mandatory. For example, triphone models have been one of the most successful context dependent units because of their ability to model well the co-articulation effect. Yet if we create distinct models for all possible contexts, the number of models becomes very high. In practical applications of building speech recognition systems, there is often a conflicting desire to have a large number of models and model parameters in order to achieve high accuracy, whilst at the same time having limited and uneven training data in form of labelled utterances of a particular language (Young et al., 2000). In the case of triphone context dependent models, tying of HMM states gives us a possible solution of how to overcome this problem.

In our work we analysed the influence of the decision tree method on the acoustic modelling. We also analysed parameters that influence the decision tree building process and tested the proposed method based on the theory of naturalness (the theory that phonological segments cluster into "natural groups" defined by universal features), (Chomsky et al., 1968). We first examined this issue within the Slovenian language and then also addressed its portability to other languages.

2. Decision tree

When building large vocabulary cross-word triphone systems, unseen triphones are unavoidable. A limitation of the data-driven clustering procedure is that it does not deal with triphones for which there are no examples in the training data. Decision tree based approach gives us a possibility to include expert linguistic knowledge into a procedure of creating acoustic models. This methodology provides appropriate models also for contexts that are not seen in the training data. Therefore, decision trees are used in speech recognition with large numbers of context dependent HMMs, to provide models for contexts not seen in the training data. Sharing data at the model level may not be the most appropriate method for models composed of distinct states (Odell, 1995). Sharing distributions at the state level allows for finer distinctions to be made between the models by allowing left and right contexts to be modelled separately.

2.1. Decision tree building process

A phonetic decision tree is a binary tree in which a yes/no phonetic question is attached to each node (Young et al., 2000). Initially, all states in a given item list (typically a specific phone state position) are placed at the root node of a tree. Depending on each answer, every node is successively split and this continues until the states have trickled down to leaf-nodes. All states in the same leaf node are then tied and trained from the same data.

The question at each node is chosen to (locally) maximise the likelihood of the training data (using a log likelihood criterion) and gives the best split of the node. This process is repeated until the increase in log likelihood falls below the specified threshold. As a final stage, the decrease in log likelihood is calculated for merging terminal nodes, which belong to different parents. Any pair of nodes for which this decrease is less than the threshold used to stop splitting is then merged (Young et al., 2000). The algorithm for building a decision tree is summarised in figure 1.

♦ Socrates/Erasmus exchange student under the multilateral agreement UL D-IV-1/99-JM/Kc.

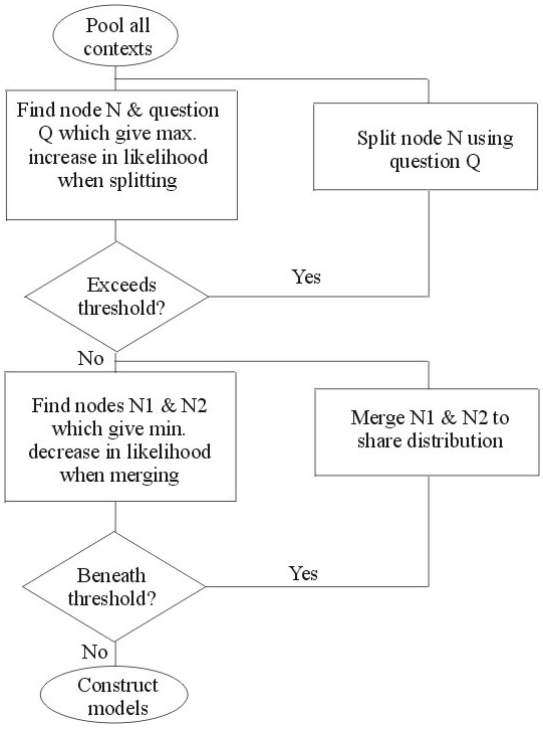


Figure 1. Algorithm for constructing decision tree (Odell, 1995)

Questions asked in the decision tree have a form:

QS "L_SL_Nasal" { m-,n-*, N-* }*

As an example, the command above defines the question "Is the left context a nasal?" where the group of nasals is represented by *{m-*, n-*,N-*}*. Only a finite set of questions can be used to divide each node. So questions have to be defined in a way that all possible natural groups of phonological segments are stated. That allows the incorporation of expert linguistic knowledge needed to predict contextual similarity when little or no data is available in order to determine which contexts are acoustically similar.

Decision tree building process has two stop criteria that determine how deep the tree will be. The first one is increase in the log likelihood that has to be achieved if node was split. In HTK (Young et al., 2000) it is defined with the command TB. The second one is the minimal occupation count that determines how many training data each node has to have. In HTK it is defined with the command RO.

3. SPE theory

Distinctive feature theory was introduced first by R. Jakobsen. He set up twelve universal inherent feature classes. Chomsky and Halle took over Jakobsens idea and defined 22 universal feature classes, which according to the standard SPE theory are sufficient for analysing expression segments of any language into distinctive oppositions.

The idea of natural phonetic groups is based on the so-called Sound pattern of English theory, "SPE", of Chomsky & Halle (1968). By this theory an inventory of expression segments can be described in terms of a hierarchical tree structure where upper nodes represent major class features (like +/- vocalic, +/- consonantal) and lower nodes cavity features, manner of articulation etc., and terminal nodes represent phones. A phonetic representation of an utterance in a given language has by this theory the form of a two-dimensional matrix in which the rows are labelled by features of universal phonetics; the columns stand for the consecutive segments of the utterance generated; and the entries in the matrix determine the binary value (+/-) of each segment with respect to the universal features (Chomsky et al., 1968). A set of phonological segments ("phonemes") sharing the same feature matrix and unequivocally defined by this matrix form a natural group. There are more degrees of naturalness. The SPE theory claims that one group is more natural than the other if the number of features defining it is smaller. The main natural groups (vowels, consonants, semi-vowels) are separated just by different values in major class features. Specific groups (e.g. back-vowels, plosives, nasals, labials) are defined by further features in the matrix and are consequently "less natural". Groups of segments that cannot be defined by a feature matrix are not natural (e.g., the pseudo group: k, a, m, h).

3.1. The use of SPE on SpeechDat2 databases

The starting point of our distinctive features composition can be described as follows:

- We intended to use the SPE as a generally accepted standard theory of phonology and with as few modifications as possible.
- Most notably, we have tried to utilise the Chomsky & Halle decomposition of English segments (1968) as directly as possible.
- Finally, we have attempted to make as few changes to the SpeechDat2 label set as possible.

Hence, our starting point can be paraphrased as attempt to "merge" the SAMPA label set used in SpeechDat2 database with the SPE.

The SPE sets up a total number of twenty-two feature classes, which according to the standard theory are sufficient for analysing expression segments (phonemes) of any language into distinctive oppositions. For a distinctive feature composition of the segments of a specific language, not all 22 feature classes are utilised. For instance, the SPE-description of English segments (Chomsky et al., 1968) makes references only to 13 feature classes. The remaining 9 classes may be regarded as redundant or "irrelevant" to English.

The set of 15 features was sufficiant to represent the set of Slovenian and English SAMPA symbols used in the SpeechDat2 database by the standard SPE theory. In general, we tried to preserve the original distinctive features used in the SPE. We had to, however, make some changes. In short, we replaced the feature vocalic with sonorant and syllabic, and added a feature front (Brøndsted, 1998). The feature +/- front is not within the set of 22 universal binary features defined in the SPE. However, the feature is needed additionally to +/-back because the SAMPA symbols include segments of a dubious phonological state, only specifiable with reference to three places of articulation: [-back, +front], [-back, -front], and [+back, -front].

3.2. Major Class Features

In standard generative phonology, the major class features sonorant, syllabic and consonantal are used to classify phonological segments into five major groups: vowels, non-syllabic liquids/nasals, syllabic liquids/nasals, glides, and obstruents. However, as the SAMPA segments defined for English and Slovenian do not include syllabic liquids/nasals, this in our case resulted in only four major groups (cf. table 1).

	Sonorant	Syllabic	Consonantal
Vowels	+	+	-
Glides	+	-	-
Syllabic Liquids and Nasals	+	+	+
Non-Syllabic Liquids and Nasals	+	-	+
Obstruents	-	-	+

Table 1: The main natural groups represented by major class features

3.3. The use of SPE on the Slovenian SpeechDat2 database

To create a distinctive feature composition table of the Slovenian SAMPA symbols used in SpeechDat2 we had to modify the phonetic transcriptions. In total, SpeechDat2 uses 46 SAMPA symbols in the Slovenian transcriptions. However, according to (Šuštaršič, 1999; Toporišič 2000) Slovenian only has 29 phonemes. Thus, 17 symbols must be considered allophonic variants. These allophones include certain composite pseudo segments (t_n, d_n, p_n, b_n, t_l, d_l) used along with the normal polyphonematic transcriptions (t n, d n ... etc.) in a way that appeared non-systematic to us. Consequently, we decided to change phonetic transcriptions in the database according to the following seven rules:

- Change string “t_n n” with two symbols “t n”
- Change string “d_n n” with two symbols “d n”
- Change string “p_n n” with two symbols “p n”
- Change string “b_n n” with two symbols “b n”
- Change string “t_l l” with two symbols “t l”
- Change string “d_l l” with two symbols “d l”
- Change symbol “W” with symbol “w”

This reduced the set of segments from 46 to 39. The resulting distinctive feature composition table of the Slovenian vowel and consonantal segments is shown in tables 2 and 3.

3.4. The use of SPE on the English SpeechDat2

Similarly we had to modify the transcriptions of the English SpeechDat2 database. The major problem was the monophonematic representation of diphtongs (as single phones). In SPE theory there are no phonological features differentiating diphthongs from monofthongs. This theory handles diphthongs with certain appropriate diphthongisation rules applied to the underlying representations (Chomsky et al., 1968). In order to provide a level of description conforming to the underlying

	i	i:	e	e:	E	E:	a	a:	u	u:	o	o:	O	O:	@	@r
Sonor.	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Syllabic	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
Conson.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
High	+	+	-	-	-	-	-	-	+	+	-	-	-	-	-	-
Back	-	-	-	-	-	-	+	+	+	+	+	+	+	+	-	-
Front	+	+	+	+	+	+	-	-	-	-	-	-	-	-	-	-
Low	-	-	-	-	+	+	+	+	-	-	-	-	+	+	-	+
Round	-	-	-	-	-	-	-	-	+	+	+	+	+	+	-	-
Tense	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	-
Anterior																
Coronal																
Voice																
Cont.																
Nasal																
Strident																

Table 2: Distinctive feature composition of Slovenian vowel segments

	b	d	g	p	t	k	dZ	ts	tS	s	S	z	Z	f	v
Sonor.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Syllabic	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Conson	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
High	-	-	+	-	-	+	+	-	+	-	+	-	+	-	-
Back															
Front															
Low															
Round															
Tense															
Anterior	+	+	-	+	+	-	-	+	-	+	-	+	-	+	+
Coronal	-	+	-	-	+	-	+	+	+	+	+	+	+	-	-
Voice	+	+	+	-	-	-	+	-	-	-	-	+	+	-	+
Cont	-	-	-	-	-	-	-	-	-	+	+	+	+	+	+
Nasal	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Strident	-	-	-	-	-	-	+	+	+	+	+	+	+	+	+

	w	j	x	r	l	m	n	N
Sonor	+	+	+	+	+	+	+	+
Syllabic	-	-	-	-	-	-	-	-
Conson.	-	-	-	+	+	+	+	+
High	+	+	-	-	-	-	-	+
Back	+	-	-	-	-			
Front	-	+	-	-	-			
Low	-	-	+	-	-	-	-	-
Round	+	-						
Tense	-	-						
Anterior	-	-	-	-	+	+	+	-
Coronal	-	-	-	+	+	-	+	-
Voice			-	+	+	+	+	+
Cont.			+	+	+	-	-	-
Nasal			-	-	-	+	+	+
Strident			-	-	-	-	-	-

Table 3: Distinctive feature composition of Slovenian consonantal segments

representation presupposed by the SPE, the diphtongs were re-written according to the 8 rules:

- Change symbol “eI” with phones “e” and “j”
- Change symbol “aI” with phones “{” and “j”
- Change symbol “OI” with phones “Q” and “j”
- Change symbol “@U” with phones “@” and “w”
- Change symbol “aU” with phones “{” and “w”
- Change symbol “I@” with phones “I” and “@”
- Change symbol “e@” with phones “e” and “@”
- Change symbol “U@” with phones “U” and “@”

The resulting distinctive feature composition of the English vowels and consonants are presented in tables 4 and 5.

	i:	u:	3:	O:	A:	I	U	e	{	Q	V	@
Sonor.	+	+	+	+	+	+	+	+	+	+	+	+
Syllabic	+	+	+	+	+	+	+	+	+	+	+	+
Conson.	-	-	-	-	-	-	-	-	-	-	-	-
High	+	+	-	-	-	+	+	-	-	-	-	-
Back	-	+	-	+	+	-	+	-	-	+	+	-
Front	+	-	-	-	-	+	-	+	+	-	-	-
Low	-	-	-	+	+	-	-	-	+	-	-	-
Round	-	+	+	+	-	-	+	-	-	+	-	-
Tense	+	+	+	+	-	-	-	-	-	-	-	-
Anterior												
Coronal												
Voice												
Cont.												
Nasal												
Strident												

Table 4: Distinctive feature composition of English vowel segments

3.5. Definition of natural groups

During the process of creating the decision tree, groups of phones are used to define questions that may be used in each node of the decision tree. This is the most important stage in the entire model-building procedure where expert phonological knowledge can be included (another one is the prior stage, where the actual set of phones to be used for segmentation and classification of the acoustic signal is established). For that reason, groups of phones for five languages - among these both Slovenian and English - were defined as a part of the COST 249 project. As the languages partly use the same phonemic label set (SAMPA), the groups are reuseable across languages. Slovenian contributes with 45 groups and English with 17 groups. During the process of creating the decision tree, two questions are created from every group defined. One is about the left context and the other about the right one. On the basis of these definitions we created English and Slovenian reference recognition systems.

Our main goal was to create another two systems for both languages that would have phone groups defined on the basis of the SPE theory. Therefore we automatically generated all natural groups of phones from the distinctive feature compositions table set up for the two languages. This resulted in 174 natural groups for Slovenian and 171 for English. The groups were used to create the set of all possible questions to be included in the process of building the experimental SPE-based speech recognition systems.

	b	d	g	p	t	k	dZ	tS	s	S	z	Z	f	T	v	D
Sonor.	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Syllabic	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Conson	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+	+
High	-	-	+	-	-	+	+	+	-	+	-	+	-	-	-	-
Back																
Front																
Low																
Round																
Tense																
Anterior	+	+	-	+	+	-	-	-	+	-	+	-	+	+	+	+
Coronal	-	+	-	-	+	-	+	+	+	+	+	-	+	-	+	
Voice	+	+	+	-	-	-	+	-	-	-	+	+	-	-	+	+
Cont	-	-	-	-	-	-	-	-	+	+	+	+	+	+	+	+
Nasal	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-
Strident	-	-	-	-	-	-	+	+	+	+	+	+	+	-	+	-

	w	j	h	r	l	m	n	N
Sonor	+	+	+	+	+	+	+	+
Syllabic	-	-	-	-	-	-	-	-
Conson.	-	-	-	+	+	+	+	+
High	+	+	-	-	-	-	-	+
Back	+	-	-	-	-			
Front	-	+	-	-	-			
Low	-	-	+	-	-	-	-	-
Round	+	-						
Tense								
Anterior	-	-	-	-	+	+	+	-
Coronal	-	-	-	+	+	-	+	-
Voice			-	+	+	+	+	+
Cont.			+	+	+	-	-	-
Nasal			-	-	-	+	+	+
Strident			-	-	-	-	-	-

Table 5: Distinctive feature composition of English consonantal segments

4. Importance of the order of questions for “unseen” contexts

We hypothesised a case of why it would be not advisable to create questions that would include all possible combinations of phonemes (including "unnatural" groups) and leave to the decision tree building process to chose the best ones by it’s own criteria. This way the decision tree building process would pick up only the important questions (likely involving only "natural" groups) and leave out the irrelevant ones. The idea emerged because of the explanation in the HTK documentation considering the problem of how to build questions for a decision tree: “There is no harm in creating extra unnecessary questions, because those which are determined to be irrelevant to the data will be ignored” (Young et al., 2000). That would yield us the optimal decision tree for this particular system without including any linguistic knowledge. By this definition also the order of the questions in the file that HTK uses for creating a decision tree should have no effect on the structure of the

decision tree. But already the first experiment showed us that the order of questions in this file *does* matter.

When we changed the order of questions in the file also the structure of decision tree has changed. Considering how questions are chosen in the process of building decision tree, we got a possible explanation for this change. For example let's suppose that we in the process of deciding how to cluster the centre state of the phone */m/*. Let's assume that we have data only for the triphones *a-m+**, *b-m+**, *c-m+** and *d-m+** where * means any context. Suppose further that we have defined the questions *QS "L_context1" {a-*, b-*, x-*}* and *QS "L_context2" {a-*, b-*}* where the first one is a superset of the first one (including also the left context '*x*'). The log-likelihood can only be calculated for data that is available for training. Therefore these two questions would cause the same increase in log likelihood if they were used for splitting the node because the left context *x-** does not appear in the training data. So if *L_context1* was used, the middle state of the model with the left context *x* would be trained from the same data as middle states of the models with left contexts *a* and *b*! Likewise, if *L_context2* was used, the middle state of the model with the left context *x* would be trained from the same data as the middle states of the models with left contexts *c* and *d* so from different data as in the first case. Both situations are presented in figure 2. Increase in log-likelihood would be the same in both cases. Therefore, only the order of questions in the file where questions are defined or the procedure that defines which question to use, if more questions give the same increase in log likelihood, would decide from which data model with left context *x* was trained. This means that for the models with contexts not seen in the training data (like *x* here) the decision from which data they'll be trained would depend on the order of questions.

From this we concluded that the phone groups that are later transformed into questions must not be defined without linguistic knowledge, because of the classification of contexts not appearing in the training data.

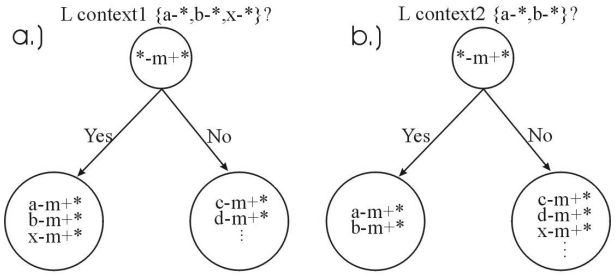


Figure 2. Effect of the order of questions on decision tree

5. Experimental methodology

The main scripts for training and testing acoustic models were implemented as Perl scripts invoking HTK. They were the outcome of the COST 249 project and intended to be used on the SpeechDat2 databases (Lindberg, 2000; Johansen, 2000) and are an extended version of the tutorial example in the HTK Book (Young et al., 2000). They can all be found on the Refrec homepage at

http://www.telenor.no/fou/prosjekter/taletek/refrec/

On this web page we can also find descriptions of standard tests and results of comparative tests done on many SpeechDat2 databases. We used hidden Markov models (HMM) having the 3-state left-right topology. We built triphone models and increased the number of Gaussian mixtures per state sequentially to 32.

5.1. The reference speech recognition systems

For building reference recognition systems we defined questions used in decision tree from groups of phones that were created as a part of the COST 249 project. For the English system we had 17 groups and for Slovenian 45 groups. During the training of acoustic models, data from labelled pronunciations of 800 speakers were used, while the data of the remaining 200 speakers was used as a test set.

The choice of good threshold values is important for the decision tree building process and requires some experimentation in practice. We therefore decided to experiment with the threshold set with the HTK RO command. This threshold determines how many training data each leaf in the decision tree must have. We built one Slovenian system with the threshold set to 100 and two English systems with thresholds set to 100 and 350, respectively (we named them sl-ref100, en-ref100 and en-ref350).

5.2. Speech recognition system with groups based on the SPE theory

In order to evaluate the effect of including the SPE theory into the decision tree building process we built five additional systems – three Slovenian and two English ones. For the model training we used the modified phonetic transcriptions as described in sec. 3.3 and 3.4. We automatically generated all natural phonetic groups from the distinctive feature compositions tables for both languages. From these groups, questions were generated that were used in the process of building decision trees for the two languages. Because of the modified phonetic transcriptions (less phones were used) and the modification of the broad classes, the number of leaves in the decision tree also changed and with that the distribution of the training data. In attempt to alter the amount of training data, we changed the threshold set with the HTK RO command for Slovenian systems from 100 to 267 and 350 and for English to 350 and 477. In this way we got five systems named sl-spe100, sl-spe267, sl-spe350, en-spe350 and en-spe477.

6. Speech recognition results

Six standard tests defined in the framework of the SpeechDat project (Johansen, 2000) were used on all reference and SPE based systems. These tests had the self-explanatory names: Yes/No test, Digits test, Connected Digits test, Application Words test, City Names test and Phonetic Rich Words test. In all tests but one (Connected Digits), each spoken test utterance consists of only one word. Therefore the word error rate (WER) is equal to the sentence error rate (SER) in these cases. Best results of tests done on all systems are given in table 6 and 7.

From these tables it can be observed that the SPE based systems performed either better or at least as good as the reference systems for both languages. The only

exception was the Application Words test on the Slovenian systems. We should also take into consideration that Yes/No, Digits and Connected Digits tests only applied to a small part of the decision tree. Specifically, the vocabulary in these tests is very limited and only a small number of triphones are therefore used.

	sl-ref100	sl-spe100	sl-spe267	sl-spe350
Yes/no	0,63	0,63	0,63	0,63
Digits	3,85	3,85	3,85	3,30
Con. Digits	4,12	3,91	3,95	3,98
App. Words	3,20	3,38	3,74	3,38
City Names	7,65	8,16	7,14	7,14
Ph. R. Words	17,62	17,36	15,93	15,51

	en-ref100	en-spe350	en-spe350	en-spe477
Yes/no	0,00	0,00	0,00	0,00
Digits	3,98	3,98	2,84	2,84
Con. Digits	5,42	5,51	4,22	4,33
App. Words	3,53	3,72	3,72	3,53
City Names	6,21	6,21	7,91	6,21
Ph. R. Words	36,83	35,01	32,68	31,56

Table 6: Lowest WER achieved by the Slovenian and English speech recognition systems in all six tests

	sl-ref100	sl-spe100	sl-spe267	sl-spe350
Con. Digits	15,75	14,56	14,56	14,32

	en-ref100	en-ref350	en-spe350	en-spe477
Con. Digits	30,72	30,92	24,50	25,10

Table 7: Lowest SER achieved by the Slovenian and English speech recognition systems in Connected Digits test

Without doubt, the most reliable evaluation of the SPE based concept can be taken from the Phonetic Rich Words test, employing the largest vocabulary (1491 words for Slovenian and 3043 for English) and more than 710 utterances. This test involves a very big part of the decision tree. This test also gave us the biggest decrease of the WER when comparing the SPE based concepts with the reference systems. The results achieved on the English systems had even bigger impact on the WER. The difference in WER of the best reference system and the best SPE based system is for Slovenian 1,85% and for the English 3,45%. Also the SER achieved with the SPE based systems in the Connected Digits test is better than the one achieved with the reference systems. The impact is again much bigger for English.

7. Conclusions

Within bounds of our experimental set-up we observed an advantage to include the SPE theory as an expert linguistic knowledge into the speech recognition systems. In general we got better results with the SPE-based processing for the English systems than for Slovenian ones. Several possible reasons can be referenced for such behaviour. One is probably the definition of phone groups for the reference systems. There were 45 phone groups defined in the Slovenian reference system while only 17 in the corresponding English one. Therefore, the increase in the number of natural groups resulting from the inclusion of the SPE theory had bigger impact on the English systems than on the Slovenian ones. Another possible reason is the presence of noise. Pronunciations in the Slovenian database were recorded in much higher presence of noise than the English ones. This could potentially have reduced the distinctive ability of some of the features used in the SPE theory.

One possible reason for achieving much better WER for the Phonetic Rich Words test and SER for the Connected Digits test with the English SPE based systems could be the fact that the English reference systems had much bigger error rates than the Slovenian ones. The lowest WER in the Phonetic Rich Words test achieved by the Slovenian reference system was 17,62% whereas it in case of English was 36,83%. The same was observed for the SER in the Connected Digits test (English reference system: 30,72%, Slovenian reference system: 15,75%).

From our experiments, we also concluded that groups of phones should never include actual "unnatural groups" and leave it to the decision tree building process to disregard them in favour of the more natural groups. That would present no significant problem to the classification of triphones that do appear in training data but would lead to the incorrect classification of triphones with contexts that do not appear in the training set.

Based on the experimental evidence we have shown that the creation of the natural groups of phonemes by the SPE theory could effectively be used in defining phone groups for the multilingual speech recognition system including multilingual triphone Markov models. When porting the HLT technology to a new target language, this provides us a promising alternative to the more widespread approach of using the union of phone group definitions from all languages (Zgank et al., 2001).

8. References

Brøndsted, T., 1998. A SPE based Distinctive Feature Composition of the CMU Label Set in the TIMIT Database. *Technical Report IR 98-1001,* Center for PersonKommunikation, Institute of Electronic Systems, Aalborg University

Chomsky, Noam, and Halle, Morris, 1968. *The Sound Pattern of English.* Harper & Row, Publishers New York, Evanston, and London.

Johansen, F.T., N. Warakagoda, B. Lindberg, G. Lehtinen, Z. Kačič, A. Žgank, K. Elenius, and G. Salvi, 2000. The COST 249 SpeechDat multilingual reference recogniser. *Paper for XL-DB.*

Lindberg, B., F.T. Johansen, N. Warakagoda, G. Lehtinen, Z. Kačič, A. Žgank, K. Elenius, and G. Salvi, 2000. A Noise Robust Multilingual Reference Recognizer Based on SpeechDat(II). *In Proc. ICSLP, International Conference on Spoken Language Processing,* Beijing,

Odell, J.J., 1995. *The Use of Context in Large Vocabulary Speech Recognition.* Dissertation submitted to the University of Cambridge for the degree of Doctor of Philosophy. Queens' College.

Šuštaršič, R., S. Komar, and B. Petek, 1999. *Illustrations of the IPA: Slovene.* Handbook of the International Phonetic Association: A Guide to the Use of the

International Phonetic Alphabet. Cambridge University Press, 135-139.

Toporišič, Jože, 2000. *Slovenska slovnica.* Maribor: Založba Obzorja.

Žgank, A., B. Imperl, F.T. Johansen, Z. Kačič, and B. Horvat. 2001. Crosslingual Speech Recognition with Multilingual Acoustic Models Based on Agglomerative and Tree-Based Triphone Clustering. *In Proc. EUROSPEECH, European Conference on Speech Communication and Technology*. Aalborg.

Young, Steve, Kershaw, Dan, Odell, Julian, Ollason, Dave, Valtchev, Vatcho, and Woodland, Phil. 2000. *The HTK Book (for HTK Version 3.0).* Cambridge: Entropic Cambridge Research Laboratory.

VIPTerm

The Virtual Terminology Information Point for the Dutch Language.

A Supranational project on terminology documentation and resources.

Prof. Dr. Frieda Steurs

Lessius Hogeschool
Dept. of Translators and Interpreters
Sint Andriesstraat 2
B-2000 Antwerp
e-mail : Frieda.Steurs@lessius-ho.be

1. VIPTerm

In 2001, the "Nederlandse Taalunie"(NTU) (Dutch Language Union) initiated a project to set up a virtual informationpoint for terminology (VIPTerm). This project can be considered as the Dutch part of the information and documentation requirements as stipulated by the TDCNet project.
The TDCNet project (European Terminology Documentation Centre Network) is an EU funded project(MLIS 4000 TDCNet 24264/0) with the main objective to create a virtual terminology directory in the form of a logical and physical network of terminology information and documentation centres in Europe.
Within this framework, both bibliographical data (data collections, literature, theses, etc.) and factual data (organisations, software, events, experts and training institutes) will be exchanged between national and regional information centres and compiled in an international terminology directory.
To reach this aim, the NTU set up a project to compile the data for both the Netherlands and Dutch speaking Belgium (Flanders).
VIPTerm (short for Virtual Terminology Information Point) will fullfil the function of a terminology institute, providing a documentation service and information point for users from different backgrounds.
The VIPTerm will also be designed to take up a function in the organisation of the terminologyfield and the networking in this field (a.o. by means of exchange, e.g. through a mailing list). This type of networking and fieldsupport is not the type of task the Nederlandse Taalunie wants to take care of through its own services. Tasks like these will have to be taken up by fieldorganisations, such as NL-TERM, the Dutch terminology association.

The main focus in the structure of this portal is to create an inventory and informationbase of organisations, events, activities , etc. that support a terminology policy for the Dutch language area. Next to this, in the future also the management, maintenance and distribution of electronic resources for Dutch, which is an important task of the NTU (cfr. Euromap and the Dutch Platform for language and speech research : TST (taal en spraaktechnologie), can be organised through this VIPTerm portal.

Both input and output formats were considered carefully in this project. By input format we refer to the actual database format that can be used to register the data.
The ISIS software has been known for some time as the central archiving system used by Unesco and other important datacollectors. The Winisis is a menu-driven generalised information storage and retrieval system designed specifically for the computational management of text-oriented data. Compared to ISIS, WinISIS has a Windows GUI. The output format, on the other hand, has special requirements as well. We need to make the database available through a number of information points, a.o. the special interactive website of the NTU ('Taalunieversum'), the ETIS portal to TDCNet, and websites of other user groups such as NL-TERM, the Dutch terminology association.

In the output structure, the webportal, the following basic categories will be taken into consideration :

- general information and history (short outline of the agents in the field : NTU; Coterm, NL-TERM, and the policy concerning terminology)

- publications, literature
- termcollections
- events/projects

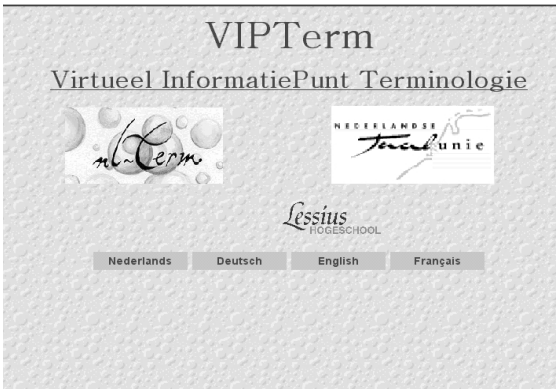
- training and education

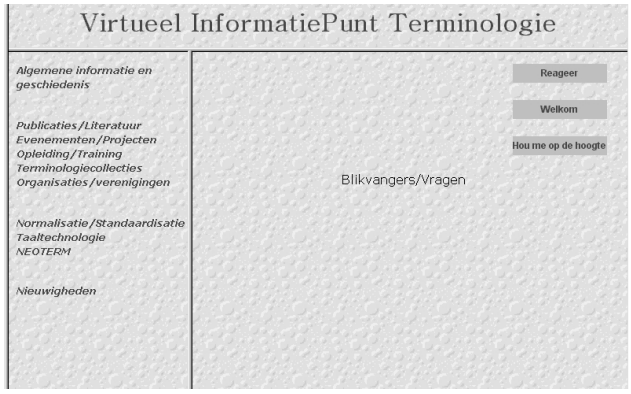
- standardisation
- language technology and terminology management tools
- neoterm
- novelties
- international links

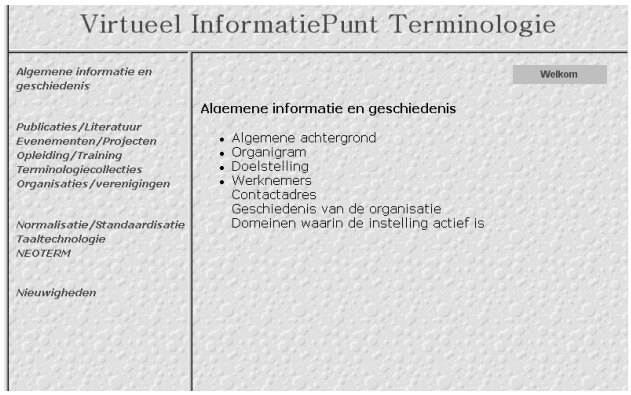
The project and pilotdesign was discussed thoroughly with our colleagues from the Fachhochschule Köln, who develop a similar project called DTP (Deutsches Terminologie Portal). We thoroughly investigated the classification of data (taking into account the existing classifications and TeDif format). We agreed on the principle to structure our portals in a similar way, so as to avoid unnessecary confusion.
Our sites will also be compatible with the ETIS server , as it was recently reprogrammed by the Union Latine.
For Dutch, an active exchange will be organised between the VIPTerm portal and ETIS, providing the data on Dutch terminology for the European level.

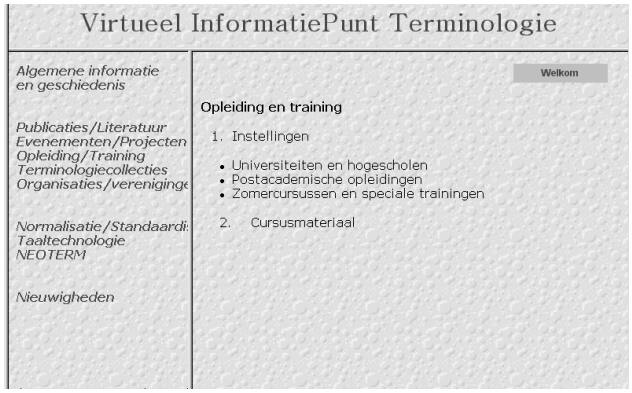
Once the analysis and study phase has been concluded, and advice has been collected through Coterm-experts and NL-TERM board members, we will build a sample portal site for this pilot project.
It will then be evaluated thouroughly and tested among a limited group of users.
If the final outcome of this evaluation is positive, then the VIPTerm project will be continued and will be organised on a more permanent basis.

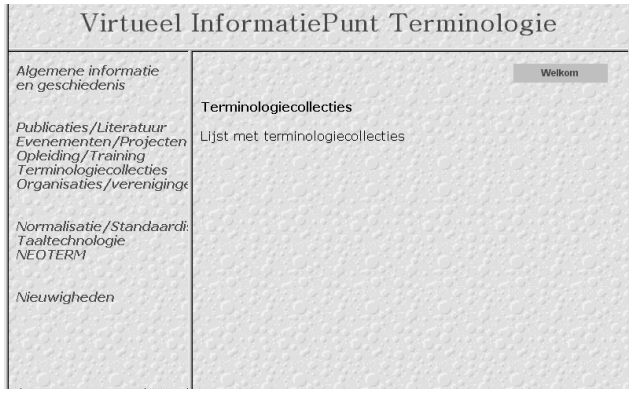
1.1. Prototype

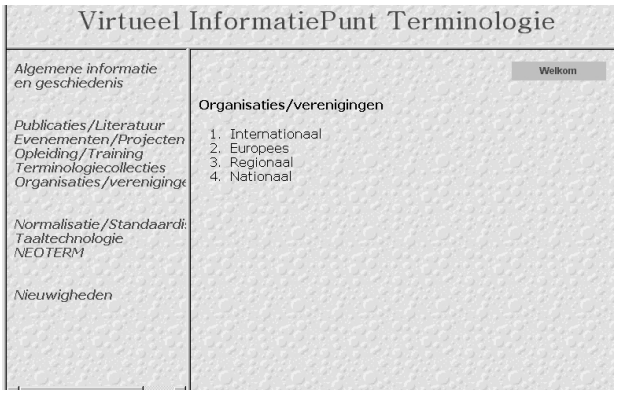


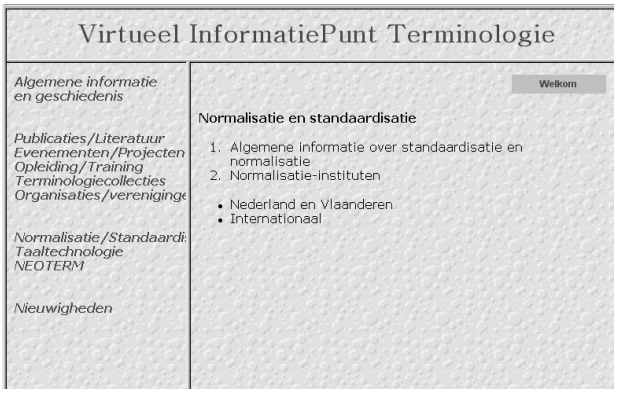


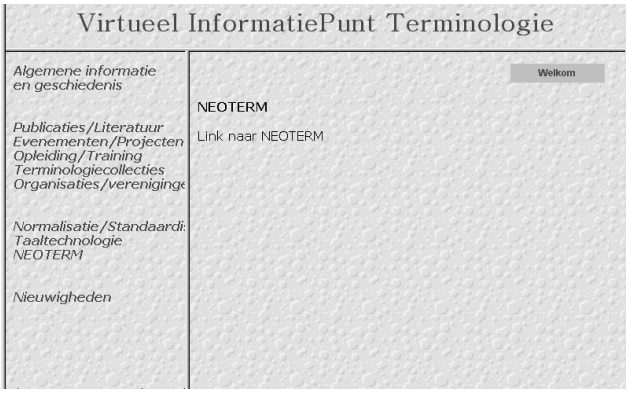


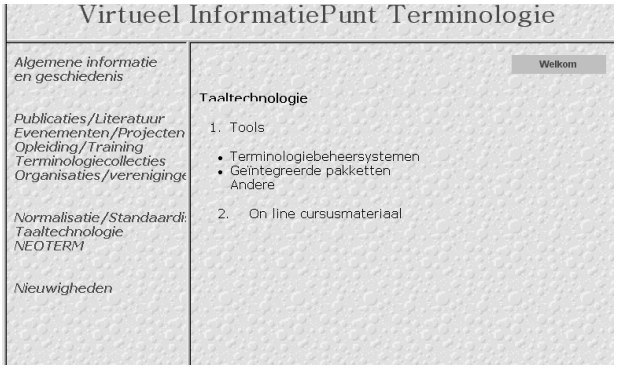












If the VIPTerm project and analogous projects such as DTP (Deutsches Terminologie Portal) are succesful and can be continued, a larger European platform for terminology is within reach and terminology awareness among experts, professionals and users will grow.

The Workshop Programme

9:00-9:30	Rebecca Hwa, Philip Resnik, Amy Weinberg	*Breaking the Resource Bottleneck for Multilingual Parsing*
9:30-10:00	Aoife Cahill, Mairead McCarthy, Josef van Genabith, Andy Way	*Automatic Annotation of the Penn-Treebank with LFG F-Structure Information*
10:00-10:30	Kiril Simov, Milen Kouylekov, Alexander Simov	*Incremental Specialization of an HPSG-Based Annotation Scheme*
10:30-11:00	Bernd Bohnet, Stefan Klatt, Leo Wanner	*A Bootstrapping Approach to Automatic Annotation of Functional Information to Adjectives with an Application to German*
11:00-11:30	Coffee break	
11:30-12:00	Adam Lopez, Mike Nossal, Rebecca Hwa, Philip Resnik	*Word-Level Alignment for Multilingual Resource Acquisition*
12:00-12:30	Necip Fazil Ayan, Bonnie J. Dorr	*Generating a Parsing Lexicon from an LCS-Based Lexicon*
12:30-13:00	Alberto Lavelli, Bernardo Magnini, Fabrizio Sebastiani	*Building Thematic Lexical Resources by Bootstrapping and Machine Learning*
13:00-14:30	Lunch break	
14:30-15:00	Anja Belz	*Learning Grammars for Noun Phrase Extraction by Partition Search*
15:00-15:30	Fermín Moscoso del Prado Martín, Magnus Sahlgren	*An Integration of Vector-Based Semantic Analysis and Simple Recurrent Networks for the Automatic Acquisition of Lexical Representations from Unlabeled Corpora*
15:30-16:00	Pavel Kveton, Karel Oliva	*Detection of Errors in Part-Of-Speech Tagged Corpora by Bootstrapping Generalized Negative n-Grams*
16:00-16:30	Coffee break	
16:30-17:00	Rayid Ghani, Rosie Jones	*A Comparison of Efficacy and Assumptions of Bootstrapping Algorithms for Training Information Extraction Systems*
17:00-17:30	Marisa Jiménez	*Using Decision Trees to Predict Human Nouns in Spanish Parsed Text*
17:30-18:00	Laura Alonso, Irene Castellón, Lluís Padró	*X-Tractor: A Tool for Extracting Discourse Markers*

Workshop Organisers

Alessandro Lenci	Università di Pisa, Italy
Simonetta Montemagni	Istituto di Linguistica Computazionale - CNR, Italy
Vito Pirrelli	Istituto di Linguistica Computazionale - CNR, Italy

Workshop Programme Committee

Table of Contents

Author Index

Preface

Provision of large-scale language resources, such as tagged corpora, lexicons and repositories of pre-classified text documents, is a crucial key to steady progress in an extremely wide spectrum of research, technological and business areas in the HLT sector. The continuously changing demands for language-specific and application-dependent annotated data (*e.g.* at the syntactic or at the semantic level), indispensable for design validation and efficient software prototyping, however, are daily confronted by the *resource bottleneck*. Handcrafted resources are often too costly and time-consuming to be produced at a sustainable pace, and, in some cases, they even exceed the limits of human conscious awareness and descriptive capability. The problem is even more acutely felt for low-resource languages, since the early stages of language resource development often require gathering considerable momentum both in terms of know-how and level of funding, of the order of magnitude normally deployed by large national projects.

Possible ways to circumvent, or at least minimise, these problems come from the literature on automatic knowledge acquisition and, more generally, from the machine-learning community. Of late, a number of machine learning algorithms have proved to fare reasonably well in the task of incrementally bootstrapping newly annotated data from a comparatively small sample of already annotated resources. Another promising route consists in automatically tracking down recurrent knowledge patterns in relatively unstructured or implicit information sources (such as free texts or machine readable dictionaries) for this information to be moulded into explicit representation structures (e.g. subcategorization frames, syntactic-semantic templates, ontology hierarchies etc.). In a similar vein, several strategies have been investigated aimed at merging or integrating structured information sources into a unitary comprehensive resource, or at customising general-purpose knowledge-bases for them to be of use in more technical domains. Finally, the growing availability of multi-lingual parallel resources has prompted the idea of using a high-resource language (generally English) to fertilize a low-resource language.

We believe that all these attempts at bootstrapping annotated language data are not only of practical interest, but also point to a bunch of germane theoretical issues. Gaining insights into the deep interrelation between representation and acquisition issues is likely to have significant repercussions on the way linguistic resources will be designed, developed and used for applications in the years to come. As the two aspects of knowledge representation and acquisition are profoundly interrelated, progress on both fronts can only be achieved, in our view of things, through a full appreciation of this deep interdependency.

The papers contained in this volume (13 out of 20 submissions) significantly confirm this general view. They focus on a variety of bootstrapping techniques to show their full potential for the provision of annotated language data. In particular, three areas of investigation are dealt with in some detail (often concurrently in the same paper). At the level of corpus annotation, parsers and annotated texts are increasingly used as a tightly integrated resource. Parsing robustness is no longer an end in itself but forms part of a virtuous incremental circle whereby finer grained, more accurate or simply more explicit levels of text analysis are built probabilistically on the basis of either under-specified or possibly more compact annotations. This process proves to be able to provide increasingly richer annotated data and sheds considerable light on the issue of inter-annotation translatability. At the level of lexicon design and building, a lot of effort is being put into merging complementary levels of language information (e.g. semantic and syntax, or semantic and morphology) to produce better lexical repositories for parsing and better computational models of the internalised lexical competence of a speaker. This strikes us as an extremely promising route, bound to throw in sharper relief the importance of simultaneously dealing with more information levels in parsing real texts at the level of accuracy required by HLT applications. Emphasis on the use of available machine learning technology for dealing with the novel challenges of HLT applications is the third research prong of this volume. Current and future needs for information extraction, classification and management appear to impose novel requirements on text processing and create novel tasks in the HLT sector. Once more, machine-learning approaches play an important role here. Perhaps even more significantly, these tasks provide, in turn, a key to a deeper understanding of the implicit assumptions underlying different machine-learning techniques.

We would like to thank all the authors who showed their interest by submitting papers to the workshop. We would also like to thank the members of the programme committee who kindly contributed to the reviewing process and the scientific and programme committees of LREC 2002.

Alessandro Lenci
<alessandro.lenci@ilc.cnr.it>
Università di Pisa, Italy

Simonetta Montemagni
<simonetta.montemagni@ilc.cnr.it>
Vito Pirrelli
<vito.pirrelli@ilc.cnr.it>
Istituto di Linguistica Computazionale - CNR, Italy

Breaking the Resource Bottleneck for Multilingual Parsing

Rebecca Hwa[1], Philip Resnik[1,2], and Amy Weinberg[1,2]

Institute for Advanced Computer Studies[1]
Department of Linguistics[2]
University of Maryland, College Park, MD 20742
{hwa, resnik, weinberg}@umiacs.umd.edu

Abstract

We propose a framework that enables the acquisition of annotation-heavy resources such as syntactic dependency tree corpora for low-resource languages by importing linguistic annotations from high-quality English resources. We present a large-scale experiment showing that Chinese dependency trees can be induced by using an English parser, a word alignment package, and a large corpus of sentence-aligned bilingual text. As a part of the experiment, we evaluate the quality of a Chinese parser trained on the induced dependency treebank. We find that a parser trained in this manner out-performs some simple baselines inspite of the noise in the induced treebank. The results suggest that projecting syntactic structures from English is a viable option for acquiring annotated syntactic structures quickly and cheaply. We expect the quality of the induced treebank to improve when more sophisticated filtering and error-correction techniques are applied.

1 Introduction

There is a substantial disparity between the quality of state of the art parsers available for English and those for other languages. English parsers such as those of Collins (1997) and Charniak (1999) were trained on hand annotated corpora such as the Penn Treebank Project (Marcus et al., 1993). However, experience has shown us that building hand-crafted treebanks from scratch is too time-consuming to be repeated for every language of interest. This bad news can be mitigated by leveraging English annotations to automatically acquired annotations for new languages. Recent work by Yarowsky and Ngai (2001) has shown that this type of transfer is possible for inducing part-of-speech tags for Chinese. In this paper, we explore the application of this technique to the more complex problem of inducing Chinese dependency trees.

The input to our system is a collection of sentence-aligned bilingual text (i.e., pairs of sentences that are translations of each other). Each English sentence is parsed using a high-quality English parser. For each pair of sentences, word alignment is performed using statistical MT models (Brown et al., 1990; Al-Onaizan et al., 1999). The alignment then anchors the projection of the English tree to the Chinese side (see Figure 1).

This paper presents an initial large-scale experiment, investigating the feasibility of inducing a Chinese dependency treebank using our projection algorithm and of training a parser on the resulting treebank. Due to the compounded errors of various components of the system, the induced Chinese dependency treebank is rather noisy. Applying filtering heuristics to the treebank improves its quality enough such that the parser trained on it out-performs some simple baselines. While the parser's performance is still significantly less than that of a parser trained on a clean, fully annotated (Chinese) treebank, this study suggests that projecting syntactic structures from English is viable for acquiring annotated syntactic structures quickly and cheaply.

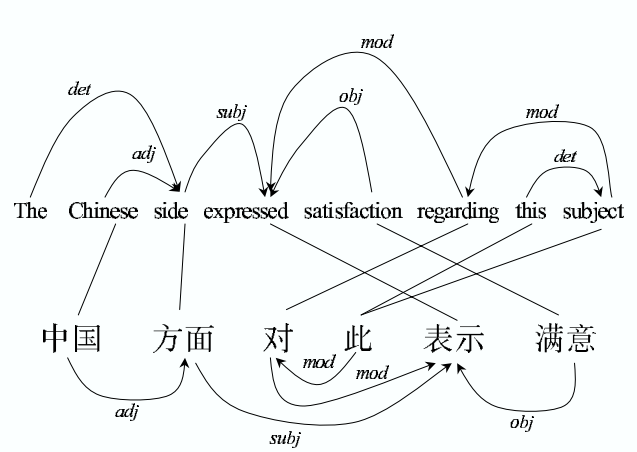


Figure 1: Given an English dependency parse tree and a set of word alignments, we infer the syntactic structure on the Chinese side via projection from its English counterpart.

2 Overview of the Algorithm

Our approach requires three resources. First, we need a sizable, sentence-aligned bilingual text as training corpus. In our experiment, we use a bilingual text of English and Chinese news articles. In Section 5 we discuss other ways in which bilingual text can be acquired and sentence aligned. Second, we require dependency parses of the English text. Our choice of dependency representation is motivated in Section 2.1. Third, word alignments are needed to relate the sentence pair on the lexical level. In this paper, we use alignments produced as a side-effect of training a statistical translation model (Brown et al., 1990; Al-Onaizan et al., 1999).

Given these resources, our system behaves as follows: for each sentence pair (E, C) in the bilingual text, the English sentence E is parsed and converted into a dependency representation. Next, word alignment is performed for the

sentence pair. Finally, the English dependency analysis is projected across the word alignment to the Chinese side according to our *Direct Projection Algorithm*, which we outline in section 2.2.

2.1 Dependency Representations as Transfer Medium

Dependency relationships specify asymmetric binary relations between two surface words: a *head* and its *modifier*. For example, in the sentence from Figure 1, *"The Chinese side expressed satisfaction regarding this subject,"* the word *side* modifies the head word *expressed*. The dependency links may optionally be annotated with information specifying grammatical relations between constituents such as *subject*, *object*, *modifier*, etc. In our example, the link between *side* and *expressed* is labeled as *subj*, indicating that the constituent *The Chinese side* is the subject of the verb *expressed*. In this section, we argue that dependency representation is right for our projection framework because it captures both structural and lexical relationships between words that are not string local; because it overcomes some of the shortcomings of evaluating against the phrase structure representation; and because it is language independent with respect to word order variations.

Syntactic analysis in terms of phrase structure has been the dominant paradigm in natural language processing, starting from early context-free grammars and continuing up to present-day stochastic formalisms. It is preferable over models that make Markov assumptions restricting interactions among words to those that occur within the window of an n-gram. Phrase structure formalisms provide a level of representation that allows significant constraint to occur between grammatical categories that are not string-local. These categories become *local* at the phrase structure level. For example, consider the following sentence from the Brown Corpus:

> *The largest hurdle the Republicans would have to face is a state law which says that before making a first race, one of two alternative courses must be taken.*

The relationship between *hurdle* and *is* exists over a long string-distance, owing to an embedded relative clause, and, similarly, *Republicans* and *face* are separated in the string by a sequence of auxiliaries and the infinitival *to*. As a result, the relationships represented in the sentence are not captured well by any n-gram model with tractable n. In contrast, the relationship between the subject NP and the predicate is easily encoded locally within a context-free rule such as S $\rightarrow$ NP VP.

To take full advantage of such relationships in models based on phrase structure, however, it is necessary to *lexicalize* the grammar formalism, so that lexically-based constraints are also localized within grammar rules. By incorporating lexical content into phrase structure rules (e.g., S(is) $\rightarrow$ NP(hurdle) VP(is)), lexicalized grammar formalisms make it possible to capture syntactic constraints such as as number agreement (e.g. the low probability of S(are) $\rightarrow$ NP(hurdle) VP(are)) as well as semantic constraints (e.g. the reasonably high probability of S(face) $\rightarrow$ NP(Republicans) VP(face)). Work taking advantage of this insight (e.g. Collins (1997; Charniak (1999)) has defined the breakthroughs leading to the current state of the art in broad-coverage parsing. Implicitly or sometimes explicitly (as in the work of Collins), what gives lexicalized context-free representations their power is the ability to probabilistically model the syntactic *dependency* relationships between words in the structure.

Moreover, dependency analysis evaluation avoids some of the shortcomings of constituency analysis evaluation (Lin, 1995; Carroll et al., 1999). Standard constituency parsing metrics compare the phrase boundaries specified by the gold standard to that of the candidate analysis. They also evaluate whether conditions on well formed trees (such as a ban on crossing branches) are respected by the candidate. However, as Lin (1995) notes, since branching structure is not directly tied to semantic interpretation, it is unclear how to interpret missing, spurious, or crossing branches. On the other hand, it is apparent that syntactic dependencies, more so than syntactic constituents, are closely tied to the who-did-what-to-whom relationships of language. Indeed, work in lexical semantics relating syntactic representations to thematic relationships such as *agent*, *theme*, *beneficiary*, has focused primarily on syntactic dependencies rather than on phrasal constituents (Baker, 1997). Since semantic dependencies form a superset based on syntactic dependencies, we are better able to gauge how likely a representation is to be interpretable, by measuring the percentage of correct dependencies.

Finally, dependency structures firmly separate precedence from dominance relations, such that word order variation between languages becomes less of a problem than in constituency trees. For example, the relative string order of a series of modifiers of a head is irrelevant in the dependency representation. All are modifiers. By contrast, a constituency tree may require a stacked structure that would not translate well if the word order were reversed in another language. In other words, dependency structures are more likely to respect a homomorphism.

These observations suggest that dependencies may be a better choice for syntactic projection across languages than phrasal constituents. To the extent that this assumption is correct, we should be able to use word alignments as a bridge between English and another language, retaining some level of confidence that if dependencies are projected across the alignment they will be correct for the new language. Experimental results from our previous work (Hwa et al., 2002), have indicated that while the assumption does not always hold true, syntactic analyses projected from English to Chinese can, in principle, yield Chinese analyses that are nearly 70% accurate (in terms of unlabeled dependencies) after application of a set of linguistically principled rules.[1]

2.2 The Direct Projection Algorithm

Our approach is based on the intuitive idea of a direct projection of dependency structures. We now describe our

[1]The experiment was performed under idealized settings, projecting human annotated English dependency analyses using human annotated word alignments.

projection algorithm in more detail. Given sentence pair (E, C), where $E = e_1, \ldots, e_n$ and $C = c_1, \ldots, c_m$, syntactic relations (denoted as $R(x, y)$) are projected from English for the following situations:

- **one-to-one** if e_i is aligned with a unique c_x and e_j is aligned with a unique c_y, if $R(e_i, e_j)$, conclude $R(c_x, c_y)$.
- **unaligned (English)** if e_j is not aligned with any word in C, then create a new empty word c_y such that for any e_i aligned with a unique c_x, $R(e_i, e_j) \Rightarrow R(c_x, c_y)$ and $R(e_j, e_i) \Rightarrow R(c_y, c_x)$.
- **one-to-many** if $e_i$is aligned with $c_x, \ldots, c_y$, then create a new empty word c_z such that c_z is the parent of $c_x, \ldots, c_y$ and set e_i to align to c_z instead. We called this a *Multiply-Aligned Component*, or MAC.
- **many-to-one** if $e_i, \ldots, e_j$ are all uniquely aligned to c_x, then delete all alignments between $e_k (i \leq k \leq j)$ and c_x except for the head of $e_i, \ldots, e_j$.

The **many-to-many** case is decomposed into a two-step process: first perform one-to-many, then perform many-to-one. In the cases of unaligned Chinese words, they are left out of the projected syntactic tree. The asymmetry of the treatment of **one-to-many** and **many-to-one** and of the unaligned words for the two languages arises from the asymmetric nature of the projection.

2.2.1 Post-Projection Transformation

The Direct Projection Algorithm by itself does not produce good dependency trees because it does not properly handle structural projection for the more complex cases when the alignment is not one-to-one. Therefore, we apply a small set of linguistically motivated rules to correct the projected trees as a post-hoc process. It is clearly an advantage to limit the correction rules to those that can apply generally, across many construction types. Wanting to avoid unending language-specific rule tweaking, we strictly limited the possible rules. Rules were permitted to refer only to closed class items, to parts of speech projected from the English analysis, or to easily enumerated lexical categories (e.g. {*dollar*, RMB, $, yen}). The majority of rule patterns are variations on the same solution to the same problem. Viewing the problem from a higher level of linguistic abstraction made it possible to find all the relevant cases in a short time and express the solution compactly; in all, fewer than twenty rules were written, and the analysis, rule writing, and verification of their correctness using the data set took a few days.

Here are two examples of the rules we developed; see (Hwa et al., 2002) for fuller discussion.

Rule for noun modification:

- If $c_x, \ldots, c_y$ are a set of Chinese words aligned to an English noun, replace the empty node introduced in the Direct Projection Algorithm by promoting the last word c_y to its place with $c_x, \ldots, c_{y-1}$ as dependents.

Rule for aspectual markers:

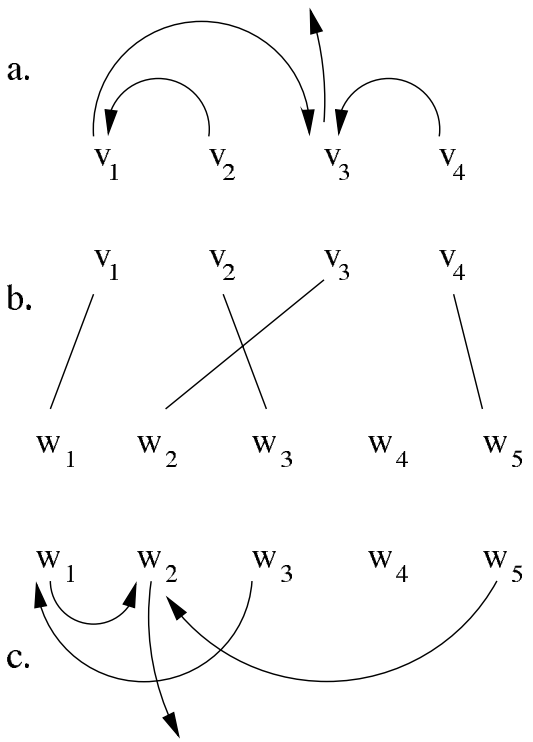


Figure 2: The direct projection of the dependency parse for $v_1 \ldots v_4$ (Figure 2a) across the word alignment (Figure 2b) results in cross dependency relationships for the link between w_1 and w_3 and the link between w_2 and w_5; and it leaves word w_4 unattached to the projected dependency tree (Figure 2c).

- If $c_x, \ldots, c_y$, a sequence of Chinese words aligned with English verbs, is followed by c_a, an aspect marker, make c_a into a modifier of the last verb c_y.

2.2.2 Remaining Shortcomings of the Direct Projection Algorithm

Although the majority of the projected trees are significantly improved, the post-projection transformation rules still do not adequately address some major deficiencies of the Direct Projection Algorithm. The algorithm does not ensure that the projected structure is indeed a well-formed structure. Thus, when given unconstrained word alignment outputs, the projected structure may contain errors such as crossing dependencies (see Figure 2). Moreover, due to the asymmetry of the algorithm, the syntactic role of unaligned foreign words cannot be inferred. The post-projection transformation rules address this problem to some extent by incorporating unaligned function words back into the parse, but an intelligent treatment of the open class of unaligned words remains a challenge of this projection approach. Furthermore, the algorithm does not address complex translation divergences (Dorr, 1993), such as the head-swapping phenomenon (in which the direction of the head-modifier dependency is reversed in the foreign language). Lopez et al. (2002) describe an alternative to the direct projection approach that addresses some of these problems.

3 Experimental Setup

Our previous results have shown that, given good English parses and clean alignments to Chinese translations, the direct projection approach from English to Chinese (together with post-processing) can lead to Chinese annotations that are substantially correct; unlabeled precision/recall on projected dependencies approaches 70% (Hwa et al., 2002). While this demonstrates that the approach holds promise in automatically inducing syntactic treebanks of reasonable quality, it is not clear how much degradation occurs when using imperfect English parsers and imperfect word alignment models. That question is our focus in this paper. We report a full-scale experiment on English and Chinese sentence pairs, evaluating the entire framework under the realistic settings of imperfect bilingual data and error-prone parsers and alignment models (see Section 3.1). Once a Chinese dependency treebank is induced, we use it to train a Chinese parser in a manner similar to that of Collins (1999). The trained parser is then evaluated on unseen test sentences taken from the Chinese Treebank (Xia et al., 2000) and compared with two baselines and an upper bound.

3.1 Resources

We use about 56,000 sentence pairs from the Hong Kong News (HKNews) corpus as our bilingual text. The data have been automatically sentence aligned and the Chinese words have been automatically segmented.[2] To parse the English sentences, we use a lexicalized statistical parser trained on the Wall Street Journal corpus (Collins, 1997).[3] To obtain word alignments for all sentence pairs, we train an off-the-shelf statistical translation model, GIZA++ (Al-Onaizan et al., 1999), using the HKNews bilingual text. Given these resources, the direction projection algorithm and the post-projection transformation process are then used to induce dependency trees for the Chinese sentences in the HKNews corpus.

3.2 Evaluation of the Induced Treebank

Because of its size, we do not directly assess the quality of the induced treebank. Instead, we evaluate the Chinese parser trained from it. To the extent that the trained parser outputs reasonable structures on unseen test sentences, it indicates that the induced treebank is a useful resource. To evaluate the quality of the trained parser, we compare it to two simple baseline dependency analyses: *always modify the previous word*, and *always modify the next word*. As an upper bound, we have also trained the same parser with clean, hand-annotated trees from the Penn Chinese Treebank (ChTB). We constructed a development set consisting of 124 sentences and a test set consisting of 88 sentences taken from the Chinese Treebank; all sentences are of 40 words or less. The remaining approximately 3800 Chinese Treebank sentences are converted into their dependency representation (similar to the algorithm described in Section 2 of the paper by Xia and Palmer (2001)) and used as training data for the upper-bound parser. We evaluate the trained parser by comparing its output (dependency) parse trees for the unseen test sentences against the human-annotated gold standard parse trees (also converted to dependency representation). The metrics used are the precision and recall scores on the unlabeled dependency relations. A parser produced dependency link is considered "correct" if the same head-modifier relationship exists in the gold standard; the dependency label does not need to match. Punctuations are not scored.

[2] We are grateful to Stefan Vogel of CMU for his assistance with this corpus.

[3] The executable of the parser is freely available at ftp://ftp.cis.upenn.edu/pub/mcollins/misc.

4 Results and Discussions

Tables 1 and 2 show performance comparisons for our automatic projection approach as compared to the lower and upper bounds. As one might expect, the quality of the treebank induced under the real-world constraints of imperfect data and components is noticeably worse than one induced using clean English parses and perfect word alignments. The Direct Projection Algorithm and its associated post-projection transformation rules are not fault-tolerant enough to recover from the compounding errors of the parser and alignment model. Without further processing, the projected treebank would contain too much noise to be useful for training a parser. Therefore, our attentions turn to filtering heuristics for poorly induced dependency trees.

We found that the most unreliable component is the word alignment model. A cursory inspection of the alignment output (for the HKNews corpus) shows that, for many sentences, the majority of the English words remain unaligned; and that often, an unusually high number of Chinese words (e.g, five or greater) are aligned to the same English word. The poor alignment output may have many causes: in particular, the sentence pair input to the alignment model is imperfect, and the alignment model does not perform well for language pairs with very dissimilar word-order patterns.

This suggests that performance might improve if we filter out sentence pairs that are known to be poorly aligned. To filter out dependency trees projected from dubious word alignments, we have devised several simple heuristics. First, we removed those sentences for which more than 30% of the English words were not aligned to any Chinese word ($EnoC \leq 0.3$). The figure 30% is empirically determined, based on the trained parser's performance on the development set. As shown in the first row of Table 1, the parser trained on the filtered treebank does outperform the modify-next baseline; however, the corpus size has been drastically cut-down from around 56,000 to less than 8,000. The second filter we apply to the corpus is to remove sentences in which the size of a multiply aligned component is greater than three (MAC > 3); that is, when more than three Chinese words are aligned to the same English word. The MAC value of 3 was also determined empirically using development data. The second line of Table 1 shows that training the parser on the induced treebank filtered by both heuristics leads to further improvement. Finally, we return to the crossing-dependency problem alluded to earlier in section 2.2.2. While we do not correct the crossing depen-

Method	Corpus Size	Precision & Recall
EnoC	7689	37.4
EnoC+MAC	5525	42.1
EnoC+MAC+NoCross	5284	42.9
Modify Prev (Baseline)	–	14.0
Modify Next (Baseline)	–	32.2

Table 1: The parser's performance on the development set (%) when the training corpus has been filtered with the following heuristics: remove sentences if too many English words have no Chinese translations (EnoC); remove sentences if too many Chinese words are aligned to one English word (MAC); remove sentences that violate many crossing-dependency constraints (NoCross).

dencies in this work, we remove sentences with the most egregious crossing-dependency violations in their analyses. Our experiments with development data suggested that a sentence should be filtered out if more than 40% of its dependency links violate the no-crossing constraint. The combination of the three filters improved the induced treebank so that a parser trained on the treebank outperforms the simple baselines; however, the draconian filters also reduced the corpus from 56,000 sentences to slightly over 5,000.

Table 2 shows the trained parser's performance on a separate test set. As before, it is compared with two baselines; and as an upper bound, we train the same parser on a clean, manually created treebank.[4] Similar to the outcome of the development set, the trained parser performs better than the baseline, but it still cannot compete with a parser trained on a clean corpus. It is interesting to note that after our current filtering techniques, the sizes of the induced treebank is comparable to the clean one. However, our method of treebank acquisition is not constrained by the laborious manual annotation process; therefore it would be easy for us to obtain a much larger bilingual corpus as a starting point, as discussed below. We conjecture that the size of the corpus will help offset the effect of the noise, as will more sophisticated sampling techniques that exclude the noisiest data.

5 Conclusion and Future Work

In this paper, we have described our framework for acquiring Chinese dependency treebanks by bootstrapping from existing linguistic resources for English. We have explicitly discussed the assumptions made and the resources required in order for our algorithm to work. An ambitious full-scale experiment using real-world data was performed to investigate the feasibility of our approach. Our results suggest that treebank acquisition through projection is indeed possible; however reducing the noise in the induced treebank is a major challenge.

This finding points us to several directions for further research. One clear avenue is to obtain larger bilingual texts, so that more data remain even when noisy sentence pairs have been filtered out. Work on mining the Web for bilingual text, such as STRAND (Resnik, 1999), BITS (Ma and Liberman, 1999), and PTMiner (Nie et al., 1999), show significant promise in this regard. Once parallel Web pages are obtained, it is possible to obtain sentence- or segment-level alignments either via alignment of HTML markup (Resnik, 1998) or via more sophisticated sentence-alignment techniques (Melamed, 1998).

Beyond simply taking a "more is better" approach to data acquisition, one way to reduce the noise in the induced treebank is to lower the error rates of the individual components in our projection framework. Of these, improving the word alignment model would benefit the overall system the most. We are actively developing alternative word alignment models that is sensitive to this syntactic projection framework (Lopez et al., 2002). Moreover, as we have shown in this study, filtering techniques that identify and remove malformed trees can help reducing noise; however, aggressive filtering alone is likely to result in over-filtering. To render nearly 90% of the bilingual text useless places too heavy a burden on even the best Web mining techniques. We are experimenting with filtering strategies that attempt to localize the potentially problematic parts of a syntactic tree so that the rest can still contribute to the training corpus. In addition, we are continuing to work on the post-projection transformation the process to improve the quality of the projected trees.

[4]The upper-bound parser's performance is on par with that of the state of the art constituency parsers trained on the Chinese Treebank, e.g. (Bikel and Chiang, 2000).

6 Acknowledgments

This work has been supported, in part, by ONR MURI Contract FCPO.810548265, NSA RD-02-5700, DARPA/ITO Cooperative Agreement N660010028910 and Mitre Contract 010418-7712. The authors would like to thank Franz Josef Och for his help with using the GIZA++ translation model; and Adam Lopez and Mike Nossal for helpful discussions and comments on this paper.

7 References

Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, I. Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation. Technical report, JHU. citeseer.nj.nec.com/al-onaizan99statistical.html.

Mark C. Baker, 1997. *Thematic Roles and Syntactic Structure*, pages 73–137. Kluwer.

Daniel Bikel and David Chiang. 2000. Two statistical parsing models applied to the chinese treebank. In *Proceedings of the Second Chinese Language Processing Workshop*, pages 1–6.

Method	Corpus	Size	Precision & Recall
Modify Prev (Baseline)	–	–	13.5
Modify Next (Baseline)	–	–	35.7
Stat. Parser	Induced HKNews	5284	42.3
Stat. Parser (Upper-bound)	Clean ChTB	3870	75.6

Table 2: A comparison of the parsers' performance against lower and upper bounds on the test set (%).

Peter F. Brown, John Cocke, Stephen A. DellaPietra, Vincent J. DellaPietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, June.

John Carroll, Guido Minnen, and Ted Briscoe. 1999. Corpus annotation for parser evaluation. In *LINC-99 workshop at the 9th Conference of the EACL*, June.

Eugene Charniak. 1999. A maximum-entropy inspired parser. Technical Report CS-99-12, Brown University.

Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the ACL*, pages 16–23, Madrid, Spain.

Michael Collins. 1999. A statistical parser for czech. In *Proceedings of the 37th Annual Meeting of the ACL*, College Park, Maryland.

Bonnie J. Dorr. 1993. *Machine Translation: A View from the Lexicon.* The MIT Press, Cambridge, MA.

Rebecca Hwa, Philip Resnik, Amy Weinberg, and Okan Kolak. 2002. Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting of the ACL.* To appear.

Dekang Lin. 1995. A dependency-based method for evaluating broad-coverage parsers. In *Proceedings of the IJCAI-95*, pages 1420–1425.

Adam Lopez, Michael Nossal, Rebecca Hwa, and Philip Resnik. 2002. Word-level alignment for multilingual resource acquisition. In *Proceedings of the Workshop on Linguistic Knowledge Acquisition and Representation: Bootstrapping Annotated Language Data.* To appear.

Xiaoyi Ma and Mark Liberman. 1999. Bits: A method for bilingual text search over the web. In *Machine Translation Summit VII.*

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

I. Dan Melamed. 1998. *Empirical Methods for Exploiting Parallel Texts.* Ph.D. thesis, University of Pennsylvania, May.

J. Nie, M. Simard, P. Isabelle, and R. Durand. 1999. Cross-language information retrieval based on parallel texts and automatic mining parallel texts from the web. In *Proceedings of the ACM SIGIR Conference.*

Philip Resnik. 1998. Parallel strands: A preliminary investigation into mining the Web for bilingual text. In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas, AMTA-98, in Lecture Notes in Artificial Intelligence, 1529*, Langhorne, PA, October 28-31.

Philip Resnik. 1999. Mining the web for bilingual text. In *Proceedings of the 37th Annual Meeting of the ACL*, June.

Fei Xia and Martha Palmer. 2001. Converting dependency structures to phrase structures. In *Proc. of the HLT Conference*, March.

Fei Xia, Martha Palmer, Nianwen Xue, Mary Ellen Ocurowski, John Kovarik, Fu-Dong Chiou, Shizhe Huang, Tony Kroch, and Mitch Marcus. 2000. Developing guidelines and ensuring consistency for chinese text annotation. In *Proceedings of the Second Language Resources and Evaluation Conference*, June.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. In *Proc. of NAACL-2001*, pages 200–207.

Automatic Annotation of the Penn-Treebank with LFG F-Structure Information

Aoife Cahill, Mairead McCarthy, Josef van Genabith, Andy Way

School of Computer Applications, Dublin City University
Dublin 9, Ireland
{acahill, mcarthy, josef, away}@computing.dcu.ie

Abstract

Lexical-Functional Grammar f-structures are abstract syntactic representations approximating basic predicate-argument structure. Treebanks annotated with f-structure information are required as training resources for stochastic versions of unification and constraint-based grammars and for the automatic extraction of such resources. In a number of papers (Frank, 2000; Sadler, van Genabith and Way, 2000) have developed methods for automatically annotating treebank resources with f-structure information. However, to date, these methods have only been applied to treebank fragments of the order of a few hundred trees. In the present paper we present a new method that scales and has been applied to a complete treebank, in our case the WSJ section of Penn-II (Marcus et al, 1994), with more than 1,000,000 words in about 50,000 sentences.

1. Introduction

Lexical-Functional Grammar f-structures (Kaplan and Bresnan, 1982; Bresnan, 2001) are abstract syntactic representations approximating basic predicate-argument structure (van Genabith and Crouch, 1996). Treebanks annotated with f-structure information are required as training resources for stochastic versions of unification and constraint-based grammars and for the automatic extraction of such resources. In two companion papers (Frank, 2000; Sadler, van Genabith and Way, 2000) have developed methods for automatically annotating treebank resources with f-structure information. However, to date, these methods have only been applied to treebank fragments of the order of a few hundred trees. In the present paper we present a new method that scales and has been applied to a complete treebank, in our case the WSJ section of Penn-II (Marcus et al, 1994), with more than 1,000,000 words in about 50,000 sentences.

We first give a brief review of Lexical-Functional Grammar. We next review previous work and present three architectures for automatic annotation of treebank resources with f-structure information. We then introduce our new f-structure annotation algorithm and apply it to the Penn-II treebank resource. Finally we conclude and outline further work.

2. Lexical-Functional Grammar

Lexical-Functional Grammar (LFG) is an early member of the family of unification- (more correctly: constraint-) based grammar formalisms (FUG, PATR-II, GPSG, HPSG etc.). It enjoys continued popularity in theoretical and computational linguistics and natural language processing applications and research. At its most basic, an LFG involves two levels of representation: c-structure (constituent structure) and f-structure (functional structure). C-structure represents surface grammatical configurations such as word order and the grouping of linguistic units into larger phrases. The c-structure component of an LFG is represented by a CF-PSG (context-free phrase structure grammar). F-structure represents abstract syntactic functions such as subject, object, predicate etc. in terms of recursive attribute-value structure representations. These abstract syntactic representations abstract away from particulars of surface configuration. The motivation is that while languages differ with respect to surface representation they may still encode the same (or very similar) abstract syntactic functions (or predicate argument structure). To give a simple example, typologically, English is classified as an SVO (subject-verb-object) language while Irish is a verb initial VSO language. Yet a sentence like *John saw Mary* and its Irish translation *Chonaic Seán Máire*, while associated with very different c-structure trees, have structurally isomorphic f-structure representations, as represented in **Figure 1**.

C-structure trees and f-structures are related in terms of projections (indicated by the arrows in the examples in **Figure 1**). These projections are defined in terms of f-structure annotations in c-structure trees (describing f-structures) originating from annotated grammar rules and lexical entries. A sample set of LFG grammar rules with functional annotations (f-descriptions) is provided in **Figure 2**. Optional constituents are indicated by brackets.

3. Previous Work: Automatic Annotation Architectures

It would be desirable to have a treebank annotated with f-structure information as a training resource for probabilistic constraint (unification) grammars and as a resource for extracting such grammars. The large number of CFG rule types in treebanks ($> 19,000$ for Penn-II) makes manual f-structure annotation of grammar rules extracted from complete treebanks prohibitively time consuming and expensive. Recently, in two companion papers (Frank, 2000; Sadler, van Genabith and Way, 2000) a number of researchers have investigated the possibility of automatically annotating treebank resources with f-structure information. As far as we are aware, we can distinguish three different types of automatic f-structure annotation architectures (these have all been developed within an LFG framework and although we refer to these as automatic f-structure an-

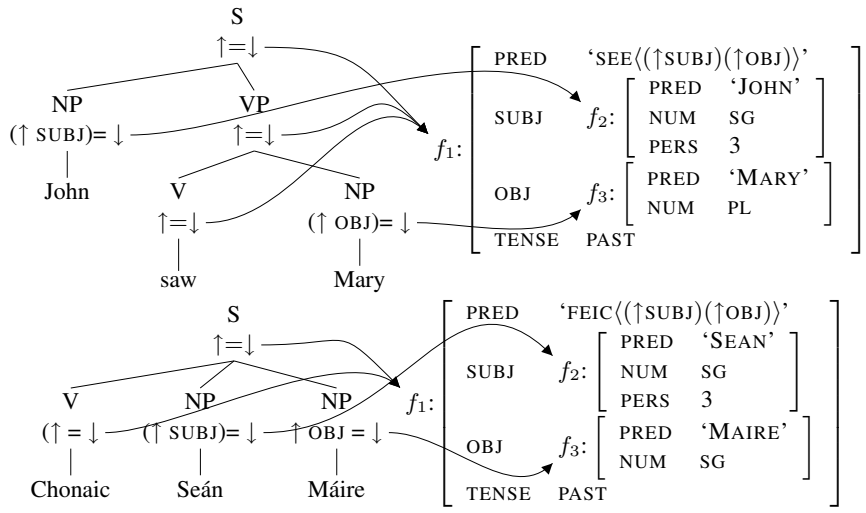


Figure 1: C- and f-structures for an English and corresponding Irish sentence

$$\text{S} \rightarrow \text{NP}_{\uparrow \text{SUBJ} = \downarrow} \quad \text{VP}_{\uparrow = \downarrow} \quad \left(\text{ADV}_{\downarrow \in \uparrow \text{ADJN}}\right)$$

$$\text{NP} \rightarrow \text{Det}_{\uparrow = \downarrow} \quad \text{N}_{\uparrow = \downarrow}$$

$$\text{VP} \rightarrow \text{V}_{\uparrow = \downarrow} \quad \left(\text{NP}_{\uparrow \text{OBJ} = \downarrow}\right) \quad \left(\text{VP}_{\uparrow \text{XCOMP} = \downarrow}\right) \quad \left(\text{S}_{\uparrow \text{COMP} = \downarrow}\right)$$

Figure 2: Sample LFG grammar rules for a fragment of English

notation architectures they could equally well be used to annotate treebanks with e.g. HPSG feature structure or with Quasi-Logical Form (QLF) (Liakata and Pulman, 2002) annotations):

- regular expression based annotation (Sadler, van Genabith and Way, 2000)
- tree description set based rewriting (Frank, 2000)
- annotation algorithms

More recently, we have learnt about the QLF annotation work by (Liakata and Pulman, 2002). Much like (Frank, 2000), their approach is based on matching configurations in a flat, set based tree description representation.

Below we will briefly describe the first two architectures. The new work presented in this paper is based on an annotation algorithm and discussed at length in Sections 4 and 5 of the paper.

3.1. Regular Repression Based Annotation

(Sadler, van Genabith and Way, 2000) describe a regular expression based automatic f-structure annotation methodology. The basic idea is very simple: first, the CFG rule set is extracted from the treebank (fragment); second, regular expression based annotation principles are defined; third, the principles are automatically applied to the rule set to generate an annotated rule set; fourth, the annotated rules are automatically matched against the original treebank trees and thereby f-structures are generated for these trees. Since the annotation principles factor out linguistic generalisations their number is much smaller than the number of CFG treebank rules. In fact, the regular expression based f-structure annotation principles constitute a principle-based LFG c-structure/f-structure interface. We will explain the method in terms of a simple example. Let us assume that from the treebank trees we extract CFG rules expanding `vp` of the form (amongst others):

```
vp:A > v:B s:C
vp:A > v:B v:C s:D
vp:A > v:B v:C v:D s:E
     ..
vp:A > v:B s:C pp:D
vp:A > v:B v:C s:D pp:E
vp:A > v:B v:C v:D s:E pp:F
     ..
vp:A > advp:B v:C s:D
vp:A > advp:B v:C v:D s:E
vp:A > advp:B v:C v:D v:E s:F
     ..
vp:A > advp:B v:C s:D pp:E
vp:A > advp:B v:C v:D s:E pp:F
vp:A > advp:B v:C v:D v:E s:F pp:G
```

Each CFG category in the rule set has been associated with a logical variable designed to carry f-structure information. In order to annotate these rules we can define a set of regular expression based annotation principles:

```
vp:A > * v:B v:C *
```

```
     @ [B:xcomp=C,B:subj=C:subj]
vp:A > *(~v) v:B *
     @ [A=B]
vp:A > * v:B s:C *
     @ [B:comp=C]
```

The first annotation principle states that if anywhere in a rule RHS expanding a `vp` category we find a `v v` sequence the f-structure associated with the second `v` is the value of an `xcomp` attribute in the f-structure associated in the first `v` ('`*`' is the Kleene star and, if unattached to any other regular expression, signifies any string). It is easy to see how this annotation principle matches many of the extracted example rules, some even twice. The second principle states that the leftmost `v` in `vp` rules is the head. The leftmost constraint is expressed by the fact that the rule RHS may consist of an initial string that may not contain a `v`: `*(~v)`. Each of the annotation principles is partial and underspecified: they underspecify CFG rule RHSs and annotate matching rules partially. The annotation interpreter applies all annotation principles to each CFG rule as often as possible and collects all resulting annotations. It is easy to see that we get, e.g., the following (partial) annotation for:

```
vp:A > advp:B v:C v:D v:E s:F pp:G
     @ [A=C,
        C:xcomp=D,C:subj=D:subj,
        D:xcomp=E,D:subj=E:subj,
        E:comp=F]
```

In their experiments with the publicly available subsection of the AP treebank, (Sadler, van Genabith and Way, 2000) achieve precision and recall results in the low to mid 90 percent region against a manually annotated "gold standard". The method is order independent, partial and robust. To date, however, the method has been applied to only small CFG rule sets (of the order of 500 rules approx.).

3.2. Rewriting of Flat Tree Description Set Representations

In a companion paper, (Frank, 2000) develops an automatic annotation method that in many ways is a generalisation of the regular expression based annotation method. The basic idea is again simple: first, trees in treebanks are translated into a flat set representation format in a tree description language; second, annotation principles are defined in terms of rewriting rules employing a rewriting system originally developed for transfer based machine translation architectures (Kay, 1999). We will illustrate the method with a simple example

```
     s:A
    /   \              dom(A,B), dom(A,C),
np:B    vp:C           dom(C,D), ..
  |       |      =>    pre(B,C),
John     v:D           cat(A,s), cat(C,vp),
           |           cat(D,v), ..
         left
```

```
dom(X,Y), dom(X,Z), pre(Y,Z),
cat(X,s), cat(Y,np), cat(Z,vp)
        ==>
subj(X,Y), eq(X,Z)
```

Trees are described in terms of (immediate and general) dominance and precedence relations, labelling functions assigning categories to nodes and so forth. In our example node identifiers `A`, `B`, etc. do double duty as f-structure variables. The annotation principle states that if node `X` dominates both `Y` and `Z` and if `Y` preceeds `Z` and the respective CFG categories are `s`, `np` and `vp` then `Y` is the subject of `X` and `Z` is the same as (i.e. is the head of) `X`.

The tree description rewriting method has a number of advantages:

- in contrast to the regular expression based method, annotation principles formulated in the flat tree description method can consider arbitrary tree fragments (and not just only local CFG rule configurations).
- in contrast to the regular expression based method which is order independent, the rewriting technology can be used to formulate both order dependent and order independent systems. Cascaded, order dependent systems can support a more compact and perspicuous statement of annotation principles as certain transformations can be assumed to have already applied earlier on in the cascade.

For a more detailed, joint presentation of the two approaches consult (Frank et al, 2002). Like the regular expression based annotation method, the tree description based set rewriting method has to date only been applied to small treebank fragments of the order of serveral hundred trees.

3.3. Annotation Algorithms

The previous two automatic annotation architectures enforce a clear separation between the statement of annotation principles and the annotation procedure. In the first case the annotation procedure is provided by our regular expression interpreter, in the second by the set rewriting machinery. A clean separation between principles and processing supports maintenance and reuse of annotation principles. There is, however, a third possible automatic annotation architecture and this is an annotation algorithm. In principle, two variants are possible. An annotation algorithm may

- directly (recursively) transduce a treebank tree into an f-structure – such an algorithm would more appropriately be referred to as a tree to f-structure transduction algorithm;
- annotate CFG treebank trees with f-structure annotations from which an f-structure can be computed by a constraint solver.

The first mention of an automatic f-structure annotation algorithm we are aware of is unpublished work by Ron Kaplan (p.c.) who as early as 1996 worked on automatically generating f-structures from the ATIS corpus to generate data for LFG-DOP (Bod and Kaplan, 1998) applications.

Kaplan's approach implements a direct tree to f-structure transduction. The algorithm walks the tree looking for different configurations (e.g. np under s, 2nd np under vp, etc.) and "folds" the tree into the corresponding f-structure. By contrast, our approach develops the second, more indirect tree annotation algorithm paradigm. We have designed and implemented an algorithm that annotates nodes in the Penn-II treebank trees with f-structure constraints. The design and the application of the algorithm is explained below.

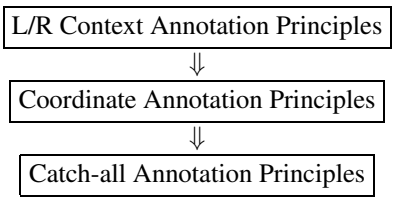
4. Automatic Annotation Algorithm Design

In our work on the automatic annotation algorithm we want to achieve the following objectives: we want an annotation method that is robust and scales to the whole of the Penn-II treebank with 19,000 CFG rules for 1,000,000 words with 50,000 sentences approx. The algorithm is implemented as a recursive procedure (in Java) which annotates Penn-II treebank tree nodes with f-structure information. The annotations describe what we call "proto-f-structures". Proto-f-structures

- encode basic predicate-argument-modifier structures;
- may be partial or unconnected (i.e. in some cases a sentence may be associated with two or more unconnected f-structure fragments rather than a single f-structure);
- may not encode some reentrancies, e.g. in the case of wh- and other movement or distribution phenomena (of subjects into VP coordinate structures etc.).

Compared to the regular expression and the set rewriting based annotation methods described above, the new algorithm is somewhat more coarse grained, both with respect to resulting f-structures and with respect to the formulation of the annotation principles.

Even though the method is encoded in the form of an annotation algorithm (i.e. a procedure) we did not want to completely hard code the linguistic basis for the annotation into the procedure. In order to achieve a clean design which supports maintainability and reusability of the annotation algorithm and the linguistic information encoded in it, we decided to design the algorithm in terms of three main components that work in sequence:



Each of the components of the algorithm is presented below.

In addition, at the lexical level, for each Penn-II preterminal category type, we have a lexical macro associating any terminal under the category with the required f-structure information. To give a simple example, a singular common noun nns, such as e.g. *company* is annotated by the lexical macro for nns as $\uparrow$ pred = company, $\uparrow$ num = sg, $\uparrow$ pers = 3rd.

4.1. L/R Context Annotation Principles

The annotation algorithm recursively traverses trees in a top-down fashion. Apart from very few exceptions (e.g. possessive NPs), at each stage of the recursion the algorithm considers local subtrees of depth one (i.e. effectively CFG rules). Annotation is driven by categorial and simple configurational information in a local subtree.

In order to annotate the nodes in the trees, we partition each sequence of daughters in a local subtree (i.e. rule RHS) into three sections: left context, head and right context. The head of a local tree is computed using Collins' Collins (1999) head lexicalised grammar annotation scheme (except for coordinate structures, where we depart from Collins' head scheme). In a preprocessing step we transform the treebank into head lexicalised form. During automatic annotation we can then easily identify the head constituent in a local tree as that constituent which carries the same terminal string as the mother of the local tree. With this we can compute left and right context: given the head constituent, the left context is the prefix of the local daughter sequence while the right context is the suffix. For each local tree we also keep track of the mother category. In addition to the positional (reduced to the simple tripartition into head with left/right context) and categorial information about mother and daughter nodes we also employ an LFG distinction between subcategorisable (subj, obj, obj2, obl, xcomp, comp ...) and non-subcategorisable (adjn, xadjn ...) grammatical functions. Subcategorisable grammatical functions characterise arguments, while non-subcategorisable functions characterise adjuncts (modifiers).

Using this information we construct what we refer to as an "annotation matrix" for each of the rule LHS categories in the Penn-II treebank grammar. The x-axis of the matrix is given by the tripartition into left context, head and right context. The y-axis is defined by the distinction between subcategorisable and non-subcategorisable grammatical functions.

Consider a much simplified example: for rules (local trees) expanding English np's the rightmost nominal (n, nn, nns etc.) on the RHS is (usually) the head. Heads are annotated $\uparrow{=}\downarrow$. Any det or quant constituent in the left context is annotated $\uparrow$ spec $=\downarrow$. Any adjp in the left context is annotated $\downarrow\in\uparrow$ adjn. Any nominal in the left context (in noun noun sequences) is annotated as a modifier $\downarrow\in\uparrow$ adjn. Any pp in the right context is annotated as $\downarrow\in\uparrow$ adjn. Any relcl in the right context as $\downarrow\in\uparrow$ relmod, any nominal (phrase - usually separated by commas following the head) as an apposition $\downarrow\in\uparrow$ app and so forth. Information such as this is used to populate the np annotation matrix, partially represented in **Table 1**.

In order to minimise mistakes, the annotation matrices are very conservative: subcategorisable grammatical functions are only assigned if there is no doubt (e.g. an np following a preposition in a pp is assigned $\uparrow$ obj $=\downarrow$; a vp following a v in a vp constituent is assigned $\uparrow$ xcomp $=\downarrow$, $\uparrow$ subj $=\uparrow$ xcomp : subj and so forth). If, for any constituent, the argument - modifier status is in doubt, we annotate the constituent as an adjunct: $\downarrow\in\uparrow$ adjn.

Treebanks have an interesting property: for each cate-

np	left context	head	right context
subcat functions	det, quant : ↑ spec =↓	n, nn, nns : ↑=↓	...
non-subcat functions	adjp : ↓∈↑ adjn n, nn, nns : ↓∈↑ adjn ...		relcl : ↓∈↑ relmod pp : ↓∈↑ adjn n, nn, nns : ↓∈↑ app

Table 1: Simplified, partial annotation matrix for np rules

gory, there is a small number of very frequently occurring rules expanding that category, followed by a large number of less frequent rules many of which occur only once or twice in the treebank (Zipf's law).

For each particular category, the corresponding annotation matrix is constructed from the most frequent rules expanding that category. In order to guarantee similar coverage for the annotation matrices for the different rule LHS in the Penn-II treebank, we design each matrix according to an analysis of the most frequent CFG rules expanding that category, such that the token occurrences of those rules cover more than 80% of the token occurrences of all rules expanding that LHS category in the treebank. In order to do this we need to look at the following number of most frequent rule types for each category given in **Table 2**.

Although constructed based on the evidence of the most frequent rule types, the resulting annotation matrices do generalise to as yet unseen rule types in the following two ways:

- during the application of the annotation algorithm, annotation matrices annotate less frequent, unseen rules with constituents matching the left/right context and head specifications. The resulting annotation might be partial (i.e. some constituents in less frequent rule types may be left unannotated).

- in addition to monadic categories, the Penn-II treebank contains versions of these categories associated with functional annotations (-LOC, -TMP etc. indicating locative, temporal, etc. and other functional information). If we include functional annotations in the categories there are approx. 150 distinct LHS categories in the CFG extracted from the Penn-II treebank resource. Our annotation matrices were developed with the most frequent rule types expanding monadic categories only. During application of the annotation algorithm, the annotation matrix for any given monadic category C is also applied to all rules (local trees) expanding C-LOC, C-TMP etc., i.e. instances of the category carrying functional information.

In our work to date we have not yet covered "constituents" marked frag(ment) and x (unknown constituents) in the Penn-II treebank.

Finally, note that L/R context annotation principles are only applied if the local tree (rule RHS) does not contain any instance of a coordinating conjunction cc. Constructions involving coordinating conjunctions are treated separately in the second component of the annotation algorithm.

4.2. Coorordinating Conjunction Annotation Principles

Coordinating constructions come in two forms: like and unlike (UCPs) constituent coordinations. Due to the (often too) flat treebank analyses these present special problems. Because of this, an integrated treatment of coordinate structures with the other annotation principles would have been too complex and messy. For this reason we decided to treat coordinate structures in a separate module. Here we only have space to talk about like constituent coordinations.

The annotation algorithm first attempts to establish the head of a coordinate structure (usually the rightmost coordination) and annotates it accordingly. It then uses a variety of heuristics to find and annotate the various coordinated elements. One of the heuristics employed simply states that if both the immediate left and the immediate right constituents next to the coordination have the same category, then find all such categories in the left context of the rule and annotate these together with the immediate left and right constituents of the coordination as individual elements ↓∈↑ coord in the f-structure set representation of the coordination.

4.3. Catch-All Annotation Principles

The final component of the algorithm utilises functional information provided in the Penn-II treebank annotations. Any constituent, no matter what category, left unannotated by the previous two annotation algorithm components, that carries a Penn-II functional annotation other than SBJ and PRD, is annotated as an adjunct ↓∈↑ adjn.

5. Results and Evaluation

The annotation algorithm is implemented in terms of a Java program. Annotation of the complete WSJ section of the Penn-II treebank takes less than 30 minutes on a Pentium IV PC. Once annotated, for each tree we collect the feature structure annotations and feed them into a simple constraint solver implemented in Prolog.

Our constraint solver can handle equality constraints, disjunction and simple set valued feature constraints. Currently, however, our annotations do not involve disjunctive constraints. This means that for each tree in the treebank we either get a single f-structure, or, in the case of partially annotated trees, a number of unconnected f-structure fragments, or, in case of feature structure clashes, no f-structure.

As pointed out above, in our work to date we have not developed an annotation matrix for frag(mentary) constituents. Furthermore, as it stands, the algorithm completely ignores "movement" (or dislocation and control)

ADJP	ADVP	CONJP	FRAG	LST	NAC	NP	NX	PP	PRN	PRT	QP	RRC
25	3	3	184	4	6	64	14	2	35	2	11	12
S	SBAR	SBARQ	SINV	SQ	UCP	VP	WHADJP	WHADVP	WHNP	WHPP	X	
11	3	20	16	68	78	146	2	2	2	1	37	

Table 2: # of most frequent rule types analysed to construct annotation matrices

phenomena marked in the Penn-II annotations in terms of coindexation (of traces). This means that the f-structures generated in our work to date miss some reentrancies a more fine-grained analysis would show.

Furthermore, because of the limited capabilities of our constraint solver, in our current work we cannot use functional uncertainty constraints (regular expression based constraints over paths in f-structure) to localise unbounded dependencies to model "movement" phenomena. Also, again because of limitations of our constraint solver, we cannot express subsumption constraints in our annotations to, e.g., distribute subjects into coordinate `vp` structures.

To give an illustration of our method, we give the first sentence of the Penn-II treebank and the f-structure generated as an example in **Figure 3**.

Currently we get the following general results with our automatic annotation algorithm summarised in **Table 3**:

# f-structure (fragments)	# sentences	percentage
0	2701	5.576
1	38188	78.836
2	4954	10.227
3	1616	3.336
4	616	1.271
5	197	0.407
6	111	0.229
7	34	0.070
8	12	0.024
9	6	0.012
10	4	0.008
11	1	0.002

Table 3: Automatic annotation results

The Penn-II treebank contains 49167 trees. The results reported in **Table 3** ignore 727 trees containing `frag`(ment) and `x` (unknown) constituents as we did not provide any annotation for them in our work to date. At this early stage of our work, 38188 of the trees are associated with a complete f-structure. For 2701 trees no f-structure is produced (due to feature clashes). 4954 are associated with 2 f-structure fragments, 1616 with 3 fragments and so forth.

5.1. Evaluation

In order to evaluate the results of our automatic annotation we distinguish between "qualitative" and "quantitative" evaluation. Qualitative evaluation involves a "gold-standard", quantitative evaluation doesn't.

5.1.1. Qualitative Evaluation

Currently, we evaluate the output generated by our automatic annotation qualitatively by manually inspecting the f-structures generated. In order to automate the process we are currently working on a set of 100 randomly selected sentences from the Penn-II treebank to manually construct gold-standard annotated trees (and hence f-structures). These can then be processed in a number of ways:

- manually annotated gold-standard trees can be compared with the automatically annotated trees using the labelled bracketing precision and recall measures from `evalb`, a standard software package to evaluate PCFG parses. This presupposes that we treat annotated tree nodes as atoms (i.e. a complex string such as `np:`$\uparrow$ `obj` $=\downarrow$ is treated as an atomic label) and that in cases where nodes receive more than one f-structure annotation the order of these is the same in both the gold-standard and the automatically annotated version.

- gold-standard and automatically generated f-structures can be translated into a flat set of functional descriptions (`pred(A,see), subj(A,B), pred(B,John), obj(A,C), pred(C,Mary)`) and precision and recall can be computed for those.

- f-structures can be transformed (or unfolded) into trees by sorting attributes alphabetically at each level of embedding and by coding reentrancies as indices. After this transformation, gold-standard and automatically generated f-structures can be compared using `evalb`. This presupposes that both the gold-standard and the automatically generated f-structure have identical "terminal" yield.

5.1.2. Quantitative Evaluation

For purely quantitative evaluation (that is evaluation that doesn't necessarily assess the quality of the generated resources) we currently employ two related measures. These measures give an indication how partial our automatic annotation is at the current stage of the project. The first measure is the percentage of RHS constituents in grammar rules that receive an annotation. The table lists the annotation percentage for RHS elements of some of the Penn-II LHS categories. Because of the functional annotations provided in Penn-II the complete list of LHS categories would contain approx. 150 entries. Note that the percentages listed below ignore punctuation markers (which are not annotated):

```
Pierre Vinken, 61 years old, will join the board as a nonexecutive
director Nov. 29.


( S ( NP-SBJ ( NP ( NNP Pierre ) ( NNP Vinken ) ) ( , , ) ADJP ( NP (
 CD 61 ) ( NNS years ) ) ( JJ old ) ) ( , , ) ) ( VP ( MD will ) ( VP
 ( VB join ) ( NP ( DT the ) ( NN board ) ) ( PP-CLR ( IN as ) ( NP (
 DT a ) ( JJ nonexecutive ) ( NN director ) ) ) ( NP-TMP ( NNP Nov. )
 CD 29 ) ) ) ) ( . . ) )


subj : headmod : 1 : num : sing
                     pers : 3
                     pred : Pierre
       num : sing
       pers : 3
       pred : Vinken
       adjunct : 2 : adjunct : 3 : adjunct : 4 : pred : 61
                                   pers : 3
                                   pred : years
                                   num : pl
                     pred : old
xcomp : subj : headmod : 1 : num : sing
                             pers : 3
                             pred : Pierre
               num : sing
               pers : 3
               pred : Vinken
               adjunct : 2 : adjunct : 3 : adjunct : 4 : pred : 61
                                           pers : 3
                                           pred : years
                                           num : pl
                             pred : old
        obj : spec : det : pred : the
              num : sing
              pers : 3
              pred : board
        obl : obj : spec : det : pred : a
                    adjunct : 5 : pred : nonexecutive
                    pred : director
                    num : sing
                    pers : 3
              pred : as
        pred : join
        adjunct : 6 : pred : Nov.
                      num : sing
                      pers : 3
                      adjunct : 7 : pred : 29
pred : will
modal : +
```

Figure 3: F-structure generated for the first sentence in Penn-II

LHS	# RHS elements	# RHS annotated	% annotated
ADJP	1653	1468	88.80
ADJP-ADV	21	21	100.00
ADJP-CLR	27	24	88.88
ADV	607	532	87.64
NP	30793	29145	94.64
PP	1090	905	83.02
S	14912	13144	88.14
SBAR	423	331	78.25
SBARQ	270	212	78.51
SQ	657	601	91.47
VP	40990	35693	87.07

The second, related measure gives the average number of f-structure fragments generated for each treebank tree (the more partial our annotation the more unconnected f-structure fragments are generated for a sentence). For 45739 sentences, the average number of fragments per sentences is currently: 1.26 (note again that the number excludes sentences containing `frag` and `x` constituents).

6. Conclusion and Further Work

In this paper we have presented an automatic f-structure annotation algorithm and applied it to annotate the Penn-II treebank resource with f-structure information. The resulting representations are proto-f-structures showing basic predicate-argument-modifier structure. Currently, 38,188 sentences (78.8% of the 48,440 trees without `frag` and `x` constituents) receive a complete f-structure; 4954 sentences are associated with two f-structure fragments, 1,616 with three fragments. 2,701 sentences are not associated with an f-structure.

In future work we plan to extend and refine our automatic annotation algorithm in a number of ways:

- We are working on reducing the the amount of f-structure fragmentation by providing more complete annotation principles.

- Currently the `pred` values (i.e. the predicates) in the f-structures generated are surface (i.e. inflected) rather than root forms. We are planning to use the output of a two-level morphology to annotate the Penn-II strings with root forms which can then be picked up by our lexical macros and used as `pred` values in the automatic annotations.

- Currently our annotation algorithm ignores the Penn-II encoding of "moved" constituents in topicalisation, wh-constructions, control constructions and the like. These (often non-local) dependencies are marked in the Penn-II tree annotations in terms of indices. In future work we intend to make our annotation algorithm sensitive to such information. There are two (possibly complementary) ways of achieving this: The first is to make the annotation algorithm sensitive to the index scheme provided by the Penn-II annotations either during application of the algorithm or in terms of undoing "movement" in a treebank preprocessing step. The latter route is explored in recent work by (Liakata and Pulman, 2002). The second possibility is to use the LFG machinery of functional uncertainty equations to effectively localise unbounded dependency relations in a functional annotation at a particular node. Functional uncertainty equations allow the statement of regular expression based paths in f-structure. Currently we cannot resolve such paths with our constraint solver.

- We are currently experimenting with probabilistic grammars extracted from the automatically annotated version of the Penn-II treebank. We will be reporting on the results of these experiments elsewhere (Cahill et al, 2002).

- We are planning to exploit the f-structure/QLF/UDRS correspondences established by (van Genabith and Crouch, 1996; van Genabith and Crouch, 1997) to generate semantically annotated versions of the Penn-II treebank.

Acknowledgements

This research was part funded by Enterprise Ireland Basic Research grant SC/2001/186.

7. References

R. Bod and R. Kaplan 1998. A probabilistic corpus-driven model for lexical-functional grammar. In: *Proceedings of Coling/ACL'98*. 145–151.

A. Cahill, M. McCarthy, J. van Genabith and A. Way 2002. Parsing with a PCFG Derived from Penn-II with an Automatic F-Structure Annotation Procedure. In: *The sixth International Conference on Lexical-Functional Grammar*, Athens, Greece, 3 July - 5 July 2002 to appear (2002)

M. Collins 1999. *Head-driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania, Philadelphia.

J. Bresnan 2001. *Lexical-Functional Syntax*. Blackwell, Oxford.

A. Frank. 2000. Automatic F-Structure Annotation of Treebank Trees. In: (eds.) M. Butt and T. H. King, *The fifth International Conference on Lexical-Functional Grammar*, The University of California at Berkeley, 19 July - 20 July 2000, CSLI Publications, Stanford, CA.

A. Frank, L. Sadler, J. van Genabith and A. Way 2002. From Treebank Resources tp LFG F-Structures. In: (ed.) Anne Abeille, *Treebanks: Building and Using Syntactically Annotated Corpora*, Kluwer Academic Publishers, Dordrecht/Boston/London, to appear (2002)

M. Kay 1999. Chart Translation. In *Proceedings of the Machine Translation Summit VII*. "MT in the great Translation Era". 9–14.

R. Kaplan and J. Bresnan 1982. Lexical-functional grammar: a formal system for grammatical representation. In Bresnan, J., editor 1982, *The Mental Representation of Grammatical Relations*. MIT Press, Cambridge Mass. 173–281.

M. Liakata and S. Pulman 2002. *From trees to predicate-argument structures*. Unpublished working paper. Centre for Linguistics and Philology, Oxford University.

M. Marcus, G. Kim, M. A. Marcinkiewicz, R. MacIntyre, M. Ferguson, K. Katz and B. Schasberger 1994. The Penn Treebank: Annotating Predicate Argument Structure. In: *Proceedings of the ARPA Human Language Technology Workshop*.

L. Sadler, J. van Genabith and A. Way. 2000. Automatic F-Structure Annotation from the AP Treebank. In: (eds) M. Butt and T. H. King, *The fifth International Conference on Lexical-Functional Grammar*, The University of California at Berkeley, 19 July - 20 July 2000, CSLI Publications, Stanford, CA.

J. van Genabith and D. Crouch 1996. Direct and Underspecified Interpretations of LFG f-Structures. In: *COLING 96*, Copenhagen, Denmark, Proceedings of the Conference. 262–267.

J. van Genabith and D. Crouch 1997. On Interpreting f-Structures as UDRSs. In: *ACL-EACL-97*, Madrid, Spain, Proceedings of the Conference. 402–409.

Incremental Specialization of an HPSG-Based Annotation Scheme

Kiril Simov, Milen Kouylekov, Alexander Simov

BulTreeBank Project
http://www.BulTreeBank.org
Linguistic Modelling Laboratory, Bulgarian Academy of Sciences
Acad. G. Bonchev St. 25A, 1113 Sofia, Bulgaria
kivs@bgcict.acad.bg, mkouylekov@dir.bg, adis_78@dir.bg

Abstract

The linguistic knowledge represented in contemporary language resource annotations becomes very complex. Its acquiring and management requires an enormous amount of human work. In order to minimize such a human effort we need rigorous methods for representation of such knowledge, methods for supporting the annotation process, methods for exploiting all results from the annotation process, even those that usually disappear after the annotation has been completed. In this paper we present a formal set-up for annotation within HPSG linguistic theory. We present also an algorithm for annotation scheme specialization based on the negative information from the annotation process. The negative information includes the analyses, rejected by the annotator.

1. Introduction

In our project (Simov et. al., 2001a), (Simov et al., 2002) we aim at the creation of syntactically annotated corpus (treebank) based on the HPSG linguistic theory (Head-driven Phrase Structure Grammar — (Pollard and Sag, 1987) and (Pollard and Sag, 1994)). Hence, the elements of the treebank are not trees, but feature graphs. The annotation scheme for the construction of the treebank is based on the appropriate language-specific version of the HPSG sort hierarchy. On one hand, such an annotation scheme is very detailed and flexible with respect to the linguistic knowledge, encoded in it. But, on the other hand, because of the massive overgeneration, it is not considered to be annotator-friendly. Thus, the main problem is: how to keep the consistency of the annotation scheme and at the same time to minimize the human work during the annotation. In our annotation architecture we envisage two sources of linguistic knowledge in order to reduce the possible analyses of the annotated sentences:

- Reliable partial grammars.
- HPSG-based grammar: universal principles, language specific principles and a lexicon.

The actual annotation process includes the following steps:

- Partial parsing step:

  This step comprises several additional steps: (1) Sentence extraction from the text archive; (2) Morphosyntactic tagging; (3) Part-of-speech disambiguation; (4) Partial parsing;

  The result is considered a 100 % accurate partial parsed sentence.

- HPSG step:

  The result from the previous step is encoded into an HPSG compatible representation with respect to the sort hierarchy. It is sent to an HPSG grammar tool, which takes the partial sentence analysis as input and evaluates all the attachment possibilities for it. The output is encoded as feature graphs.

- Annotation step:

  The feature graphs from the previous step are further processed as follows : (1) their intersection is calculated; (2) on the base of the differences, a set of constraints over the intersection is calculated as well; (3) during the actual annotation step, the annotator tries to extend the intersection to full analysis, adding new information to it. The constraints determine the possible extensions and also propagate the information, added by the annotator, in order to minimize the incoming choices.

This architecture is being currently implemented by establishing an interface between two systems: CLaRK system for XML based corpora development (Simov et. al., 2001b) and TRALE system for HPSG grammar development (TRALE is a descendant of (Götz and Meurers, 1997)). The project will result in an HPSG corpus based on feature graphs and reliable grammars. One of the intended applications of these language resources consists of their exploration for improving the accuracy of the implemented HPSG grammar.

The work, reported in this paper, is a step towards establishing an incremental mechanism, which uses already annotated sentences for further specializing of the HPSG grammar and for reducing the number of the possible HPSG analyses. In fact, we consider the rejected analyses as negative information about the language and therefore the grammar has to be appropriately tuned in order to rule out such analyses.

The structure of the paper is as follows: in the next section we define formally what a corpus is with respect to a grammar formalism and apply this definition to the definition of an HPSG corpus. In Sect. 3. we present a logical formalism for HPSG, define a normal form for grammars in the logical formalism and on the basis of this normal form we define feature graphs that constitute a good representation for both — HPSG grammars and HPSG corpora. Sect. 4. presents the algorithm for specialization of an

HPSG grammar on the basis of accepted and rejected by the annotator analyses produced by the grammar. Then Sect. 5. demonstrates an example of such specialization. The last section outlines the conclusions and outlook.

2. HPSG Corpus

In our work we accept that the corpus is complete with respect to the analyses of the sentences in it. This means that each sentence is presented with all its acceptable syntactic structures. Thus a good grammar will not overgenerate, i.e. it will not assign more analyses to the sentences than the analyses, which already exist in the corpus. Before we define what an HPSG corpus is like, let us start with a definition of a grammar-formalism-based corpus in general. Such an ideal corpus has to ensure the above assumption.

Definition 1 (Grammar Formalism Corpus) *A corpus C in a given grammatical formalism G is a sequence of analyzed sentences where each analyzed sentence is a member of the set of structures defined as a* **strong generative capacity** *(*$\mathtt{SGC}$*) of a grammar Γ in this grammatical formalism:*

$\forall S.S \in C \rightarrow S \in \mathtt{SGC}(\Gamma)$,

where Γ is a grammar in the formalism G, and if $\sigma(S)$ is the phonological string of S and $\Gamma(\sigma(S))$ is the set of all analyses assigned by the grammar Γ to the phonological string $\sigma(S)$, then

$\forall S'.S' \in \Gamma(\sigma(S)) \rightarrow S' \in C$.

The grammar Γ is unknown, but implicitly represented in the corpus C. We could state that if such a grammar does not exist, then we consider the corpus inconsistent or uncomplete.

In order to define a corpus in HPSG with respect to this definition, we have to define a representation of HPSG analysis over the sentences. This analysis must correspond to a definition of strong generative capacity in HPSG. Fortunately, there exist such definitions - (King 1999) and (Pollard 1999). We adopt them for our purposes. Thus in our work we choose:

- A logical formalism for HPSG — King's Logic (SRL) (King 1989);
- A definition of strong generative capacity in HPSG as a set of feature structures closely related to the special interpretation in SRL (exhaustive models) along the lines of (King 1999) and (Pollard 1999).
- A definition of corpus in HPSG as a sequence of sentences that are members of $\mathtt{SGC}(\Gamma)$ for some grammar Γ in SRL.

It is well-known that an HPSG grammar in SRL formally comprises two parts: a signature and a theory. The signature defines the ontology of the linguistic objects in the language and the theory constraints the shape of the linguistic objects. Usually the descriptions in the theory part are presented as implications. In order to demonstrate in a better way the connection between the HPSG grammar in SRL and the HPSG corpus, we offer a common representation of the grammar and the corpus.

We define a normal form for HPSG grammars which ideologically is very close to the feature structures defining the strong generative capacity in HPSG as it has proposed in the work of (King 1999) and (Pollard 1999). We define both the corpus and the grammar in terms of clauses (considered as graphs) in a special kind of matrices in SRL. The construction of new sentence analyses can be done using the inference mechanisms of SRL. Another possibility is such a procedure to be defined directly using the representations in the normal form. In order to distinguish the elements in our normal form from the numerous of kinds of feature structures we call the elements in the normal form *feature graphs*. One important characteristic about our feature graphs is that they are viewed as descriptions in SRL, i.e. as syntactic entities.

In other works (Simov, 2001) and (Simov, 2002) we showed how from a corpus, consisting of feature graphs, a corpus grammar could be extracted along the lines of Rens Bod's ideas on Data-Oriented Parsing Model (Bod, 1998). Also, in (Simov, 2002) we showed how one could use the positive information in the corpus in order to refine an existing HPSG grammar. In this paper we discuss and illustrate the usage of the negative information compiled as a by-product during the annotation of the corpus.

3. Logical Formalism for HPSG

In this section we present a logical formalism for HPSG. Then a normal form (exclusive matrices) for a finite theory in this formalism is defined and then we show how it can be represented as a set of feature graphs. These graphs are considered a representation of grammars and corpora in HPSG.

3.1. King's Logic — SRL

This section presents the basic notions of Speciate Reentrancy Logic (SRL) (King 1989).

$\Sigma = \langle \mathcal{S}, \mathcal{F}, \mathcal{A} \rangle$ is a finite **SRL signature** iff $\mathcal{S}$ is a finite set of *species*, $\mathcal{F}$ is a set of *features*, and $\mathcal{A} : \mathcal{S} \times \mathcal{F} \rightarrow Pow(\mathcal{S})$ is an *appropriateness function*. $\mathcal{I} = \langle \mathcal{U}_\mathcal{I}, \mathcal{S}_\mathcal{I}, \mathcal{F}_\mathcal{I} \rangle$ is a **SRL interpretation** of the signature Σ (or Σ-interpretation) iff

$\mathcal{U}_\mathcal{I}$ is a non-empty set of objects,
$\mathcal{S}_\mathcal{I}$ is a total function from $\mathcal{U}_\mathcal{I}$ to $\mathcal{S}$,
  called **species assignment function**,
$\mathcal{F}_\mathcal{I}$ is a total function from $\mathcal{F}$ to the set of partial function from $\mathcal{U}_\mathcal{I}$ to $\mathcal{U}_\mathcal{I}$ such that
  for each $\phi \in \mathcal{F}$ and each $\upsilon \in \mathcal{U}_\mathcal{I}$, if $\mathcal{F}_\mathcal{I}(\phi)(\upsilon)\downarrow$[1]
    then $\mathcal{S}_\mathcal{I}(\mathcal{F}_\mathcal{I}(\phi)(\upsilon)) \in \mathcal{A}(\mathcal{S}_\mathcal{I}(\upsilon), \phi)$, and
  for each $\phi \in \mathcal{F}$ and each $\upsilon \in \mathcal{U}_\mathcal{I}$, if $\mathcal{A}(\mathcal{S}_\mathcal{I}(\upsilon), \phi)$
    is not empty then $\mathcal{F}_\mathcal{I}(\phi)(\upsilon)\downarrow$,
$\mathcal{F}_\mathcal{I}$ is called **feature interpretation function**.

τ is a term iff τ is a member of the smallest set $\mathcal{TM}$ such that (1) : $\in \mathcal{TM}$, and (2) for each $\phi \in \mathcal{F}$ and each $\tau \in \mathcal{TM}$, $\tau\phi \in \mathcal{TM}$. For each Σ-interpretation $\mathcal{I}$, $\mathcal{P}_\mathcal{I}$ is a **term interpretation function** over $\mathcal{I}$ iff (1) $\mathcal{P}_\mathcal{I}(:)$ is the identity function from $\mathcal{U}_\mathcal{I}$ to $\mathcal{U}_\mathcal{I}$, and (2) for each $\phi \in \mathcal{F}$ and each $\tau \in \mathcal{TM}$, $\mathcal{P}_\mathcal{I}(\tau\phi)$ is the composition of the partial functions $\mathcal{P}_\mathcal{I}(\tau)$ and $\mathcal{F}_\mathcal{I}(\phi)$ if they are defined.

[1] $f(o)\downarrow$ means the function f is defined for the argument o.

δ is a **description** iff δ is a member of the smallest set $\mathcal{D}$ such that (1) for each $\sigma \in \mathcal{S}$ and for each $\tau \in \mathcal{TM}$, $\tau \sim \sigma \in \mathcal{D}$, (2) for each $\tau_1 \in \mathcal{TM}$ and $\tau_2 \in \mathcal{TM}$, $\tau_1 \approx \tau_2 \in \mathcal{D}$ and $\tau_1 \not\approx \tau_2 \in \mathcal{D}$, (3) for each $\delta \in \mathcal{D}$, $\neg\delta \in \mathcal{D}$, (4) for each $\delta_1 \in \mathcal{D}$ and $\delta_2 \in \mathcal{D}$, $[\delta_1 \wedge \delta_2] \in \mathcal{D}$, $[\delta_1 \vee \delta_2] \in \mathcal{D}$, and $[\delta_1 \rightarrow \delta_2] \in \mathcal{D}$. **Literals** are descriptions of the form $\tau \sim \sigma$, $\tau_1 \approx \tau_2$, $\tau_1 \not\approx \tau_2$ or their negation. For each Σ-interpretation $\mathcal{I}$, $\mathcal{D}_\mathcal{I}$ is a **description denotation function** over $\mathcal{I}$ iff $\mathcal{D}_\mathcal{I}$ is a total function from $\mathcal{D}$ to the powerset of $\mathcal{U}_\mathcal{I}$, such that

$\mathcal{D}_\mathcal{I}(\tau \sim \sigma) = \{\upsilon \in \mathcal{U}_\mathcal{I} \mid \mathcal{P}_\mathcal{I}(\tau)(\upsilon)\downarrow, \mathcal{S}_\mathcal{I}(\mathcal{P}_\mathcal{I}(\tau)(\upsilon)) = \sigma\}$,
$\mathcal{D}_\mathcal{I}(\tau_1 \approx \tau_2) = \{\upsilon \in \mathcal{U}_\mathcal{I} \mid \mathcal{P}_\mathcal{I}(\tau_1)(\upsilon)\downarrow, \mathcal{P}_\mathcal{I}(\tau_2)(\upsilon)\downarrow,$ and $\mathcal{P}_\mathcal{I}(\tau_1)(\upsilon) = \mathcal{P}_\mathcal{I}(\tau_2)(\upsilon)\}$,
$\mathcal{D}_\mathcal{I}(\tau_1 \not\approx \tau_2) = \{\upsilon \in \mathcal{U}_\mathcal{I} \mid \mathcal{P}_\mathcal{I}(\tau_1)(\upsilon)\downarrow, \mathcal{P}_\mathcal{I}(\tau_2)(\upsilon)\downarrow,$ and $\mathcal{P}_\mathcal{I}(\tau_1)(\upsilon) \neq \mathcal{P}_\mathcal{I}(\tau_2)(\upsilon)\}$,
$\mathcal{D}_\mathcal{I}(\neg\delta) = \mathcal{U}_\mathcal{I} \setminus \mathcal{D}_\mathcal{I}(\delta)$,
$\mathcal{D}_\mathcal{I}([\delta_1 \wedge \delta_2]) = \mathcal{D}_\mathcal{I}(\delta_1) \cap \mathcal{D}_\mathcal{I}(\delta_2)$,
$\mathcal{D}_\mathcal{I}([\delta_1 \vee \delta_2]) = \mathcal{D}_\mathcal{I}(\delta_1) \cup \mathcal{D}_\mathcal{I}(\delta_2)$, and
$\mathcal{D}_\mathcal{I}([\delta_1 \rightarrow \delta_2]) = (\mathcal{U}_\mathcal{I} \setminus \mathcal{D}_\mathcal{I}(\delta_1)) \cup \mathcal{D}_\mathcal{I}(\delta_2)$.

Each subset $\theta \subseteq \mathcal{D}$ is an **SRL theory**. For each Σ-interpretation $\mathcal{I}$, $\mathcal{T}_\mathcal{I}$ is a **theory denotation function** over $\mathcal{I}$ iff $\mathcal{T}_\mathcal{I}$ is a total function from the powerset of $\mathcal{D}$ to the powerset of $\mathcal{U}_\mathcal{I}$ such that for each $\theta \subseteq \mathcal{D}$, $\mathcal{T}_\mathcal{I}(\theta) = \cap\{\mathcal{D}_\mathcal{I}(\delta) | \delta \in \theta\}$. $\mathcal{T}_\mathcal{I}(\emptyset) = \mathcal{U}_\mathcal{I}$. A theory θ is **satisfiable** iff for some interpretation $\mathcal{I}$, $\mathcal{T}_\mathcal{I}(\theta) \neq \emptyset$. A theory θ is **modelable** iff for some interpretation $\mathcal{I}$, $\mathcal{T}_\mathcal{I}(\theta) = \mathcal{U}_\mathcal{I}$, $\mathcal{I}$ is called a **model** of θ. The interpretation $\mathcal{I}$ **exhaustively models** θ iff

$\mathcal{I}$ is a model of θ, and
for each $\theta' \subseteq \mathcal{D}$,
if for some model $\mathcal{I}'$ of θ,
$\mathcal{T}_{\mathcal{I}'}(\theta') \neq \emptyset$,
then $\mathcal{T}_\mathcal{I}(\theta') \neq \emptyset$.

An HPSG grammar $\Gamma = \langle\Sigma, \theta\rangle$ in SRL consists of: (1) a signature Σ which gives the ontology of entities that exist in the universe and the appropriateness conditions on them, and (2) a theory θ which gives the restrictions upon these entities.

3.2. Exclusive Matrices

Following (King and Simov, 1998) in this section we define a normal form for finite theories in SRL — called **exclusive matrix**. This normal form possesses some desirable properties for representation of grammars and corpora in HPSG.

First, we define some technical notions. A **clause** is a finite set of literals interpreted *conjunctively*. A **matrix** is a finite set of clauses interpreted *disjunctively*.

A matrix μ is an **exclusive matrix** iff for each clause $\alpha \in \mu$,

(E0) if $\lambda \in \alpha$ then λ is a positive literal,
(E1) $:\approx: \in \alpha$,
(E2) if $\tau_1 \approx \tau_2 \in \alpha$ then $\tau_2 \approx \tau_1 \in \alpha$,
(E3) if $\tau_1 \approx \tau_2 \in \alpha$ and $\tau_2 \approx \tau_3 \in \alpha$ then $\tau_1 \approx \tau_3 \in \alpha$,
(E4) if $\tau\phi \approx \tau\phi \in \alpha$ then $\tau \approx \tau \in \alpha$,
(E5) if $\tau_1 \approx \tau_2 \in \alpha$, $\tau_1\phi \approx \tau_1\phi \in \alpha$ and $\tau_2\phi \approx \tau_2\phi \in \alpha$ then $\tau_1\phi \approx \tau_2\phi \in \alpha$,
(E6) if $\tau \approx \tau \in \alpha$ then for some $\sigma \in \mathcal{S}$, $\tau \sim \sigma \in \alpha$,
(E7) if for some $\sigma \in \mathcal{S}$, $\tau \sim \sigma \in \alpha$ then $\tau \approx \tau \in \alpha$,
(E8) if $\tau_1 \approx \tau_2 \in \alpha$, $\tau_1 \sim \sigma_1 \in \alpha$ and $\tau_2 \sim \sigma_2 \in \alpha$ then $\sigma_1 = \sigma_2$,
(E9) if $\tau \sim \sigma_1 \in \alpha$ and $\tau\phi \sim \sigma_2 \in \alpha$ then $\sigma_2 \in \mathcal{A}(\sigma_1, \phi)$,
(E10) if $\tau \sim \sigma \in \alpha$, $\tau\phi \in \mathit{Term}(\mu)$ and $\mathcal{A}(\sigma, \phi) \neq \emptyset$ then $\tau\phi \approx \tau\phi \in \alpha$,
(E11) if $\tau_1 \not\approx \tau_2 \in \alpha$ then $\tau_1 \approx \tau_1 \in \alpha$ and $\tau_2 \approx \tau_2 \in \alpha$,
(E12) if $\tau_1 \approx \tau_1 \in \alpha$ and $\tau_2 \approx \tau_2 \in \alpha$ then $\tau_1 \approx \tau_2 \in \alpha$ or $\tau_1 \not\approx \tau_2 \in \alpha$, and
(E13) $\tau_1 \approx \tau_2 \notin \alpha$ or $\tau_1 \not\approx \tau_2 \notin \alpha$,

where $\{\sigma, \sigma_1, \sigma_2\} \subseteq \mathcal{S}$, $\phi \in \mathcal{F}$, and $\{\tau, \tau_1, \tau_2, \tau_3\} \subseteq \mathcal{TM}$, and $\mathtt{Term}$ is a function from the powerset of the sets of literals to the powerset of $\mathcal{TM}$ such that

$\mathtt{Term}(\alpha) = \{\tau \mid (\neg)\tau\phi \approx \tau' \in \alpha, \tau \in \mathcal{TM}, \phi \in \mathcal{F}^*\} \cup$
$\{\tau \mid (\neg)\tau' \approx \tau\phi \in \alpha, \tau \in \mathcal{TM}, \phi \in \mathcal{F}^*\} \cup$
$\{\tau \mid (\neg)\tau\phi \not\approx \tau' \in \alpha, \tau \in \mathcal{TM}, \phi \in \mathcal{F}^*\} \cup$
$\{\tau \mid (\neg)\tau' \not\approx \tau\phi \in \alpha, \tau \in \mathcal{TM}, \phi \in \mathcal{F}^*\} \cup$
$\{\tau \mid (\neg)\tau\phi \sim \sigma \in \alpha, \tau \in \mathcal{TM}, \phi \in \mathcal{F}^*\}$.

There are two important properties of an exclusive matrix $\mu = \{\alpha_1, \ldots, \alpha_n\}$: (1) each clause α in μ is satisfiable (for some interpretation $\mathcal{I}$, $\mathcal{T}_\mathcal{I}(\alpha) \neq \emptyset$), and (2) each two clauses α_1, α_2 in μ have disjoint denotations (for each interpretation $\mathcal{I}$, $\mathcal{T}_\mathcal{I}(\alpha_1) \cap \mathcal{T}_\mathcal{I}(\alpha_1) = \emptyset$). Also in (King and Simov, 1998) it is shown that each finite theory with respect to a finite signature can be converted into an exclusive matrix which is semantically equivalent to the theory. Relying on the definition of model (where each object in the domain is described by the theory) and the property that each two clauses in an exclusive matrix have disjoint denotation, one can easy prove the following proposition.

Proposition 2 *Let θ be a finite SRL theory with respect to a finite signature, μ be the corresponding exclusive matrix and $\mathcal{I} = \langle\mathcal{U}, \mathcal{S}, \mathcal{F}\rangle$ be a model of θ. For each object $\upsilon \in \mathcal{U}$ there exists a unique clause $\alpha \in \mu$ such that $\upsilon \in \mathcal{T}(\alpha)$.*

3.3. Feature Graphs

As it was mentioned above, an HPSG corpus will comprise a set of feature structures representing the HPSG analyses of the sentences. We interpret these feature structures as descriptions in SRL (clauses in an exclusive matrix).

Let $\Sigma = \langle\mathcal{S}, \mathcal{F}, \mathcal{A}\rangle$ be a finite signature. A directed, connected and rooted graph $\mathcal{G} = \langle\mathcal{N}, \mathcal{V}, \rho, \mathcal{S}\rangle$ such that

$\mathcal{N}$ is a set of **nodes**,
$\mathcal{V}: \mathcal{N} \times \mathcal{F} \rightarrow \mathcal{N}$ is a partial **arc function**,
ρ is a **root node**,
$\mathcal{S}: \mathcal{N} \rightarrow \mathcal{S}$ is a total **species assignment function**, such that
for each $\nu_1, \nu_2 \in \mathcal{N}$ and each $\phi \in \mathcal{F}$
if $\mathcal{V}\langle\nu_1, \phi\rangle\downarrow$ and $\mathcal{V}\langle\nu_1, \phi\rangle = \nu_2$,
then $\mathcal{S}\langle\nu_2\rangle \in \mathcal{A}\langle\mathcal{S}\langle\nu_1\rangle, \phi\rangle$,

is a **feature graph** wrt Σ.

A feature graph $\mathcal{G} = \langle\mathcal{N}, \mathcal{V}, \rho, \mathcal{S}\rangle$ such that for each node $\nu \in \mathcal{N}$ and each feature $\phi \in \mathcal{F}$ if $\mathcal{A}\langle\mathcal{S}\langle\nu\rangle, \phi\rangle\downarrow$ then $\mathcal{V}\langle\nu, \phi\rangle\downarrow$ is called a **complete feature graph** (or complete graph).

According to our definition feature graphs are a kind of feature structures which are treated syntactically rather than semantically. We use complete feature graphs for representing the analyses of the sentences in the corpus.

We say that the feature graph $\mathcal{G}$ is **finite** if and only if the set of nodes is finite.

For each graph $\mathcal{G}=\langle\mathcal{N},\mathcal{V},\rho,\mathcal{S}\rangle$ and node ν in $\mathcal{N}$ with $\mathcal{G}\,|_\nu=\langle\mathcal{N}_\nu,\mathcal{V}\,|_{\mathcal{N}_\nu},\rho_\nu,\mathcal{S}\,|_{\mathcal{N}_\nu}\rangle$ we denote the **subgraph** of $\mathcal{G}$ starting on node ν.

Let $\mathcal{G}_1=\langle\mathcal{N}_1,\mathcal{V}_1,\rho_1,\mathcal{S}_1\rangle$ and $\mathcal{G}_2=\langle\mathcal{N}_2,\mathcal{V}_2,\rho_2,\mathcal{S}_2\rangle$ be two graphs. We say that graph $\mathcal{G}_1$ **subsumes** graph $\mathcal{G}_2$ ($\mathcal{G}_2\sqsubseteq\mathcal{G}_1$) iff there is an *isomorphism* $\gamma:\mathcal{N}_1\rightarrow\mathcal{N}_2'$, $\mathcal{N}_2'\subseteq\mathcal{N}_2$, such that

$\gamma(\rho_1)=\rho_2$,

for each $\nu,\nu'\in\mathcal{N}_1$ and each feature ϕ,

$\mathcal{V}_1\langle\nu,\phi\rangle=\nu'$ iff $\mathcal{V}_2\langle\gamma(\nu),\phi\rangle=\gamma(\nu')$, and

for each $\nu\in\mathcal{N}_1$, $\mathcal{S}_1\langle\nu\rangle=\mathcal{S}_2\langle\gamma(\nu)\rangle$.

The intuition behind the definition of subsumption by isomorphism is that each graph describes "exactly" a chunk in some SRL interpretation in such a way that every two distinct nodes are always mapped to distinct objects in the interpretation.

For each two graphs $\mathcal{G}_1$ and $\mathcal{G}_2$ if $\mathcal{G}_2\sqsubseteq\mathcal{G}_1$ and $\mathcal{G}_1\sqsubseteq\mathcal{G}_2$ we say that $\mathcal{G}_1$ and $\mathcal{G}_2$ are **equivalent.** For convenience, in the following text we consider each two equivalent graphs equal.

For a finite feature graph $\mathcal{G}=\langle\mathcal{N},\mathcal{V},\rho,\mathcal{S}\rangle$, we define a translation to a clause. Let

$Term(\mathcal{G})=\{:\}\cup\{\tau\mid\tau\doteq:\phi_1\ldots\phi_n,n\leq\|\mathcal{N}\|,\mathcal{V}\langle\rho,\tau\rangle\downarrow\}$[2] be a set of terms. We define a clause $\alpha_\mathcal{G}$:

$\alpha_\mathcal{G}=\{\tau\sim\sigma\mid\tau\in Term(\mathcal{G}),\mathcal{V}\langle\rho,\tau\rangle\downarrow,\mathcal{S}\langle\mathcal{V}\langle\rho,\tau\rangle\rangle=\sigma\}\cup$

$\{\tau_1\approx\tau_2\mid\tau_1\in Term(\mathcal{G}),\tau_2\in Term(\mathcal{G}),$

$\mathcal{V}\langle\rho,\tau_1\rangle\downarrow,\mathcal{V}\langle\rho,\tau_2\rangle\downarrow,$ and $\mathcal{V}\langle\rho,\tau_1\rangle=\mathcal{V}\langle\rho,\tau_2\rangle\}\cup$

$\{\tau_1\not\approx\tau_2\mid\tau_1\in Term(\mathcal{G}),\tau_2\in Term(\mathcal{G}),$

$\mathcal{V}\langle\rho,\tau_1\rangle\downarrow,\mathcal{V}\langle\rho,\tau_2\rangle\downarrow,$ and $\mathcal{V}\langle\rho,\tau_1\rangle\neq\mathcal{V}\langle\rho,\tau_2\rangle\}$.

We interpret a finite feature graph via the interpretation of the corresponding clauses

$\mathcal{R}_\mathcal{I}(\mathcal{G})=\mathcal{T}_\mathcal{I}(\alpha_\mathcal{G})$.

Let $\mathcal{G}$ be an infinite feature graph. Then we interpret it as the intersection of the interpretations of all finite feature graphs that subsume it:

$\mathcal{R}_\mathcal{I}(\mathcal{G})=\cap_{\mathcal{G}\sqsubseteq\mathcal{G}',\mathcal{G}'<\omega}\mathcal{R}_\mathcal{I}(\alpha_{\mathcal{G}'})$.

The clauses in an exclusive matrix μ can be represented as feature graphs. Let μ be an exclusive matrix and $\alpha\in\mu$, then

$\mathcal{G}_\alpha=\langle\mathcal{N}_\alpha,\mathcal{V}_\alpha,\rho_\alpha,\mathcal{S}_\alpha\rangle$ is a feature graph such that

$\mathcal{N}_\alpha=\{|\tau|_\alpha\mid\tau\approx\tau\in\alpha\}$ is a set of nodes,

$\mathcal{V}_\alpha:\mathcal{N}_\alpha\times\mathcal{F}\rightarrow\mathcal{N}_\alpha$ is a partial **arc function**, such that

$\mathcal{V}_\alpha\langle|\tau_1|_\alpha,\phi\rangle\downarrow$ and $\mathcal{V}_\alpha\langle|\tau_1|_\alpha,\phi\rangle=|\tau_2|_\alpha$ iff

$\tau_1\approx\tau_1\in\alpha,\tau_2\approx\tau_2\in\alpha,\phi\in\mathcal{F}$, and $\tau_1\phi\approx\tau_2\in\alpha$,

ρ_α is the root node $|:|_\alpha$, and

$\mathcal{S}_\alpha:\mathcal{N}_\alpha\rightarrow\mathcal{S}$ is a **species assignment function**, such that

$\mathcal{S}_\alpha\langle|\tau|_\alpha\rangle=\sigma$ iff $\tau\sim\sigma\in\alpha$.

Proposition 3 *Let μ be an exclusive matrix and $\alpha\in\mu$. Then the graph $\mathcal{G}_\alpha$ is semantically equivalent to α.*

3.4. Inference with Feature Graphs

In this paper we do not present a concrete inference mechanism exploiting feature graphs. As it was mentioned above, one can use the general inference mechanisms of SRL in order to construct sentence analyses. However, a much better solution is to employ an inference mechanism, which uses directly the graph representation of a theory. Such an inference mechanism can be defined along the lines of *Breadth-First Parallel Resolution* in (Carpener 1992) despite the difference in the treatment of the feature structure in (Carpener 1992) (Note that (Carpener 1992) treats feature structures as semantic entities, but we consider our feature graphs syntactic elements.). One has to keep in mind that finding models in SRL is undecidable (see (King, Simov and Aldag 1999)) and some restrictions in terms of time or memory will be necessary in order to use Breadth-First Parallel Resolution-like algorithm. A presentation of such an algorithm is beyond the scope of this paper.

[2] $\|X\|$ is the cardinality of the set X

3.5. Graph Representation of an SRL Theory

Each finite SRL theory can be represented as a set of feature graphs. In order to make this graph transformation of a theory completely independent from the SRL particulars, we also need to incorporate within the graphs the information from the signature that is not present in the theory yet. For each species the signature encodes the defined features as well as the species of their possible values. We explicate this information in the signature by constructing a special theory:

$$\theta_\Sigma=\{\bigvee_{\sigma\in\mathcal{S}}[\bigwedge_{\mathcal{A}(\sigma,\phi)\neq\emptyset,\phi\in\mathcal{F}}[:\phi\approx:\phi]]\}.$$

Then for each theory θ we form the theory $\theta^e=\theta\cup\theta_\Sigma$ which is semantically equivalent to the original theory (because we add only the information from the signature which is always taken into account, when a theory is interpreted). We convert the theory θ^e into an exclusive matrix which in turn is converted into a set of graphs $\mathcal{GR}$ called **graph representation** of θ.

The graph representation of a theory inherits from the exclusive matrixes their properties: (1) each graph $\mathcal{G}$ in $\mathcal{GR}$ is satisfiable (for some interpretation $\mathcal{I}$, $\mathcal{R}_\mathcal{I}(\mathcal{G})\neq\emptyset$), and (2) each two graphs $\mathcal{G}_1,\mathcal{G}_2$ in $\mathcal{GR}$ have disjoint denotations (for each interpretation $\mathcal{I}$, $\mathcal{R}_\mathcal{I}(\mathcal{G}_1)\cap\mathcal{R}_\mathcal{I}(\mathcal{G}_2)=\emptyset$). We can reformulate here also the Prop. 2.

Proposition 4 *Let θ be a finite SRL theory with respect to a finite signature, μ be the corresponding exclusive matrix, $\mathcal{GR}$ be the graph representation of θ and $\mathcal{I}=\langle\mathcal{U}_\mathcal{I},\mathcal{S}_\mathcal{I},\mathcal{F}_\mathcal{I}\rangle$ be a model of θ. For each object $\upsilon\in\mathcal{U}$ there exists a unique graph $\mathcal{G}\in\mathcal{GR}$ such that $\upsilon\in\mathcal{R}(\mathcal{G})$.*

There exists also a correspondence between complete graphs with respect to a finite signature and the objects in an interpretation of the signature.

Definition 5 (Object Graph) *Let $\Sigma=\langle\mathcal{S},\mathcal{F},\mathcal{A}\rangle$ be a finite signature, $\mathcal{I}=\langle\mathcal{U}_\mathcal{I},\mathcal{S}_\mathcal{I},\mathcal{F}_\mathcal{I}\rangle$ be an interpretation of Σ and υ be an object in $\mathcal{U}$, then the graph $\mathcal{G}_\upsilon=\langle\mathcal{N},\mathcal{V},\rho,\mathcal{S}\rangle$, where*

$\mathcal{N}=\{\upsilon'\in\mathcal{U}\mid\exists\tau\in\mathcal{TM}$ and $\mathcal{P}(\tau)(\upsilon)=\upsilon'\}$

$\mathcal{V}:\mathcal{N}\times\mathcal{F}\rightarrow\mathcal{N}$ is a partial **arc function***, such that*

$\mathcal{V}\langle\upsilon_1,\phi\rangle\downarrow$ and $\mathcal{V}\langle\upsilon_1,\phi\rangle=\upsilon_2$ iff

$\upsilon_1\in\mathcal{N},\upsilon_2\in\mathcal{N},\phi\in\mathcal{F}$, and $\mathcal{F}_\mathcal{I}(\phi)(\upsilon_1)=\upsilon_2$,

$\rho=\upsilon$ is the root node, and

$\mathcal{S}:\mathcal{N}\rightarrow\mathcal{S}$ is a **species assignment function***, such that*

$\mathcal{S}\langle\upsilon'\rangle=\mathcal{S}_\mathcal{I}\langle\upsilon'\rangle$,

is called **object graph**.

It is trivial to check that each object graph is a complete feature graph. Also, one easy can see the connection between the graphs in the graph representation of a theory and object graphs of objects in a model of the theory.

Proposition 6 *Let θ be a finite SRL theory with respect to a finite signature, $\mathcal{GR}$ be the graph representation of θ, $\mathcal{I} = \langle \mathcal{U}_\mathcal{I}, \mathcal{S}_\mathcal{I}, \mathcal{F}_\mathcal{I} \rangle$ be a model of θ, υ be an object in $\mathcal{U}_\mathcal{I}$, and $\mathcal{G}_\upsilon = \langle \mathcal{N}, \mathcal{V}, \rho, \mathcal{S} \rangle$ be its object graph. For each node $\nu \in \mathcal{N}$, there exists a graph $\mathcal{G}_i \in \mathcal{GR}$, such that $\mathcal{G}\,|_\nu \sqsubseteq \mathcal{G}_i$.*

This can be proved by using the definition of a model of a theory, the Prop. 4 and the definition of a subgraph started at a node.

3.6. Outcomes: Feature Graphs for HPSG Grammar and Corpus

Thus we can sum up that feature graphs can be used for both:

- Representation of an HPSG corpus. Each sentence in the corpus is represented as a complete feature graph. One can easily establish a correspondence between the objects in an exhaustive model of (King 1999) and complete feature graphs or a correspondence between the elements of strong generative capacity of (Pollard 1999) and complete feature graphs. Thus complete feature graphs are a good representation for an HPSG corpus;

- Representation of an HPSG grammar as a set of feature graphs. The construction of a graph representation of a finite theory demonstrates that using feature graphs as grammar representation does not impose any restrictions over the class of possible finite grammars in SRL. Therefore we can use feature graphs as a representation of the grammar used during the construction of an HPSG corpus, as described above.

Additionally, we can establish a formal connection between a grammar and a corpus using the properties of feature graphs.

Definition 7 (Corpus Grammar) *Let C be an HPSG corpus and Γ be an HPSG grammar. We say that Γ is a* **grammar of the corpus** *C if and only if for each graph $\mathcal{G}_C$ in C and each node $\nu \in \mathcal{G}_C$ there is a graph $\mathcal{G}_G$ in G such that $\mathcal{G}_C\,|_\nu \sqsubseteq \mathcal{G}_G$.*

It follows by the definition that if C is an HPSG corpus and Γ is a corpus grammar of C then Γ accepts all analyses in C.

4. Incremental Specialization using Negative Information

Let us now return to the annotation process. We start with an HPSG grammar which together with the signature determines the annotation scheme. We convert this grammar into a graph representation $\mathcal{GR}_0$. In the project we rely on the existing system (TRALE) for processing of HPSG grammars (TRALE is based on (Götz and Meurers, 1997)). TRALE works with HPSG grammars represented as general descriptions, but the result from the sentence processing is equivalent to a complete feature graph. It is also relatively easy to convert the grammar into a set of feature graphs.

Having $\mathcal{GR}_0$ we can analyze partial analyses of the sentences as it was described in the introduction. The partial analyses are used in order to reduce the number of the possible analyses. Let us suppose that the set of complete feature graphs $\mathcal{GRA}$ is returned by the TRALE system. Then these graphs are processed by the annotator within the CLaRK system and some of the analyses are accepted to be true for the sentence. Thus, they are added to the corpus and the rest of the analyses are rejected. Let $\mathcal{GRN}$ be the set of rejected analyses and $\mathcal{GRC}$ be the set of all analyses in the corpus up to now plus the new accepted ones. Our goal now is to specialize the initial grammar $\mathcal{GR}_0$ into a grammar $\mathcal{GR}_1$ such that it is still a grammar of the corpus $\mathcal{GRC}$ and it does not derive any of the graphs in $\mathcal{GRN}$. Using Prop. 6 we can rely on a very simple test for acceptance or rejection of a complete graph by the grammar: "If for each node in a complete graph there exists a graph in the grammar that subsumes the subgraph started at the same node, then the complete graph is accepted by the grammar." So, in order to reject a graph $\mathcal{G}$ in $\mathcal{GRN}$ it is enough to find a node ν in $\mathcal{G}$ such that for the subgraph $\mathcal{G}\,|_\nu$ there is no graph $\mathcal{G}' \in \mathcal{GR}_1$ such that $\mathcal{G}\,|_\nu \sqsubseteq \mathcal{G}'$. We will use this dependency in the process of guiding the specialization of the initial grammar.

In order to apply this test we have to consider not only the graphs in $\mathcal{GRC}$ and $\mathcal{GRN}$, but also their complete subgraphs. We process further the graphs in $\mathcal{GRN}$ and $\mathcal{GRC}$ in order to determine which information encoded in these graphs is crucial for the rejection of the graphs in $\mathcal{GRN}$. Let $sub(\mathcal{GRN})$ be the set of the complete graphs in $\mathcal{GRN}$ and their complete subgraphs and let $sub(\mathcal{GRC})$ be the set of the complete graphs in $\mathcal{GRC}$ and their complete subgraphs. We divide the set $sub(\mathcal{GRN})$ into two sets: $\mathcal{GRN}^+$ and $\mathcal{GRN}^-$, where $\mathcal{GRN}^+ = sub(\mathcal{GRN}) \cap sub(\mathcal{GRC})$ contains all graphs that are equivalent to some graph as well in $\mathcal{GRP}$[3] and $\mathcal{GRN}^- = sub(\mathcal{GRN}) \setminus sub(\mathcal{GRC})$ contains subgraphs that are presented only in $sub(\mathcal{GRN})$.

Then we choose all graphs $\mathcal{G}$ in $\mathcal{GR}_0$ such that for some $\mathcal{G}' \in \mathcal{GRN}^-$ it holds $\mathcal{G}' \sqsubseteq \mathcal{G}$. Let this set be $\mathcal{GR}_0^-$. This is the set of graphs in the grammar $\mathcal{GR}_0$ which we have to modify in order to achieve our goal.

Then we select from $sub(\mathcal{GRC})$ all graphs such that they are subsumed by some graph from $\mathcal{GR}_0^-$. Let this set be $\mathcal{GRP}$. These are the graphs that might be rejected by the modified grammar. Thus, the algorithm has to disallow such a rejection.

Thus our task is to specialize the graphs in the set $\mathcal{GR}_0^-$ in such a way that the new grammar (after substitution of $\mathcal{GR}_0^-$ with the new set of more specific graph into $\mathcal{GR}_0$) accepts all graphs in $\mathcal{GRP}$ and rejects all graphs in $\mathcal{GRN}$.

The algorithm works by performing the following steps:

[3] This is based on the fact that the accepted analyses can share some subgraphs with the rejected analyses.

1. It calculates the set $\mathcal{GRN}^-$;
2. It selects a subset $\mathcal{GR}_0^-$ of $\mathcal{GR}_0$;
3. It calculates the set $\mathcal{GRP}$;
4. It tries to calculate a new set of graphs $\mathcal{GR}_1^-$ such that each graph $\mathcal{G}$ in the new set $\mathcal{GR}_1^-$ is either member of $\mathcal{GR}_0^-$ or it is subsumed by a graph in $\mathcal{GR}_0^-$. Each new graph in $\mathcal{GR}_1^-$ can not have more nodes than the nodes in the biggest graph in the sets $\mathcal{GRP}$ and $\mathcal{GRN}$. This condition ensures the algorithm termination. If the algorithm succeeds to calculate a new set $\mathcal{GR}_1^-$ then it proceeds with the next step. Otherwise it stops without producing a specialization of the initial grammar.
5. It checks whether each graph in $\mathcal{GRP}$ is subsumed by a graph in $\mathcal{GR}_1^-$. If 'yes' then it prolongs the execution with the next step. Otherwise it returns to step 4 and calculates a new set $\mathcal{GR}_1^-$.
6. It checks whether there is a graph in $\mathcal{GRN}$ such that it is subsumed by a graph in $\mathcal{GR}_1^-$ and all its complete subgraphs in $\mathcal{GRN}^-$ are subsumed by a graph in $\mathcal{GR}_1^-$. If 'yes' then it returns to step 4 and calculates a new set $\mathcal{GR}_1^-$. Otherwise it returns the set $\mathcal{GR}_1^-$ as a specialization of the grammar $\mathcal{GR}_0^-$.

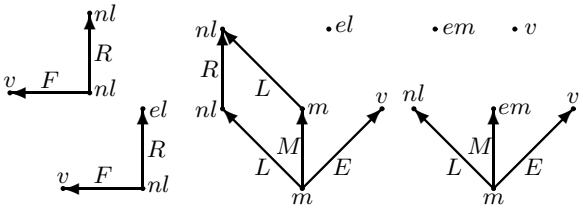
When the algorithm returns a new set of graphs $\mathcal{GR}_1^-$ which is a specialization of the graph set $\mathcal{GR}_0^-$, then we substitute the graph set $\mathcal{GR}_0^-$ with $\mathcal{GR}_1^-$ in the grammar $\mathcal{GR}_0$ and the result is a new, more specific grammar $\mathcal{GR}_1$ such that it accepts all graphs in the corpus $\mathcal{GRC}$ and rejects all graphs in $\mathcal{GRN}$.

In general, of course, there exist more than one specialization. Deciding which one is a good one becomes a problem, which cannot be solved only on the base of the graphs in the two sets $\mathcal{GRP}$ and $\mathcal{GRN}$. In this case two repairing strategies are possible: either additional definition of criteria for choosing the best extension, or the application of some statistical evaluations.

If the algorithm fails to produce a new set of graphs $\mathcal{GR}_1^-$ then there is an inconsistency in the acceptance of the graphs in $\mathcal{GRC}$ and/or in the rejection of the graphs in $\mathcal{GRN}$. This could happen if the annotator marks as wrong an analysis (or a part of it) which was marked as true for some previous analysis.

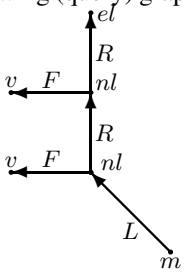
5. Example

In this section we present an example. This example is based on the notion of list and member relation encoded as feature graphs. The lists are encoded by two species: nl for non-empty lists and el for empty lists. Two features are defined for non-empty lists: F for the first element of the list and R for the rest of the list. The elements of a list are of species v. The member relation is also encoded by two species: m for the recursive step of the relation and em for the non-recursive step. For the recursive step of the relation (species m) three features are defined: L pointing to the list, E for the element which is a member of the list and M for the next step in the recursion of the relation. The next set of graphs constitutes an incomplete grammar for member relation on lists. The incompleteness results from the fact that there is no restriction on the feature E.



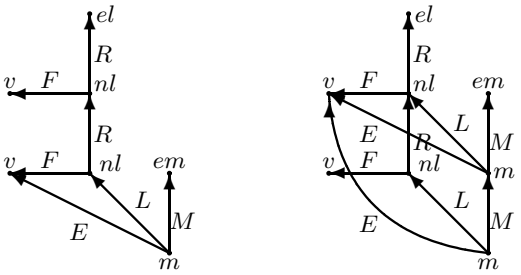
Here the two graphs on the left represent the fact that the rest of a non-empty list could be a non-empty list or an empty list. They also state that each non-empty list has a value. Then there are two graphs for the species m. The first states that the relation member can have a recursive step as a value for the feature M if and only if the list of the second recursive step is the rest of the list of the first recursive step. The second graph just completes the appropriateness for the species m saying that the value of the feature L is also of species non-empty list when the value of the feature M is non-recursive step of the member relation. There are also three graphs with single nodes for the case of empty lists, non-recursive steps of member relations and for the values of the lists. They are presented at the top right part of the picture. Now let us suppose that the annotator would like to enumerate all members of a two-element list by evaluation of the following (query) graph with respect to the above grammar.

Query graph:



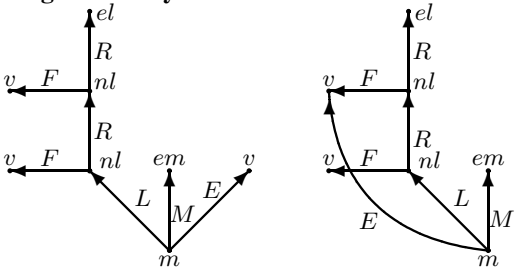
The grammar returns two acceptable analyses. One for the first element of the list and one for the second element of the list.

Positive analyses:

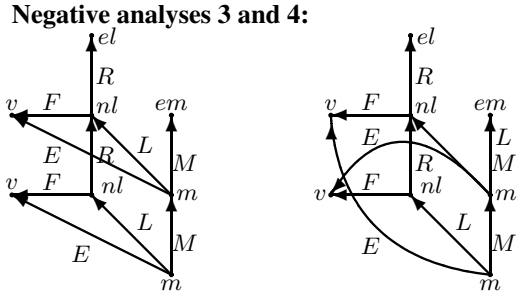


The grammar also accepts 11 wrong analyses in which the E features either point to wrong elements of the list or they are not connected with element of the list at all. Here are the wrong analyses.

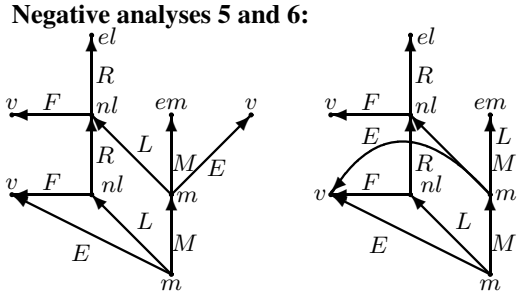
Negative analyses 1 and 2:



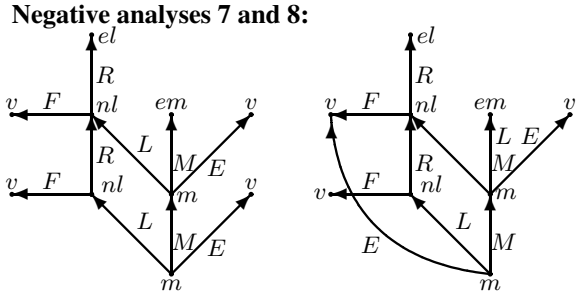
Negative analyses 3 and 4:



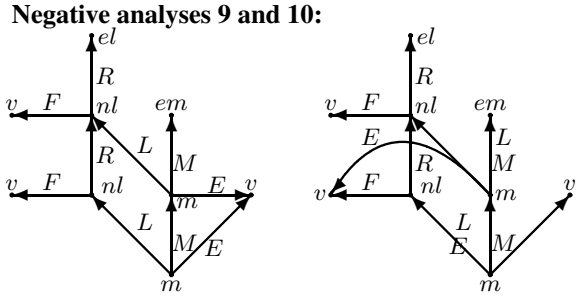
Negative analyses 5 and 6:



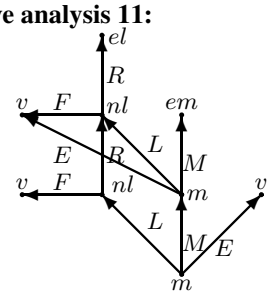
Negative analyses 7 and 8:



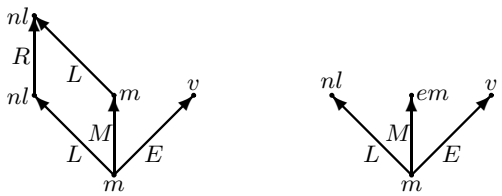
Negative analyses 9 and 10:



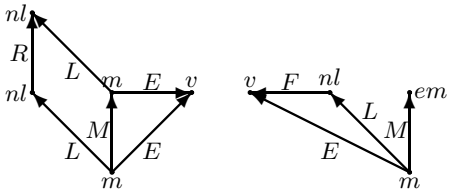
Negative analysis 11:



The next step is to determine the set $\mathcal{GRN}^-$. This set contains 12 complete graphs: all graphs in the set $\mathcal{GRN}$ and one subgraph that is not used in the positive analyses. We will not list these graphs here. The graphs from the grammar that subsumes the graphs in $\mathcal{GRN}^-$ are the two graphs for the member relation. We repeat them here.



Now we have to make them more specific in order to reject the negative examples from $\mathcal{GRN}^-$ but still to accept the two positive examples. The next two graphs are an example of such more specific graphs.



By the first graph the negative examples 3, 4, 5, 7, 8, 10 and 11 are rejected, and by the second graph the negative examples 1, 2, 5, 6, 7, 8, 9, 10 are rejected. Thus both specializations are necessary in order to reject all negative examples. The new grammar still accepts the two positive examples.

6. Conclusion

The presented approach is still very general. It defines a declarative way to improve an annotation HPSG grammar represented as a set of feature graphs. At the moment we have implemented only partially the connection between TRALE system and CLaRK system. Thus, a demonstration of the practical feasibility of the approach remains for future work.

Similar approach can be established on the base of the positive information only (see (Simov, 2001) and (Simov, 2002)), but the use of the negative information can speed up the algorithm. Also, the negative as well as positive information can be used in creation of a performance model for the new grammar along the lines of (Bod, 1998).

7. Acknowledgements

The work reported here is done within the BulTreeBank project. The project is funded by the Volkswagen Stiftung, Federal Republic of Germany under the Programme *"Co-operation with Natural and Engineering Scientists in Central and Eastern Europe"* contract I/76 887.

We would like to thank Petya Osenova for her comments on earlier versions of the paper. All errors remain ours, of course.

8. References

Rens Bod. 1998. *Beyond Grammar: An Experience-Based Theory of Language*. CSLI Publications, CSLI, California, USA.

Bob Carpenter. 1992. *The Logic of Typed Feature Structures*. Cambridge Tracts in Theoretical Computer Science 32. Cambridge University Press.

T. Götz and D. Meurers. 1997. *The ConTroll system as large grammar development platform*. In *Proceedings of the ACL/EACL post-conference workshop on Computational Environments for Grammar Development and Linguistic Engineering*. Madrid, Spain.

P.J. King. 1989. *A Logical Formalism for Head-Driven Phrase Structure Grammar*. Doctoral thesis, Manchester University, Manchester, England.

P.J. King. 1999. *Towards Thruth in Head-Driven Phrase Structure Grammar*. In V. Kordoni (Ed.), *Tübingen Studies in HPSG,* Number 132 in Arbeitspapiere des SFB 340, pp 301-352. Germany.

P. King and K. Simov. 1998. The automatic deduction of classificatory systems from linguistic theories. In *Grammars*, volume 1, number 2, pages 103-153. Kluwer Academic Publishers, The Netherlands.

P. King, K. Simov and B. Aldag. 1999. The complexity of modelability in finite and computable signatures of a constraint logic for head-driven phrase structure grammar. In *The Journal of Logic, Language and Information*, volume 8, number 1, pages 83-110. Kluwer Academic Publishers, The Netherlands.

C.J. Pollard and I.A. Sag. 1987. *Information-Based Syntax and Semantics*, vol. 1. CSLI Lecture Notes 13. CSLI, Stanford, California, USA.

C.J. Pollard and I.A. Sag. 1994. *Head-Driven Phrase Structure Grammar*. University of Chicago Press, Chicago, Illinois, USA.

C.J. Pollard. 1999. *Strong Generative Capacity in HPSG.* in Webelhuth, G., Koenig, J.-P., and Kathol, A., editors, *Lexical and Constructional Aspect of Linguistic Explanation,* pp 281-297. CSLI, Stanford, California, USA.

K. Simov. 2001. *Grammar Extraction from an HPSG Corpus.* In: Proc. of the RANLP 2001 Conference, Tzigov chark, Bulgaria, 5–7 Sept., pp. 285–287.

K. Simov, G. Popova, P. Osenova. 2001. *HPSG-based syntactic treebank of Bulgarian (BulTreeBank).* In: *"A Rainbow of Corpora: Corpus Linguistics and the Languages of the World",* edited by Andrew Wilson, Paul Rayson, and Tony McEnery; Lincom-Europa, Munich, pp. 135–142.

K. Simov, Z. Peev, M. Kouylekov, A. Simov, M. Dimitrov, A. Kiryakov. 2001. *CLaRK - an XML-based System for Corpora Development.* In: Proc. of the Corpus Linguistics 2001 Conference, pages: 558-560.

K. Simov. 2002. *Grammar Extraction and Refinement from an HPSG Corpus.* In: Proc. of ESSLLI-2002 Workshop on Machine Learning Approaches in Computational Linguistics, August 5-9.(to appear)

K.Simov, P.Osenova, M.Slavcheva, S.Kolkovska, E.Balabanova, D.Doikoff, K.Ivanova, A.Simov, M.Kouylekov. 2002. *Building a Linguistically Interpreted Corpus of Bulgarian: the BulTreeBank.* In: Proceedings from the LREC conference, Canary Islands, Spain.

A Bootstrapping Approach to Automatic Annotation of Functional Information to Adjectives with an Application to German

Bernd Bohnet, Stefan Klatt and Leo Wanner

Computer Science Department
University of Stuttgart
Breitwiesenstr. 20-22
70565 Stuttgart, Germany
{bohnet|klatt|wanner}@informatik.uni-stuttgart.de

Abstract

We present an approach to automatic classification of adjectives in German with respect to a range functional categories. The approach makes use of the grammatical evidence that (i) the functional category of an adjectival modifier determines its relative ordering in an NP, and (ii) only modifiers that belong to the same category may appear together in a coordination. The coordination context algorithm is discussed in detail. Experiments carried out with this algorithm are described and an evaluation of the experiments is presented.

1. Introduction

Traditionally, corpora are annotated with POS, syntactic structures, and, possibly, also with word senses. However, for certain word categories, further types of information are needed if the annotated corpora are to serve as source, e.g., for the construction of NLP lexica or for various NLP-applications. Among these types of information are the semantic and functional categories of adjectives that occur as premodifiers in nominal phases (NPs) (Raskin and Nirenburg, 1995). In this paper, we focus on the functional categories such as 'deictic', 'numerative', 'epithet', 'classifying', etc. As is well-known from the literature (Halliday, 1994; Engel, 1988), the functional category of an adjectival modifier in an NP predetermines its relative ordering with respect to other modifiers in the NP in question, the possibility of a coordination with other modifiers, and to a certain extent, also the reading in the given communicative context. Consider, e.g. in German,

(1) *Viele junge kommunale Politiker ziehen aufs Land*
'Many young municipal politicians move to the country side'.

but

**Viele kommunale junge Politiker ziehen aufs Land*
'Many municipal young politicians move to the country side'.

(2) *Viele ehemalige Politiker ziehen aufs Land*
'Many previous politicians move to the country side.'

but

**Ehemalige viele Politiker ziehen aufs Land*
'Previous many politicians move to the country side.'

Jung 'young' and *kommunal* 'municipal', *viele* 'many' and *ehemalig* 'previous' belong to different functional categories, which makes them unpermutable in the above NPs and implies a specific relative ordering: category(*jung*) $<$ category(*kommunal*) and category(*viele*) $<$ category(*ehemalig*). In contrast, *jung* 'young' and *dynamisch* 'dynamic' belong to the same category; they can be permuted in an NP without an impact on the grammaticality of the example:

(3) *Viele junge, dynamische Politiker ziehen aufs Land*
'Many young, dynamic politicians move to the country side'.

and

Viele dynamische, junge Politiker ziehen aufs Land
'Many dynamic, young politicians move to the country side'.

They can also appear in a coordination:

(4) *Viele junge und dynamische Politiker ziehen aufs Land*
'Many young and dynamic politicians move to the country side'.

Viele dynamische und junge Politiker ziehen aufs Land
'Many dynamic and young politicians move to the country side'.

while, e.g., *viele* and *kommunal* cannot:

(5) **Viele junge und kommunale Politiker ziehen aufs Land*
'Many young and municipal Stuttgart politicians move to the country side'.

In such applications as natural language generation and machine translation, it is important to have the function of the adjectives specified in the lexicon. However, as yet, no large lexica are available that would contain this information. Therefore, an automatic corpus-based annotation of functional information seems the most suitable option.

In what follows, we present a bootstrapping approach to the functional annotation of German adjectives in corpora. The next section presents a short outline of the theoretical assumptions we make with respect to the function of adjectival modifiers and their occurrence in NPs and coordination contexts, before in Section 3. the preparatory stage

and the annotation algorithms are specified. Section 4. contains then the description of the experiments we carried out in order to evaluate our approach, and Section 5. contains the discussion of these experiments. In Section 6., we give some references to work that is related to ours. In Section 7., finally, we draw some conclusions and outline the directions we intend to take in this area in the future.

2. The Grammatical Prerequisits

Grammarians often relate the default ordering of adjectival modifiers to their semantic or functional categories; see, among others, (Dixon, 1982; Engel, 1988; Dixon, 1991; Frawley, 1992; Halliday, 1994). (Vendler, 1968) motivates it by the order of the transformations for the derivation of the NP in question. (Quirk et al., 1985) state that the position of an adjective in an NP depends on how inherent this adjective's meaning is: adjectives with a more inherent meaning are placed closer to the noun than those with a less inherent meaning. (Seiler, 1978) and (Helbig and Buscha, 1999) argue that the order is determined by the scope of the individual adjectival modifiers in an NP. For an overview of the literature on the topic, see, e.g., (Raskin and Nirenburg, 1995).

As mentioned above, we follow the argumentation that the order of adjectives in an NP is determined by their functional categories. In this section, we first outline the range of functions of adjectival modifiers known from the literature especially for German, present then the function-dependent default ordering, and discuss, finally, the results of an empirical study carried out to verify the theoretical postulates and thus to prepare the grounds for the automatic functional category annotation procedure.

2.1. Ranges of Functions of Adjectival Modifiers

In the literature, different ranges of functional categories of adjectival premodifiers have been discussed. For instance, (Halliday, 1994), proposes for English the following categories of the elements in an NP that precede the noun:

(i) deictic: *this*, *those*, *my*, *whose*,...;

(ii) numerative: *many*, *second*, *preceding*, ...;

(iii) epithet: *old*, *blue*, *pretty*, ...;

(iv) classifier: *electric*, *catholic*, *vegetarian*, *Spanish*, ...

In (Engel, 1988), a slightly different range of categories is given for German adjectival premodifiers:

(i) quantitative: *viele* 'many', *einige* 'some', *wenige* 'few', ...

(ii) referential: *erst* 'first', *heutige* 'today's', *diesseitige* 'from-this-side', ...

(iii) qualificative: *schön* 'beautiful', *alt* 'old', *gehoben* 'upper', ...

(iv) classifying: *regional* 'regional', *staatlich* 'state', *katholisch* 'catholic', ...

(v) origin: *Stuttgarter* 'from-Stuttgart', *spanisch* 'Spanish', *marsianisch* 'from-Mars', ...

The function of a modifier may vary with the context of the NP in question or even be ambiguous (Halliday, 1994; Tucker, 1995). Thus, Ger. *zweit* 'second' belong to the referential category in the NP *zweiter Versuch* 'second attempt'; in *zweiter Preis* 'second price', it belongs to the classifying category. *Fast* in *fast train* can be considered as qualificative or as classifying (if *fast train* means 'train classified as express').

Two modifiers are considered to belong to the same category if they can appear together in a coordination or can be permutated in an NP:

(6) a. Ger. *eine rote oder weiße Rose*
'a red or a white rose'

b. *dritter oder vierter Versuch*
'third or fourth attempt'

c. *elektrische oder mechanische Schreibmaschine*
'an electric or mechanic typewriter'

but not

(7) a. $^{??}$*eine rote und langstielige Rose*
'a red and long-stemmed rose'

b. **rote und holländische Rosen*
'red and Dutch roses'

c. **eine schöne oder elektrische Schreibmaschine*
'a beautiful or electric typewriter'

The credibility of the coordination test is limited, however. Consider

(8) $^{??}$ *Eine schöne und rote Rose*
'a beautiful and red rose'

where *schön* 'beautiful' and *rot* 'red' both belong to the qualitative category, but still do not permit a coordination easily.

Adjectival modifier function taxonomies are certainly language-specific (Frawley, 1992). Nonetheless, as the taxonomies suggested by Halliday and Engel show, they may overlap to a major extent. Often, the difference is more of a terminological than of a semantic nature. In our work, we adopt Engel's taxonomy.

2.2. The Default Ordering of Adjectival Modifiers

Engel (Engel, 1988) suggests the following default ordering of modifier functions:
quantitative $<$ referential $<$ qualificative $<$ classifying $<$ origin

Cf, e.g.:

quant.	referent.	qual.	class.	origin
viele	ehemalige	junge	kommunale	Stuttgarter
'many'	'previous'	'young'	'municipal'	'Stuttgart'

as in

(9) *Viele ehemalige junge kommunale Stuttgarter Politiker ziehen aufs Land*
'Many previous young municipal Stuttgart politicians move to the country side'.

According to Engel, a violation of this default ordering leads to ungrammatical NPs. (1–3) in the Introduction illustrate this violation.

2.3. Empirical Evidence for the Theoretical Claims

In the first stage of our work, we sought empirical evidence for the theoretical claims with respect to the functional category motivated ordering and the functional category motivated coordination restrictions. Although, in general, these claims have been buttressed by our study, counterexamples were found in the corpus with respect to both of them.

2.3.1. Default Ordering: Counterexamples

Especially adjectives of the category 'origin' tended to occur before classifying or qualificative modifiers instead of being placed immediately left to the noun–as would be required by the default ordering. For instance, *spanisch* 'Spanish' occured in 3.5% of its occurrences in the corpus in other positions; cf., for illustration:

(10) a. (*das*) *spanische höfische Bild*
'(the) Spanish courtly picture'

b. (*der*) *spanische schwarze Humor*
'(the) Spanish black humour'

c. (*der*) *spanischen sozialistischen Partei*
'(the) Spanish socialist party$_{\mathrm{dat}}$'

To be noted is that in such NPs as (*der*) *spanische schwarze Humor* and *deutsche katholische Kirche* 'German catholic church' the noun and the first modifier form a multiword lexeme rather than a freely composed NP (i.e. *schwarzer Humor* 'black humour' and *katholische Kirche* 'catholic church'). That is, the preceding modifiers (*spanisch* 'Spanish'/*deutsch* 'German') function as modifiers of the respective multiword lexeme, not of the noun only. This is also in accordance with (Helbig and Buscha, 1999)'s scope proposal.

2.3.2. Coordination Restrictions: Counterexamples

It is mainly ordinals that occur, contrary to the theoretical claim, in coordinations with modifiers that belong to a different category. For instance, *erst* 'first' appears in the corpus in 9.74% cases of its occurrence in such "heterogeneous" coordinations. Cf., for illustration:

(11) a. (*die*) *erste und wichtigste Aufgabe*
'(the) first and the most important task'

b. (*eines der*) *ersten und augenfälligsten Projekte*
'one of the first and conspicuous projects'

c. (*die*) *oberste und erste Pflicht*
'(the) supreme and first duty'

As a rule, in such cases the ordinals have a classifying function, which is hard to capture, however.

2.3.3. Grammaticality of the Counterexamples

An evaluation of the counterexamples found in the corpus revealed that not all of these examples can, in fact, be considered as providing counter evidence for the theoretical claims. The grammaticality of a considerable number of these examples has been questioned by several speakers of German; cf., for instance:

(12) a. *(*die*) *ersten und fehlerhaften Informationen*
'(the) first and erroneous informations'

b. $^{??}$*jüngster und erster Präsident*
'youngest and first president'

c. $^{??}$(*die*) *oberste und erste Pflicht*
'(the) supreme and first duty'

3. The Approach

The empirical study of the relative ordering of adjectival modifiers in NPs and of adjectival modifier coordinations in the corpus showed that the theoretical claims made with respect to the interdependency between functional categories and ordering respectively coordination context restrictions are not always proved right. However, deviances from these claims encountered are not numerous enough to question these claims. Therefore, in our approach to the automatic annotation of adjectival modifiers in NPs with functional information outlined below, we make use of them.

The basic idea underlying the approach can be summarized as follows:

1. take a small set of samples for each functional category as point of departure;
2. look in the corpus for coordinations in which one of the elements is in the starting set (and whose functional category is thus known) and the other element is not yet annotated and annotate it with the category of the first element;

   alternatively:

   look in the corpus for all NP-contexts in which one of the elements is in the starting set, assign to its left and right neighbors all categories that these can may have according to the default ordering;
3. attempt to further constrict the range of categories of all modifiers that are still assigned more than one category;
4. add the unambiguously annotated modifers to the set of samples and repeat the annotation procedure;
5. terminate if all adjectival modifiers have been annotated a unique functional category or no further constrictions are possible.

Note that we do not take the punctuation rule into account, which states that adjectival modifiers of the same category are separated by a comma, while modifiers of different categories are not separated. This is because this rule is considered to be unreliable in practice. Furthermore, we do not use such hints as that classifying modifiers do not appear in comparative and superlative forms. See, however, Section 7.

3.1. The Preparatory Stage

The preparatory stage consists of three phases: (i) preprocessing the corpus, (ii) pre-annotation of modifiers whose category is *a priori* known, and (iii) compilation of the sets of modifiers from which the annotation algorithms start.

3.1.1. Preprocessing the Corpus

To have the largest possible corpus at the lowest possible cost, we start with a corpus that is not annotated with POS. When preprocessing the corpus, first token sequences are identified in which one or several tokens with an attributive adjectival suffix (*-e*, *-es*, *-en*, *-er*, or *-em*) are written in small letters and are followed by a capitalized token assumed to be a noun.[1] The tokens with an attributive suffix may be separated by a blank, a comma or have the conjunction *und* 'and' or the disjunction *oder* 'or' in between: cf.:

(13) a. (*das*) *erste richtige Beispiel*
'(the) first correct example'

b. *rote, blaue und grüne oder schwarze Hosen*
'red, blue and green or black pants'

Note that this strategy does not capture certain marginal NP-types; e.g.:

(a) NPs with an irregular adjectival suffix; e.g., *-a*: (*eine*) *lila Tasche* '(a) purple bag', *rosa Haare* 'pink hair', etc.;

(b) NPs with adjectival modifiers that start with a capital.

However, NPs of type (a) are very rare and can more reliably be annotated manually. NPs of type (b) are, first of all, modifiers at the beginning of sentences and attributive uses of proper nouns; cf. *Sorgenloses* 'free of care' in *Sorgenloses Leben – das ist das, was ich will!* lit. 'Free-of-care life—this is what I want' and *Franfurter* 'Frankfurt' in *Frankfurter Würstchen* 'Frankfurt sausages'. The first type appears very seldom in the corpus and can thus be neglected; for the second type, other annotation strategies proved to be more appropriate (Klatt, forthcoming).

After the token sequence identification, wrongly selected sequences are searched for (cf., e.g., *eine schöne Bescherung* 'a nice mess', where *eine* 'a' is despite its suffix obviously not an adjective but an article). This is done by using a morphological analysis program.

3.1.2. Pre-Annotation

In the pre-annotation phase, the following tasks are carried out:

- Adjectival modifiers of the category 'quantitative' are manually searched for and annotated. This is because the set of these modifiers is very small (*einige* 'some', *wenige* 'few', *viele* 'many', *mehrere* 'several') and would not justify the attempt of an automatic annotation.
- In (Engel, 1988), ordinals are by default considered to be referential. Therefore, we use a morphological analysis program to identify ordinals in order to annotate them accordingly in a separate procedure.
- Engel considers attributive readings of verb participles to be qualitative. This enables us to annotate participles with the qualitative function tag before the actual annotation algoritm is run.

3.1.3. Compiling the Starting Sets

Once the corpus is preprocessed and the pre-annotation is done, the starting sample sets for the annotation algorithms are compiled: for each category, a starting set of samples is manually chosen. The number of samples in each set is not fixed. In the experiments we carried out to evaluate our approach the size of sets varied from one to four (cf. Tables 3 and 5 below).

3.2. The Annotation Algorithms

The annotation program consists of two algorithms that can be executed in sequence or independently of each other. The first algorithm processes coordination contexts only. The second algorithm processes NP-contexts in general.

3.2.1. The Coordination Context Algorithm

The coordination context algorithm makes use of the knowledge that two adjectival modifiers that appear together in a conjunction or disjunction belong to the same functional category. As mentioned above, it loops over the set of modifiers whose category is already known (at the beginning, this is the starting set) looking for coordinations in which one of the elements is member of this set and the other element is not yet annotated. The element not yet annotated is assigned the same category as carried by the element already annotated.

The algorithm can be summarized as follows:

1. For each starting set in the starting set configuration do:
   (a) Mark each element in the set as `starting element` and as `processed`.
   (b) Retrieve all coordinations in which one of the `starting elements` occurs;
   for the not yet annotated elements in the coordinations do
   – mark each of them as `preprocessed`;
   – annotate each of them with the same category as assigned to its already annotated respective neighbor;
   – make a frequency distribution of them.
   (c) determine the element in the above frequency distribution with the highest frequency that is not marked as `processed` and mark this element as the next iteration `candidate` of the functional category in question.
2. Take the next iteration `candidate` with the highest frequency of the sets of all categories and mark it as `processed`. Stop, if no next iteration `candidate`

[1] Recall that in German nouns are capitalized.

can be found in any of the newly annotated elements of one of the categories.

3. Find all new corresponding coordination neighbors, add these elements to the set of `preprocessed` elements for the given category and make a new frequency distribution.

4. Determine the next iteration `candidate` for the given category as done in step 1c.

5. Continue with step 2.

Note that the coordination context algorithm does not loop over one of the categories a predetermined number of times and passes on then to the next category in order to repeat the same procedure. Rather, the switch from category to category is determined solely on the basis of the frequency distribution: the most frequent modifier not yet annotated is automaticaly chosen for annotation—independently of the category that has been assigned before. This strategy has two major advantages:

- it takes into account that the distribution of the modifiers in the corpus over the functional categories is extremely unbalanced: the set of 'quantitatives' counts only a few members while the set of 'qualitatives' is very large.

- it helps avoid an effect of "over-annotation" in the course of which the choice of an element that has already been selected as next iteration `candidate` for a specific category as next iteration `candidate` for a different category would lead to a revision of the annotation of all other already annotated elements involved in coordinations with this element.

Especially the second advantage contributes to the quality of our annotation approach. However, obviously enough, this algorithm assigns only one functional category to each adjective. That is, a multiple category assignment that is desirable in certain contexts must be pursued by another algorithm. This is done by the NP-context algorithm discussed in the next subsection.

Table 1 shows a few iterations of the coordination context algorithm with the starting sets of Experiment 1 in Section 4.. Here and henceforth the functional categories are numbered as follows:

1	2	3	4	5
↓	↓	↓	↓	↓
quant.	referent.	qualificat.	class.	origin

In the first iteration, the most frequent "next iteration candidate" of category 1 is *solch* 'such'with a frequency of 10, the most frequent of category 2 is *letzt* 'last' with a frequency of 71, and so on. The candidate of category 4 *wirtschaftlich* 'economic' possesses the highest frequency; therefore it is chosen for annotation and taken as "next iteration starting element" (see Step 2 in the algorithm outline). After adding all elements that occur in a coordination with *wirtschaftlich* to the candidate list, in iteration 2 the next element for annotation (and thus also the starting element) is chosen. This is done as described above for Iteration 1.

It.	cat.1	cat.2	cat.3	cat.4	cat.5
1	solch (10)	letzt (71)	klein (195)	wirtschaftlich (350)	französisch (93)
2	solch (10)	letzt (71)	klein (195)	sozial (295)	französisch (93)
3	solch (10)	letzt (71)	klein (195)	kulturell (208)	französisch (93)
4	solch (10)	letzt (71)	klein (195)	gesellschaftlich (119)	französisch (93)
5	solch (10)	letzt (71)	mittler (370)	gesellschaftlich (119)	französisch (93)
6	solch (10)	letzt (71)	alt (84)	gesellschaftlich (119)	französisch (93)
7	solch (10)	letzt (71)	alt (84)	ökonomisch (105)	französisch (93)
8	solch (10)	letzt (71)	alt (84)	ökologisch (118)	französisch (93)
9	solch (10)	letzt (71)	alt (84)	militärisch (74)	französisch (93)
10	solch (10)	letzt (71)	alt (84)	militärisch (74)	amerikanisch (95)

Table 1: An excerpt of the first iterations of the coordination context algorithm

3.3. The NP-Context Algorithm

The NP-context algorithm is based on the functional category motivated relative ordering of adjectival modifiers in an NP as proposed by Engel (see Section 2.).

In contrast to the coordination-context algorithm, which always ensures a non-ambiguous determination of the category of an adjective, the NP-context algorithm is more of an auxiliary nature. It helps to (i) identify cases where an adjective can be assigned multiple categories, (ii) make hypotheses with respect to categories of adjectival modifiers that do not appear in coordinations, (iii) verify the category assignment of the coordination-context algorithm.

The NP-context algorithm allows for a non-ambiguous determination of the category only in the case of a "comprehensive" NP, i.e., when all positions of an NP (from 'quantitative' to 'origin' are instantiated. Otherwise, relative statements of the kind as in the following case are possible:

> Given the NP (*der*) *schöne, junge, grüne Baum* '(the) beautiful, young, green tree', from which we know that *jung* 'young' is qualitative, we can conclude that *schön* may belong to one of the following three categories: quantitative, referential, or also qualitative, and that *grün* is either qualitative or classifying.

In other words, the following rules underlie the NP-context algorithm:
Given an adjective in an NP whose category X is known:

- assign to all left neighbors of this adjective the categories Y with $Y = 1, 2, \ldots, X$ (i.e., all categories with the number $\leq X$)

- assign to all right neighbors of this adjective the categories Z with $Z = X, X+1, \ldots, 5$ (i.e., all categories with the number $\geq X$

The NP-context algorithm varies slightly depending on the task it is used for—the verification of the categories assigned by the coordination-context algorithm or putting forward hypotheses with respect to the category of adjectives. When being used for the first task, it looks as follows:

1. for all adjectives that received a category tag during the coordination-context algorithm do
   - overtake this tag for all instances of these adjectives in the NP-contexts
2. do for each `candidate` that has been annotated a category
   - for each of the five categories C do
     - assign tentatively C to `candidate`
     - evaluate the NP-context of `candidate` as follows:
       (a) if the other modifiers in the context do not possess category tags, mark the context as unsuitable for the verification procedure
       (b) else, if with respect to the numerical category labels (see above) there is a decreasing pair of adjacent labels (i.e. of neighbor adjectives), mark this NP-context as rejecting C as category of `candidate`, otherwise mark the NP-context as accepting C as category of `candidate`
3. Choose the category whose choice received the highest number of confirmative coordination contexts

Table 2 shows the result of the verification of the category of a few adjectives. The first column contains the adjective whose category is verified. The second column contains the numerical category labels; with a '+' the category prognosticated by the coordination-context algorithm is marked.[2] In the third column, the number of confirmations of the corresponding category by NP-contexts is indicated (i.e. in the case of *neu* 'new', 6083 NP-contexts confirm category 3 ('qualificative') of *neu*, 5048 confirm category 4 ('classifying'), etc.). In the fourth column, the number of NP-contexts is specified that do not provide any evidence for the corresponding category. And in the fifth column the number of NP-contexts is indicated that negate the corresponding function. For four adjectives in Table 2 (*neu* 'new', *groß* 'big', *finanziell* 'financial', and *bosnisch* 'Bosnian') the NP-context algorithm confirmed the category suggested by the coordination-context algorithm; for two adjectives different categories were suggested (for *deutsch* 'German' 4 (classifying) instead of 5 (origin) and for *politisch* 'political' 5 instead of 4).

In the current version of the NP-context algorithm, for adjectival modifiers of category 4 or 5, the correct category is quite often listed as the second best choice. To avoid an incorrect annotation, further measures need to be taken (see also Section 7.).

[2] In all six cases, the coordination-context algorithm assignment was correct.

neu	+3	6083	697	112
	4	5048	697	1147
	2	4289	697	1906
	1	4195	697	2000
	5	3360	697	2835
groß	+3	6015	353	74
	2	5314	353	775
	4	5070	353	1019
	1	4391	353	1698
	5	3634	353	2455
deutsch	4	4992	498	109
	+5	4933	498	168
	3	4911	498	190
	2	1111	498	3990
	1	397	498	4704
politisch	5	3615	253	11
	+4	3519	253	107
	3	3353	253	273
	2	267	253	3359
	1	160	253	3466
finanziell	+4	1322	130	1
	5	1321	130	2
	3	1310	130	13
	2	46	130	1277
	1	25	130	1298
bosnisch	+5	223	24	2
	4	217	24	8
	3	214	24	11
	2	17	24	208
	1	11	24	214

Table 2: Examples of categorial classification by the NP-context algorithm

4. Experiments with the Coordination Algorithm

To evaluate the performance of the algorithms suggested in the previous section, we carried out experiments in two phases, three experiments each phase. The phases varied with respect to the size of the corpora used; the experiments in each phase varied with respect to the size of the starter sets.

In what follows, the experiments with the coordination algorithm only are discussed.

4.1. The Data

The experiments of the first phase were run on the *Stuttgarter-Zeitung* (STZ) corpus, which contains 36 Mio tokens; the experiments of the second phase were run on the corpus that consisted of the STZ-corpus and the *Frankfurter Rundschau* (FR) corpus with 40 Mio tokens; cf. Table 3. The first row in Table 3 shows the number of adjectival modifier coordinations and the number of premodifier NPs without coordinations in the STZ-corpus and in the STZ+FR-corpus; the second row shows the number of different adjectives that occur in all of these constructions in the respective corpus.

	STZ		STZ+FR	
	coord	NP	coord	NP
# contexts	18648	67757	36985	120673
# diff. adjectives	5894	10035	8003	12993

Table 3: Composition of the adjectival premodifier contexts in our corpora

		number of adjectival mods.						
exp	type	2	3	4	5	6	7	$\sum$
1-3	coord	17228	1238	149	31	2		18648
1-3	NP	66692	1059	6				67757
4-6	coord	34035	2598	298	47	6	1	36985
4-6	NP	118886	1772	15				120673

Table 4: Statistics on the size of the adjectival groups in STZ and STZ+FR

This gives us a ratio of 6.7 between the number of NPs and the number of different adjectives (i.e., the average number of NPs in which a specific adjective occurs) for the STZ-corpus and a ratio of 10.0 for the STZ+FR-corpus. Not surprisingly, larger corpora show a higher adjective repetition rate than small corpora do.

Table 4 contains the statistics on the size of modifier coordinations and the number of adjectival modifiers in NPs in general across both of our corpora. Adjectival modifier groups of size 3 or greater were thus very seldom.

Table 5 contains the data on the composition of both corpora with respect to ordinals and participles of which we assume to know *a priori* to which category they belong: ordinals to the category 2 ('referential') and participles to the category 3 ('qualitative'); see Section 2.

The starter sets consisted for the experiments 1 and 4 of one sample per category: an adjectival modifier of the corresponding category with a high frequency in the STZ-corpus. For the experiments 2 and 5, two, respectively three, high frequency samples for each category were added to starter sets. For the experiments 3 and 6, the starter sets were further extended by an additional modifier which has been assigned a wrong category in the experiments before. Table 6 shows the composition of the starter sets used for the experiments.

Apart from these "regular" members of the starter sets, to the starter sets of category 2 all ordinals and to the starter sets of category 3 all participles available in the respective corpus were added.

To have reliable data for the evaluation of the performance of the annotation program, we let an independent expert annotate 1000 adjectives with functional category

	STZ		STZ+FR	
	ordin.	part.	ordin.	part.
# diff. modifs	24	2023	25	2851
# total occur.	914	5135	2291	10045

Table 5: The distribution of ordinals and participles in STZ and STZ+FR

exp.	cat.1	cat.2	cat.3	cat.4	cat.5
1/4	ander	heutig	groß	politisch	deutsch
2/5	ander	heutig	groß	politisch	deutsch
	solch	letzt	alt	demokratisch	amerikanisch
		einzig	rot	kommunal	französisch
3/6	ander	heutig	groß	politisch	deutsch
	solch	letzt	alt	demokratisch	amerikanisch
		einzig	rot	kommunal	französisch
		mittler	schön	katholisch	russisch

Table 6: The composition of the starter sets

exp.	in total	assigned	¬assigned	p (%)
1	5894	5515	379	82.90%
2	5894	5515	379	84.30%
3	5894	5515	379	84.44%

Table 7: Results of the experiments 1 to 3

information. The manually annotated data were then compared with the output of our program to estimate the precision figures (see below).

4.2. Phase 1 Experiments

In the experiments 1 to 3, we were able to assign a functional category to 93,6% of the adjectival modifiers with all three starter sets. In 379 cases, the program could not assign a category; we discuss these cases in Section 5.. Table 7 summarizes the results of the experiments 1 to 3 ('p' stands for "precision").

Many of the 1000 manually annotated tokens occur only a few times in the corpus (and appear thus in a few coordinations). Low frequency tokens negatively influence the precision rate of the algorithm. The diagrams in Figures 1 to 3 illustrate the number of erroneous annotations in the experiments 1 to 3 in relation to the number of coordinations in which a token chosen as next for annotation appears as element at the moment when *n* tokens from the manually annotated token set have already been annotated. For instance, in Experiment 1, the first time when less than or 100 coordinations are considered to determine the category of a token, 9 of the 1000 members of the test set were annotated correctly, the first time when less than or 75 coordinations are considered, 17 of 1000 received the correct category, the first time when less than or 50 coordinations are considered, 31 tokens received the correct category and one a wrong one. And so on. Note, when less than or 5 coordinations were considered for the first time, only 41 annotations (out of 565) were wrong. This gives us a precision rate of $((565 - 41)/565) \times 100 = 92.74\%$.

Figures 2 and 3 show the annotation statistics for Experiments 2 and 3. Note that in Experiment 2 the precision rate for high frequency adjectives is considerably better than in Experiment 1: when 5 coordination contexts are available for the annotation decision, only 26 mistakes were made (instead of 41 in Experiment 1). Figure 3 shows that by a further extension of the starter set, no reasonable improvement of the results is achieved.

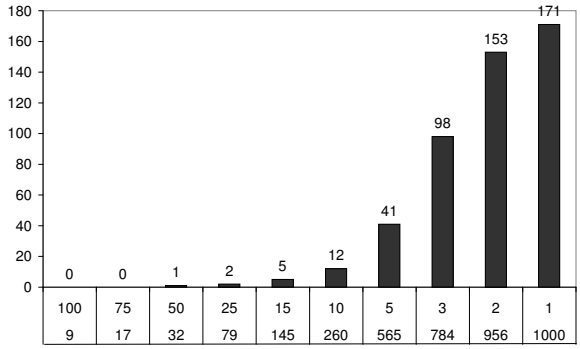


Figure 1: The annotation statistics in Experiment 1

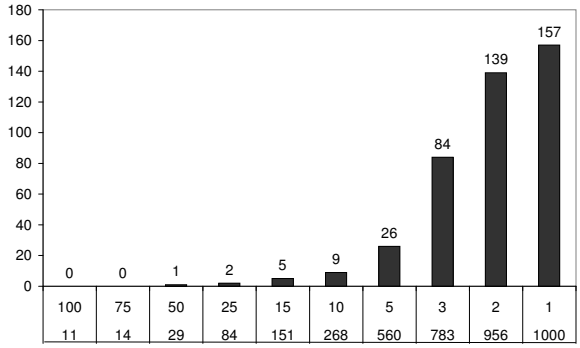


Figure 2: The annotation statistics in Experiment 2

4.3. Phase 2 Experiments

In experiments 4 to 6 we were able to assign with all three starter sets a functional category to 94,1% of the adjectival modifiers, i.e/, to 0.5% more than in the experiments of Phase 1. However, as Table 8 shows, the precision rate decreased slightly. Figures 4 to 6 show the annotation statistics for the Phase 2 experiments.

5. Discussion

In what follows, we first discuss the first 20 iterations of the coordination algorithm in Experiment 1 and Experiment 2, respectively, and present then the overall results of the experiments.

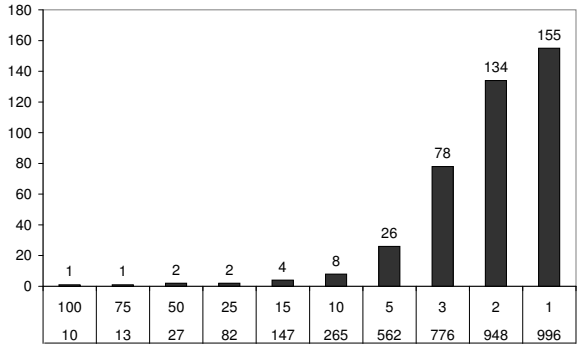


Figure 3: The annotation statistics in Experiment 3

exp.	in total	assigned	¬assigned	p (%)
4	8003	7558	445	84.08 %
5	8003	7558	445	84.08%
6	8003	7558	445	84.92%

Table 8: Results of the experiments 4 to 6

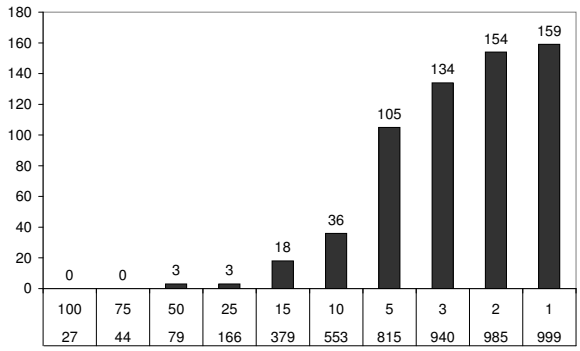


Figure 4: The annotation statistics in Experiment 4

5.1. A Snapshot of the Iterations in Experiments 1 and 2

Table 9 shows the first twenty iterations in Experiment 1, and Table 10 the first twenty iterations in Experiment 2. They look very similar despite the different starting sets in both experiments. Thus, in both nearly the same modifiers are annotated in nearly the same order—except *neu*, which is in Experiment 1 annotated in iteration 14, while in Experiment 2 in iteration 3. At the first glance, one might think that both experiments show the same results. However, as already pointed out above, the bigger starter set in Experiment 2 results in a considerably better precision rates with high and middle frequency adjectives.

5.2. Evaluation of the Experiments

Table 11 shows the distribution of the adjectival modifiers in the six experiments among the five functional categories.

Let us now consider some wrong annotations and some cases where the program was not able to assign a category.

In Table 12, some wrong annotations of category '3' (qualitative) in Experiment 1 are listed. The first column of the table specifies in which iteration of the algorithm the

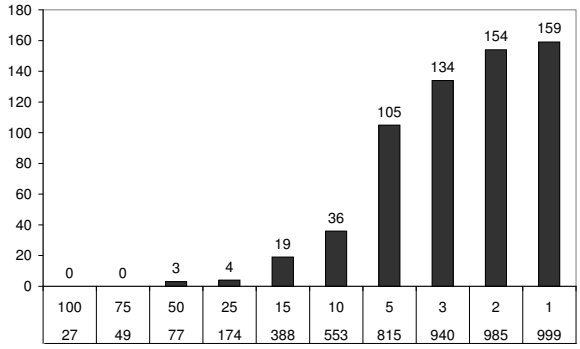


Figure 5: The annotation statistics in Experiment 5

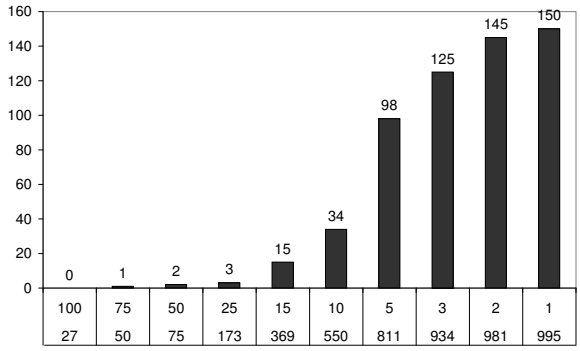


Figure 6: The annotation statistics in Experiment 6

Nr.	adjective	cat.	it. freq	freq
1	wirtschaftlich	4	350	851
2	sozial	4	295	707
3	kulturell	4	208	382
4	klein	3	195	688
5	mittler	3	370	482
6	gesellschaftlich	4	119	178
7	ökonomisch	4	105	167
8	ökologisch	4	118	164
9	französisch	5	93	251
10	amerikanisch	5	95	286
11	europäisch	5	102	179
12	ausländisch	5	88	128
13	alt	3	84	473
14	neu	3	307	417
15	britisch	5	81	100
16	italienisch	5	78	118
17	militärisch	4	74	127
18	letzt	2	71	99
19	finanziell	4	68	264
20	technisch	4	78	258

Table 9: The first 20 iterations in Experiment 1

Nr.	adjective	cat.	it. freq	freq
1	wirtschaftlich	4	356	851
2	sozial	4	307	707
3	neu	3	304	417
4	kulturell	4	210	382
5	klein	3	199	688
6	mittler	3	373	482
7	gesellschaftlich	4	119	178
8	ökonomisch	4	105	167
9	ökologisch	4	119	164
10	europäisch	5	102	179
11	ausländisch	5	88	128
12	britisch	5	81	100
13	italienisch	5	78	118
14	militärisch	4	74	127
15	finanziell	4	68	264
16	technisch	4	78	258
17	religiös	4	67	132
18	englisch	5	67	76
19	jung	3	63	112
20	personell	4	60	141

Table 10: The first 20 iterations in Experiment 2

exp.	cat.1	cat.2	cat.3	cat.4	cat.5	$\sum$
1	8	39	4506	711	251	5515
2	8	39	4434	785	249	5515
3	7	76	4377	791	264	5515
4	13	55	5938	1186	366	7558
5	13	55	5938	1186	366	7588
6	13	63	5926	1200	356	7558

Table 11: Distribution of the adjectival modifiers

Nr.	adjective	cat.	it. freq	freq
64	unter	3	33	43
151	marktwirtschaftlich	3	16	26
780	sozialdemokratisch	3	4	9
782	kommunistisch	3	4	17
807	katholisch	3	5	77
808	evangelisch	3	57	57
809	protestantisch	3	13	17
810	anglikanisch	3	5	5
811	reformerisch	3	4	4

Table 12: Some errors in Experiment 1

respective adjective has been assigned a category. 'it freq' (*iteration frequency*) specifies the number of the coordinations with this adjective as element that were available in the corresponding iteration; 'total freq' specifies how many times the adjective occured in coordinations in the corpus in total.

The correct category of *unter* 'under' would have been 2 ('referential'); that of *marktwirtschaftlich* 'free-enterprise' 4 ('classifying'), that of *kommunistisch* 'communist' 4, etc. Note the case of *katholisch* 'catholic'. Its total frequency of 77 is much higher as that of the adjectives processed before. However, it was chosen with an iteration frequency of only 5, i.e., only 5 coordinations have been considered to determine its category. The consequence is that the following adjectives (cf. iterations 808-811) also received a wrong annotation.

Table 13 shows the first 10 of the 445 adjectives that have not been assigned a category in Experiment 6.

Consider, e.g., the coordination constructions in which, e.g., *neunziger* 'ninety/nineties' occurs: *achtziger* 'eighty/eighties' COORD *neunziger* (11 times) and *siebziger* 'seventy/seventies' COORD *achtziger* COORD *neunziger* (1 time). That is, we run into a deadlock here:

	adjective	freq
1.	sechziger	248
2.	siebziger	195
3.	fünfziger	147
4.	dreißiger	102
5.	achtziger	93
6.	zwanziger	81
7.	vierziger	61
8.	zehner	21
9.	neunziger	12
10.	deutsch-polnisch	6

Table 13: Unprocessed adjectives in Experiment 6

gradable adj.
- scalar gradables
  - attitude-based
  - numerical scale
  - literal scale
  - member
- non-scalar gradables

non-scalar adj.
- proper non-scalars
- event-related non-scalars
- true relative non-scalars

Figure 7: The taxonomy that underlies the adjective classification by Raskin and Nirenburg

neunziger cannot be assigned a category because all its coordination neighbors did not receive a category either.

6. Related Work

To our knowledge, ours is the first approach to the automatic classification of adjectives with respect to a range of functional categories. In the past, approaches to the classification of adjectives focused on the classification with respect to semantic taxonomies. For instance, (Raskin and Nirenburg, 1995) discuss a manual classification procedure in the framework of the *MikroKosmos*. The taxonomy they refer to is is shown in Figure 7.

Obviously, an automatization of the classification with respect to this taxonomy is still beyond the state of the art in the field. On the other side, (Engel, 1988)'s functional categories seem to suffice to solve, e.g., the problem of word ordering in text generation.

(Hatzivassiloglou and McKeown, 1993) suggest an algorithm for clustering adjectives according to meaning. However, they do not refer to a predetermined (semantic) typology or set of functional categories.

(Hatzivassiloglou and McKeown, 1997) determine the orientation of the adjectives (negative vs. positive). The orientation is a useful lexical information since it has an impact on the use of adjectives in coordinations: only adjectives with the same orientation appear easily in conjunctions; cf. $^{??}$*stupid and pretty* but *stupid but pretty*. So far, we do not annotate orientation information.

(Shaw and Hatzivassiloglou, 1999)'s work explicitly addresses the problem of the relative ordering of adjectives. In contrast to ours, their approach suggests a pairwise relative ordering of concrete adjectives, not of functional or semantic categories.

7. Conclusions and Future Work

We presented two simple algorithms for the classification of adjectives with respect to a range of functional categories. One of these algorithms, the *coordination context* algorithm, has been discussed in detail. The precision rate achieved by this algorithm is encouraging. It is better for high frequency adjectives than for low frequency adjectives.

Our approach can be considered as a first step into the right direction. In order to achieve better results, we intend to extend our approach along two lines:

- incorporation of additional linguistic clues (e.g., that classifier modifiers do not appear in comparative and superlative forms, that modifiers of the same category can be separated by a comma while those of different categories cannot, etc.);
- combination of our strategies with strategies for the recognition of certain semantic categories (e.g., of city and region names, of human properties, etc.)

The middle-range goal of our project is to compile a lexicon for NLP that contains besides the standard lexical and semantic information functional information.

8. References

R. Dixon. 1982. *Where Have All the Adjectives Gone? and Other Essays in Semantics and Syntax*. Mouton, Berlin/Amsterdam/New York.

R. Dixon. 1991. *A New Approach to English Grammar, On Semantic Principles*. Clarendon Paperbacks, Oxford.

U. Engel. 1988. *Deutsche Grammatik*. Julius Groos Verlag, Heidelberg.

W. Frawley. 1992. *Linguistic Semantics*. Erlbaum, Hillsdale, NJ.

M.A.K. Halliday. 1994. *An Introduction to Functional Grammar*. Edward Arnold, London.

V. Hatzivassiloglou and K.R. McKeown. 1993. Towards the automatic identification of adjectival scales: Clustering adjectives according to meaning. In *Proceedings of the ACL '93*, pages 172–182, Ohio State University.

V. Hatzivassiloglou and K.R. McKeown. 1997. Predicting the semantic orientation of adjectives. In *Proceedings of the ACL '97*, pages 174–181, Madrid.

G. Helbig and J. Buscha. 1999. *Deutsche Grammatik. Ein Handbuch für den Ausländerunterricht*. Langenscheidt Verlag Enzyklopädie, Leipzig.

Stefan Klatt. forthcoming. *Ein Werkzeug zur Annotation von Textkorpora und Informationsextraktion*. Ph.D. thesis, Universität Stuttgart.

R. Quirk, S. Greenbaum, G. Leach, and J. Svartvik. 1985. *A Comprehensive Grammar of the English Language*. Longman, London.

V. Raskin and S. Nirenburg. 1995. Lexical semantics of adjectives. a microtheory of adjectival meaning. Technical Report MCCS-95-287, Computing Research Laboratory, New Mexico State University, Las Cruces, NM.

H. Seiler. 1978. Determination: A functional dimension for interlanguage comparison. In H. Seiler, editor, *Language Universals*. Narr, Tübingen.

J. Shaw and V. Hatzivassiloglou. 1999. Ordering among premodifiers. In *Proceedings of the ACL '99*, pages 135–143, University of Maryland, College Park.

G.H. Tucker. 1995. *The Treatment of Lexis in a Systemic Functional Model of English with Special Reference to Adjectives and their Structure*. Ph.D. thesis, University of Wales College of Cardiff, Cardiff.

Z. Vendler. 1968. *Adjectives and Nominalization*. Mouton, The Hague.

Word-level Alignment for Multilingual Resource Acquisition

Adam Lopez*, Michael Nossal*, Rebecca Hwa*, Philip Resnik*†

*University of Maryland Institute for Advanced Computer Studies
†University of Maryland Department of Linguistics
College Park, MD 20742
{alopez, nossal, hwa, resnik}@umiacs.umd.edu

Abstract

We present a simple, one-pass word alignment algorithm for parallel text. Our algorithm utilizes synchronous parsing and takes advantage of existing syntactic annotations. In our experiments the performance of this model is comparable to more complicated iterative methods. We discuss the challenges and potential benefits of using this model to train syntactic parsers for new languages.

1 Introduction

Word alignment is an exercise commonly assigned to students learning a foreign language. Given a pair of sentences that are translations of each other, the students are asked to draw lines between words that mean the same thing.

In the context of multi-lingual natural language processing, word alignment (more simply, *alignment*) is also a necessary step for many applications. For instance, it is required in the parameter estimation step for training statistical translation models (Al-Onaizan et al., 1999; Brown et al., 1990; Melamed, 2000). Alignments are also useful for foreign language resource acquisition. Yarowsky and Ngai (2001) use an alignment to project part-of-speech (POS) tags from English to Chinese, and use the resulting noisy corpus to train a reliable Chinese POS tagger. Their result suggests that is worthwhile to consider more ambitious endeavors in resource acquisition.

Creating a syntactic treebank (e.g., the Penn Treebank Project (Marcus et al., 1993)) is time-consuming and expensive. As a consequence, state-of-the-art stochastic parsers which rely on such treebanks exist only in languages such as English for which they are available. If syntactic annotation could be projected from English to a language for which no treebank has been developed, then the treebank bottleneck may be overcome (Cabezas et al., 2001).

In principle, the success of treebank acquisition in this manner depends on a few key assumptions. The first assumption is that syntactic relationships in one language can be directly projected to another language using an accurate alignment. This theory is explored in Hwa et al. (2002b). A second assumption is that we have access to a reliable English parser and a word aligner. Although high-quality English parsers are available, high-quality aligners are more difficult to come by. Most alignment research has out of necessity concentrated on unsupervised methods. Even the best results are much worse than alignments created by humans. Therefore, this paper focuses on producing alignments that are tailored to the aims of syntactic projection. In particular, we propose a novel alignment model that, given an English sentence, its dependency parse tree, and its translation, simultaneously generates alignments and a dependency tree for the translation.

Our alignment model aims to improve alignment accuracy while maintaining sensitivity to constraints imposed by the syntactic transfer task. We hypothesize that the incorporation of syntactic knowledge into the alignment model will result in higher quality alignments. Moreover, by generating alignments and parse trees simultaneously, the alignment algorithm avoids irreconcilable errors in the projected trees such as crossing dependencies. Thus, our two objectives complement each other.

To verify these hypotheses, we have performed a suite of experiments, evaluating our algorithm on the quality of the resulting alignments and projected parse trees for English and Chinese sentence pairs. Our initial experiments demonstrate that our approach produces alignments and dependency trees whose quality is comparable to those produced by current state-of-the art systems.

We acknowledge that the strong assumptions we have stated for the success of treebank acquisition do not always hold true (Hwa et al., 2002a; Hwa et al., 2002b). Therefore, it will be necessary to devise a training algorithm that learns syntax even in the face of substantial noise introduced by failures in these assumptions. Although this last point is beyond the scope of this paper, we will allude to potential syntactic transfer approaches that are possible with our system, but infeasible under other approaches.

2 Background

Synchronous parsing appears to be the best model for syntactic projection. Synchronous parsing models the translation process as dual sentence generation in which a word and its translation in the other sentence are generated in lockstep. Translation pairs of both words and phrases are generated in a manner consistent with the syntax of their respective languages, but in a way that expresses the same relationship to the rest of the sentence. Thus, alignment and syntax are produced simultaneously and induce mutual constraints on each other. This model is ideal for the pursuit of our objectives, because it captures our complementary goals in an elegant theoretical framework.

Synchronous parsing requires both parses to adhere to the constraints of a given monolingual parsing model. If we assume context-free grammars, then each parse must be context-free. If we assume dependency grammars, then each parse must observe the planarity and connectivity con-

straints typical of such grammars (e.g. Sleator and Temperley (1993)).

In contrast, many alignment models (Melamed, 2000; Brown et al., 1990) rely on a bag-of-words model. This model presupposes no structural constraints on either input sentence beyond its linear order. To see why this type of model is problematic for syntactic transfer, consider what happens when syntax subsequently interacts with its output. Projecting dependencies across such an alignment may result in a dependency tree that violates planarity and connectivity constraints (Figure 1).

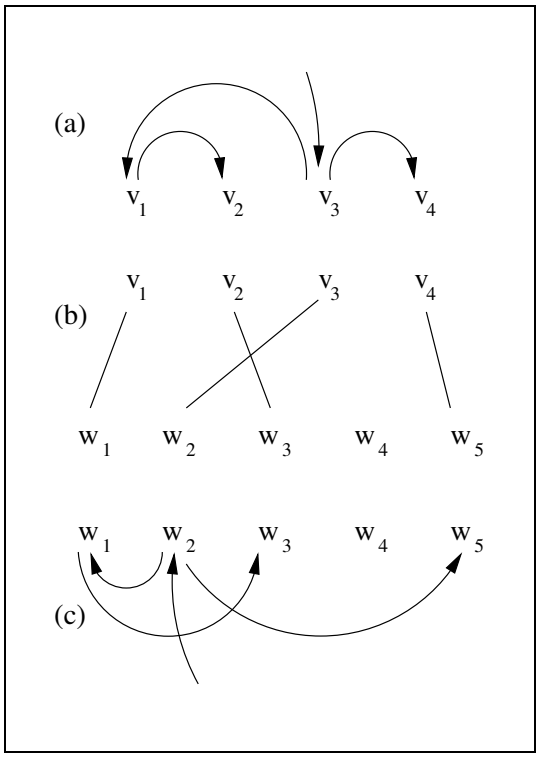


Figure 1: Violation of dependency grammar constraints caused by projecting a dependency parse across a bag-of-words alignment. Combining the syntax of (a) with the alignment of (b) produces the syntax of (c). In this example, the link (w_1, w_3) crosses the link (w_2, w_5) violating the planarity constraint. The word w_4 is unconnected, violating the connectivity constraint.

Once the fundamental assumptions of the syntactic model have been breached, there is no clear way to recover. For this reason, we would prefer not to use bag-of-words alignment models, although in many respects they remain state-of-the-art for alignment.

A canonical example of synchronous parsing is the Stochastic Inversion Transduction Grammar (SITV) (Wu, 1995). The SITV model imposes the constraints of context-free grammars on the synchronous parsing environment. However, we regard context-free grammars as problematic for our task, because recent statistical parsing models (Charniak, 2000; Collins, 1999; Ratnaparkhi, 1999) owe much of their success to ideas inherent to dependency parsing. We therefore adopt an algorithm described in Alshawi and Douglas (2000).[1] Their algorithm constructs synchronous dependency parses in the context of a domain-specific speech-to-speech translation system. In their system, synchronous parsing only enforces a contiguity constraint on phrasal translations. The actual syntax of the sentence is not assumed to be known. Nevertheless, their model is a synchronous parser for dependency syntax, and we adopt it for our purposes.

3 Our Modified Alignment Algorithm

We introduce parse trees as an optional input to the algorithm of Alshawi and Douglas (2000). We require that output dependency trees conform to dependency trees that are provided as input. If no parse tree is provided, our algorithm behaves identically to that of Alshawi and Douglas (2000).

3.1 Definitions

Our input is a parallel corpus that has been segmented into sentence pairs. We represent a sentence pair as the pair of word sequences ($V = v_1...v_m$, $W = w_1...w_n$). The algorithm iterates over the sentence pairs producing alignments.

We define a dependency parse as a rooted tree in which all words of the sentence appear once, and each node in the tree is such a word (Figure 2). An in-order traversal of the tree produces the sentence. A word is said to be *modified* by any words that appear as its children in the tree; conversely, the parent of a word is known as its *headword*. A word is said to *dominate* the span of all words that are descended from it in the tree, and is likewise known as the *headword* of that span.[2] Subject to these constraints, the dependency parse of V is expressed as a function $p_V : \{1...m\} \rightarrow \{0...m\}$ which defines the headword of each word in the dependency graph. The expression $p_V(i) = 0$ indicates that word v_i is the root node of the graph (the headword of the sentence). The dependency parse of W, $p_W : \{1...n\} \rightarrow \{0...n\}$ is defined in the same way.

An alignment is expressed as a function $a : \{1...m\} \rightarrow \{0...n\}$ in which $a(i) = j$ indicates that word v_i of V is aligned with word w_j of W. The case in which $a(i) = 0$ denotes *null alignment* (i.e. the word v_i does not correspond to any word in W). Under the constraints of synchronous parsing, we require that if $a(i) \neq 0$, then $p_W(a(i)) = a(p_V(i))$. In other words, the headword of a word's translation is the translation of the word's headword (Figure 3). We also require that the analogous condition hold for the inverse alignment map $a^{-1} : \{1...n\} \rightarrow \{0...m\}$.

3.2 Algorithm Details

Our algorithm (Appendix) is a bottom-up dynamic programming procedure. It is initialized by considering all

[1] An alternative to dependency grammar is the richer formalism of Synchronized Tree-Adjoining Grammar (TAG) (Shieber and Schabes, 1990). However, Synchronized TAG raises issues of computational complexity and has not yet been exploited in a stochastic setting.

[2] Elsewhere, the terms *connectivity* and *planarity* are used to define these constraints.

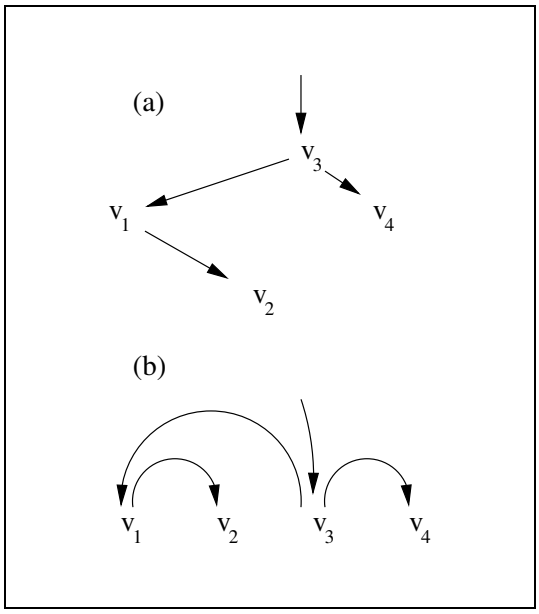


Figure 2: A dependency parse. In (a) the sentence is depicted in a tree form that makes the dominance and headword relationships clear (v_3 is the headword of the sentence). In (b) the same tree is depicted in more familiar sentence form, with the links drawn above the words.

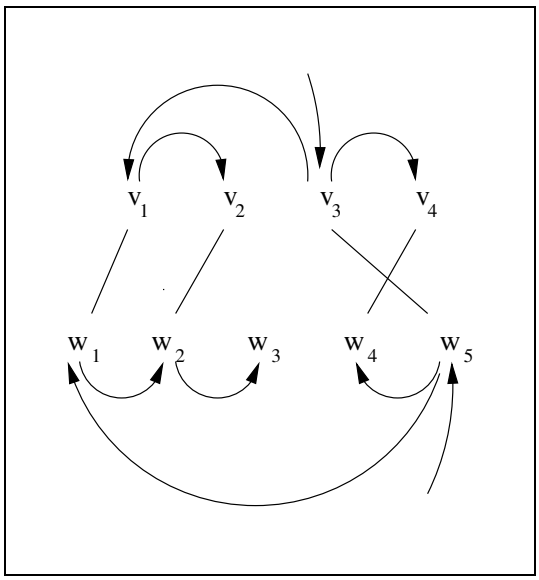


Figure 3: Synchronous dependency parses. Notice that all dependency links are symmetric across the alignment. In addition, the unaligned word w_3 is connected in the parse of W.

possible alignments of one word to another word or to null. Alshawi and Douglas (2000) considered alignments of two words to one or no words, but we found in our evaluations that restricting the initialization step to one word produced better results. In fact, Melamed (2000) argues in favor of exclusively one-to-one alignments. However, we may later explore in more detail the effects of initializing from multiword alignments.

As in Alshawi and Douglas (2000) each possible one-to-one alignment is scored using the ϕ^2 metric (Gale and Church., 1991), which is used to compute the correlation between $v_i \in V$ and $w_j \in W$ over all sentence pairs (V, W) in the corpus. Sentence co-occurrence counts are not the only possible data set with which we can use this metric. Therefore, we denote this type of initialization by ϕ^2_A to distinguish from a case we consider in Section 4.7, in which we use ϕ^2 initialized from counts of Giza++ alignment links. The latter case is denoted by ϕ^2_G.

To compute alignments of larger spans, the algorithm combines adjacent sub-alignments. During this step, one sub-alignment becomes a modifier phrase. Interpreting this in terms of dependency parsing, the aligned headwords of the modifier phrase become modifiers of the aligned headwords of the other phrase. At each step, the score of the alignment is computed. Following Alshawi and Douglas (2000) we simply add the score of the sub-alignments. Thus the overall score of any aligned subphrase can be computed as follows.

$$\sum_{(i,j):a(i)=j} \phi^2(v_i, w_j)$$

The output of the algorithm is simply the highest-scoring alignment that covers the entire span of both V and W.

3.3 Treatment of Null Alignments

Null alignments present a few practical issues. For experiments involving ϕ^2_A, we adopt the practice of counting a null token in the shorter sentence of each pair.[3] An alternative solution to this problem would involve initialization from a word association model that explicitly handles nulls, such as that of Melamed (2000).

An implication of the synchronous parsing constraint given in Section 3.1 is that null aligned words must be leaf words within their respective dependency graphs. In certain cases this may not lead to the best synchronized parse. We remove this condition. Effectively, we consider each sentence to consist of the same number of tokens, some of which may be null tokens. (usually, this will introduce null tokens into only the shorter sentence, but not necessarily). The null tokens behave like words with regards to the synchronous parsing constraint, but they do not impact phrase contiguity.[4] In only the resulting surface dependency graphs, we remove null tokens by contracting all edges between the null token and its parent and naming the resultant node with the word on the parent node. Recall from graph theory that contraction is an operation whereby an edge is removed and the nodes at its endpoints are conflated. [5] Thus, words that modify a null token are interpreted as modifiers of the the null token's headword. This is illustrated in Figure 4. One important implication of this is that we can only allow a null token to be the headword

[3] Srinivas Bangalore, personal communication.

[4] a null token is considered to be contiguous with any other subphrase – another way to view this is that a null token is an unseen word that may appear at any location in the sentence in order to satisfy contiguity constraints.

[5] see e.g., Gross and Yellen (1999)

of the sentence if it has a single modifier. Otherwise, the result of the graph contraction would not be a rooted tree. We found that this treatment of null alignments resulted in a slight improvement in alignment results.

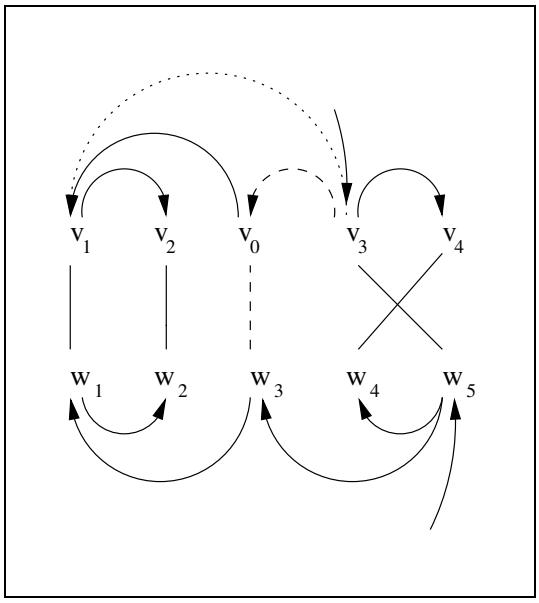


Figure 4: Effect of null words on synchronous parses. In this case, word w_3 has been aligned to the null token v_0. However, v_0 can still dominate other words in the parse of V. Once the structure has been completed, the edge between v_0 and v_3 (indicated by the dashed line) will contract. This will cause the dependency between v_1 and v_0 to become the inferred dependency (indicated by the dotted line) between v_1 and v_3.

3.4 Analysis

In the case that there are no parses available, the computational complexity of the algorithm is $O(m^3n^3)$, but with a parse of V (and an efficient enumeration of the subphrase combinations allowed by the parse) the complexity reduces to $O(m^3n)$. If both parses are available the complexity would be reduced to $O(mn)$.

It is important to note that as it is presented, our algorithm does not search the entire space of possible alignment/tree combinations. Melamed observes that two modifications are required to accomplish this.[6] The first modification entails the addition of four new loop parameters to enumerate the possible headwords of the four monolingual subspans. These additional parameters add a factor of $O(m^2n^2)$. Second, Melamed points out that for a small subset of legal structures, it must be possible to combine subphrases that are not adjacent to one another. The most efficient solution to this problem adds two more parameters, for a total of $O(m^6n^6)$. The best known optimization reduces the total complexity to $O(m^5n^5)$. This is far too complex for a practical implementation, so we chose to use the original $O(m^3n^3)$ algorithm for our evaluations. Thus we recognize that our algorithm does not search the entire space of synchronous parses. It inherently incorporates a greedy heuristic, since for each subphrase, it considers only the most likely headword.

[6] I. Dan Melamed, personal communication.

4 Evaluation

We have performed a suite of experiments to evaluate our alignment algorithm. The qualities of the resulting alignments and dependency parse trees are quantified by comparisons with correct human-annotated parses. We compare the alignment output of our algorithm with that of the basic algorithm described in Alshawi and Douglas (2000) and the well-known IBM statistical model described in Brown et al. (1990) using the freely available implementation (Giza++) described in Al-Onaizan et al. (1999). We also compare the output dependency trees against several baselines and against projected dependency trees created in the manner described in (Hwa et al., 2002a). We found that our model, which combines cross-lingual statistics with syntactic annotation, produces alignments and trees that are are comparable to the best results of other methods.

4.1 Data Set

The language pair we have focused on for this study is English-Chinese. The training corpus consists of around 56,000 sentence pairs from the Hong Kong News parallel corpus. Because the training corpus is solely used for word co-occurrence statistics, no annotation is performed on it.

The development set was constructed by obtaining manual English translations for 47 Chinese sentences of 25 words or less, taken from sections 001-015 of the Chinese Treebank (Xia et al., 2000). A separate test set, consisting of 46 Chinese sentences of 25 words or less, was constructed in a similar fashion.[7] To obtain correct English parses, we used a context-free parser (Collins, 1999) and converted its output to dependency format. To obtain correct Chinese parses, Chinese Treebank trees were converted to dependency format. Both sets of parses were hand-corrected. The correct alignments for the development and test set were created by two native Chinese speakers using annotation software similar to that described in Melamed (1998).

4.2 Metrics for evaluating alignments

As a measure of alignment accuracy, we report Alignment Precision (AP) and Alignment Recall (AR) figures. These are computed by by comparing the alignment links made by the system with the links in the correct alignment. We denote the set of guessed alignment links by G_a and the set of correct alignment links by C_a. Precision is given by $AP = \frac{|C_a \cap G_a|}{|G_a|}$. Recall is given by $AR = \frac{|C_a \cap G_a|}{|C_a|}$. We also compute the F-score (AF), which is given by $AF = \frac{2 \cdot AP \cdot AR}{AP + AR}$. Null alignments are ignored in all computations. Our evaluation metric is similar to that of Och and Ney (2000).

[7] These sentences have already been manually translated into English as part of the NIST MT evaluation preview (See http://www.nist.gov/speech/tests/mt/). The sentences were taken from sections 038, 039, 067, 122, 191, 207, 249.

Synchronous Parsing Method	AP	AR	AF	CTP
sim-Alshawi (ϕ_A^2)	40.6	36.5	38.4	18.5
sim-Alshawi (ϕ_A^2) + English parse	43.8	39.3	41.4	39.9
sim-Alshawi (ϕ_A^2) + English parse + Chinese bigrams	42.9	38.5	40.6	39.4
sim-Alshawi (ϕ_A^2) + both bigrams	41.5	37.3	39.3	16.5
Giza++ initialization (ϕ_G^2)	51.2	45.9	**48.4**	11.6
Giza++ initialization (ϕ_G^2)+ English parse	49.6	44.6	47.0	**44.7**

Baseline Method	AP	AR	AF	CTP
Same Order Alignment	15.7	14.1	14.8	NA
Random Alignment (avg scores)	7.8	7.0	7.4	NA
Forward-chain	NA	NA	NA	37.3
Backward-chain	NA	NA	NA	12.9
Giza++	68.7	40.9	**51.3**	NA
Hwa et al. (2002a)	NA	NA	NA	**44.1**

Table 1: Alignment Results for All Methods.
AP = Alignment Precision. AR = Alignment Recall. AF = Alignment F-Score. CTP = Chinese Tree Precision.
All scores are reported as percentages of 100.
The best scores in each table appear in bold.

4.3 Metrics for evaluating projected parse trees

As a measure of induced dependency tree accuracy, we report unlabeled Chinese Tree Precision (CTP). This is computed by comparing the output dependency tree with the correct dependency trees. We denote the set of guessed dependency links by G_p and the set of correct alignment links by C_p. A small number of words (mostly punctuation) were not linked to any parent word in the correct parse; links containing these words are not included in either C_p or G_p. Precision is given by $CTP = \frac{|C_p \cap G_p|}{|G_p|}$. For dependency trees, $|C_p| = |G_p|$, since each word contributes one link relating it to its headword. Thus, recall is the same as precision for our purposes.

4.4 Baseline Results

We first present the scores of some naïve algorithms as a baseline in order to provide a lower bound for our results. The results of the baseline experiments are included with all other results in Table 1. Our first baseline (Same Order Alignment) simply maps character v_i in the English sentence to character w_i in the Chinese sentence, or w_n in the case of $i > n$. Our second baseline (Random Alignment), randomly aligns word v_i to word w_j subject to the constraint that no words are multiply aligned. We report the average scores over 100 runs of this baseline. The best Random Alignment F-score was 10.0% and the worst was 5.3% with a standard deviation of 0.9%.

For parse trees, we use two simple baselines. In the first (Forward-Chain), each word modifies the word immediately following it, and the last word is the headword of the sentence. For the second baseline (Backward-Chain), each word modifies the word immediately preceding it, and the first word is the headword of the sentence. No alignment was performed for these baselines.

The remaining baselines relate to the Giza++ algorithm. Giza++ produces the best word alignments. For reasons described previously, Giza++ alignments do not combine easily with syntax. However, Hwa et al. (2002a) contains an investigation in which trees output from a projection across Giza++ alignment are modified using several heuristics, and subsequently improved using linguistic knowledge of Chinese. We report the Chinese Tree Precision obtained by this method.

4.5 Synchronous Parsing Results

Our first set of alignments combines the ϕ_A^2 cross-lingual co-occurrence metric described previously with either English parse or no parse trees. In this set, ϕ_A^2 with no parse is nearly identical to the approach described in Alshawi and Douglas (2000) (excepting our treatment of null alignments). Thus, it serves as a useful point of comparison for runs that make use of other information. In Table 1 we refer to it as sim-Alshawi.

What we find is that incorporating parse trees results in a modest improvement over the baseline approach of sim-Alshawi. Why aren't the improvements more substantial? One observation is that using parses in this manner results in only passive interaction with the cross-lingual ϕ_A^2 scores. In other words, the parse filters out certain alignments, but cannot in any other way counteract the biases inherent in the word statistics. Nevertheless, it appears to be modest progress.

4.6 Results of Using Bigrams to Approximate Parses

The results suggest that using parses to constrain the alignment is helpful. It is possible that using both parses would result in a more substantial improvement. However, we have already stated that we are interested in the case of asynchronous resources. Under this scenario, we only have access to one parse. Is there some way that we can approximate syntactic constraints of a sentence without having access to its parse?

The parsers of (Charniak, 2000; Collins, 1999; Ratnaparkhi, 1999) make substantial use of *bilexical dependencies*. Bilexical dependencies capture the idea that linked words in a dependency parse have a statistical affinity for each other: they often appear together in certain contexts. We suspect that bigram statistics could be used as a proxy for actual bilexical dependencies.

We constructed a simple test of this theory: for each English sentence $V = v_1...v_m$ in the development set with parse $p_V : \{1...m\} \rightarrow \{0...m\}$, we first construct the set of all bigrams $B = \{(v_i, v_j) : 1 \leq i < j \leq m\}$. We then partitioned B into two sets: bigrams of linked words, i.e. $L = \{(v_i, v_j) : (v_i, v_j) \in B; p_V(v_i) = v_j \text{ or } p_V(v_j) = v_i\}$ and unlinked words $U = B - L$. We used the Bigram Statistics Package (Pedersen, 2001), to collect bigram statistics over the entire dev/train corpus and compute the average statistical correlation of each set using a variety of metrics (loglikelihood, dice, χ^2, ϕ^2). The results indicated that bigrams in the linked set L were more correlated than those in the unlinked set U under all metrics. We repeated this experiment with the development sentences in Chinese, with similar results. Although this is by no means a conclusive experiment, we took the results as an indication that using bigram statistics as an approximation of a parse might be helpful where no parse was actually available.

To incorporate bigram statistics into our alignment model, we modified the scoring function in the following manner: each time a dependency link is introduced between words and we do not have access to the source parse, we add into the alignment score the bigram score of the two words. The bigram score is based on the ϕ^2 metric computed for bigram correlation. We call this ϕ^2_B. The resulting alignment score can now be given by the following formula.

$$\sum_{(i,j):a(i)=j} \phi^2_A(v_i, w_j) + \sum_{(i,j):i<j, p_W(i)=j \wedge p_W(j)=i} \phi^2_B(w_i, w_j)$$

Our results indicate that using Chinese bigram statistics in conjunction with English parse trees in this manner results in a small decrease in the score along all measures. Nonetheless, there is an intuitively appealing interpretation of using bigrams in this way. The first is that the modification of the scoring function provides competitive interaction between parse information and cross-lingual statistics. The second is that if bigram statistics represent a weak approximation of syntax, then perhaps the iterative refinement of this statistic (e.g. by taking counts only over words that were linked in a previous iteration) would satisfy our objective of syntactic transfer.

4.7 Results of Using Better Word Statistics

Our results show that using parse information and coarse cross-lingual word statistics provides a modest boost over an approach using only the cross-lingual word statistics. We also decided to investigate what happens when we seed our algorithm with better cross-lingual statistics

To test this, we initialize our co-occurrence counts from alignment links output by the Giza++ alignment of our corpus. We still use ϕ^2 to compute the correlation. We call this ϕ^2_G. Predictably, using the better word correlation statistics improves the quality of the alignment output in all cases. In this scenario, adding parse information does not seem to improve the alignment score. However, parse trees induced in this manner achieve a higher precision than any of the other methods. It outscores the baseline algorithms by a significant amount, and produces results comparable to the baseline of Hwa et al. (2002a). It is important to note, however, that the baseline of Hwa et al. (2002a) is achieved only after the application of numerous linguistic rules to the output of the Giza++ alignment. Additionally, the trees themselves may contain errors of the type described in Section 2. In contrast, our tree precision results directly from the application of our synchronous parsing algorithm, and all of the output trees are valid dependency parses.

5 Future Work

We believe that a fundamental advantage of our baseline model is its simplicity. Improving upon it will be considerably easier than improving upon a complex model such as the one described in Brown et al. (1990). Improvements may proceed along several possible paths. One path would involve reformulating the scoring functions in terms of statistical models (e.g. generative models). A natural complement to this path would be the introduction of iteration with the goal of improving the alignments and the accompanying models. In this approach, we could attempt to learn a coarse statistical model of the syntax of the low-density language after each iteration of the alignment. This information could in turn be used as evidence in the next iteration of the alignment model, hopefully improving its performance. Our results have already established a set of statistics that could be used in the initial iteration of such a task. The iterative approach resonates with an idea proposed in Yarowsky and Ngai (2001), regarding the use of learned part-of-speech taggers in subsequent alignment iterations.

An orthogonal approach would be the application of additional linguistic information. Our results indicated that syntactic knowledge can help improve alignment. Additional linguistic knowledge obtained from named-entity analyses, phrasal boundary detection, and part-of-speech tags might also improve alignment.

Although our output dependency trees represent definite progress, trees with such low precision cannot be used directly to train statistical parsers that assume correct training data (Charniak, 2000; Collins, 1999; Ratnaparkhi, 1999). There are two possible methods of improving upon the precision of this training data. The first is the use of noise-resistant training algorithms such as those described in (Yarowsky and Ngai, 2001). The second is the possibility of improving the precision yield by removing obviously bad training examples from the set. Unlike the baseline model, our word alignment model provides an obvious means of doing this. One possibility is to use a score gleaned from the alignment algorithm as a means of ranking dependency links, and removing links whose score is above some threshold. We hope that a dual approach of improving the precision of the training examples, while simultaneously reducing the sensitivity of the training algorithm, will result in the ability to train a reasonably accurate statistical parser for the new language. Our eventual objective

is to train a parser in this manner.

6 Related work

Al-Onaizan et al. (1999), Brown et al. (1990) and Melamed (2000) focus on the description of statistical translation models based on the bag-of-words model. Alignment plays a crucial part in the parameter estimation methods of these models, but they remain problematic for syntactic transfer for reasons described in Section 2. The work of Hwa et al. (2002b) is an investigation into the combination of syntax with the output of this type of model. Och et al. (1999) presents a statistical translation model that performs phrasal translation, but it relies on shallow phrases that are discovered statistically, and makes no use of syntax. Yamada and Knight (2001) create a full-fledged syntax-based translation model. However, their model is unidirectional; it only describes the syntax of one sentence, and makes no provision for the syntax of the other. Wu (1995) presents a complete theory of synchronous parsing using a variant of context-free grammars, and exhibits several positive results, though not for syntax transfer. Alshawi and Douglas (2000) present the synchronous parsing algorithm on which our work is based. Much like the work on translation models, however, this work is interested in alignment primarily as a mechanism for training a machine translation system. Variations on the synchronous parsing algorithm appear in Alshawi et al. (2000a) and Alshawi et al. (2000b), but the algorithm of Alshawi and Douglas (2000) appears to be the most complete.

7 Conclusion

We have described a new approach to alignment that incorporates dependency parses into a synchronous parsing model. Our results indicate that this approach results in alignments whose quality is comparable to those produced by complicated iterative techniques. In addition, our approach demonstrates substantial promise in the task of learning syntactic models for resource-poor languages.

8 Acknowledgements

This work has been supported, in part, by ONR MURI Contract FCPO.810548265, DARPA/ITO Cooperative Agreement N660010028910, NSA Contract RD-02-5700 and Mitre Contract 010418-7712. The authors would like to thank I. Dan Melamed and Srinivas Bangalore for helpful discussions; Franz Josef Och for help with Giza++; and Lingling Zhang, Edward Hung, and Gina Levow for creating the gold standard annotations for the development and test data.

9 References

Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, I. Dan Melamed, Franz Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation: Final report. In *Summer Workshop on Language Engineering*. John Hopkins University Center for Language and Speech Processing.

Hiyan Alshawi and Shona Douglas. 2000. Learning dependency transduction models from unannotated examples. *Philosophical Transactions of the Royal Society*, 358:1357–1372.

Hiyan Alshawi, Srinivas Bangalore, and Shona Douglas. 2000a. Learning dependency translation models as collections of finite state head transducers. *Computational Linguistics*, 26:1357–1372.

Hiyan Alshawi, Srinivasa Bangalore, and Shona Douglas. 2000b. Head transducer models for speech translation and their automatic acquisition from bilingual data. *Machine Translation*, 15:105–124.

Peter F. Brown, John Cocke, Stephen Della Pietra, Vincent J. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85.

Clara Cabezas, Bonnie Dorr, and Philip Resnik. 2001. Spanish language processing at university of maryland: Building infrastructure for multilingual applications. In *Proceedings of the Second International Workshop on Spanish Language Processing and Language Technologies (SLPLT-2)*.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Meeting of the North American Chapter of the Association for Computational Linguistics*.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

William A. Gale and Kenneth W. Church. 1991. Identifying word correspondences in parallel texts. In *Proceedings of the Fourth DARPA Speech and Natural Language Processing Workshop*, pages 152–157.

Jonathan Gross and Jay Yellen, 1999. *Graph Theory and Its Applications*, chapter 7.5: Transforming a Graph by Edge Contraction, pages 263–266. Series on Discrete Mathematics and Its Applications. CRC Press.

Rebecca Hwa, Philip Resnik, and Amy Weinberg. 2002a. Breaking the resource bottleneck for multilingual parsing. In *Proceedings of the Workshop on Linguistic Knowledge Acquisition and Representation: Bootstrapping Annotated Language Data*. To appear.

Rebecca Hwa, Philip Resnik, Amy Weinberg, and Okan Kolak. 2002b. Evaluating translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting of the ACL*. To appear.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

I. Dan Melamed. 1998. Annotation style guide for the blinker project. Technical Report IRCS 98-06, University of Pennsylvania.

I. Dan Melamed. 2000. Models of translational equivalence among words. *Computational Linguistics*, 26(2):221–249, Jun.

Franz Josef Och and Hermann Ney. 2000. Improved statis-

tical alignment models. In *Proceedings of the 38th Annual Meeting of the ACL*, pages 440–447.

Franz Josef Och, Christoph Tillmann, and Hermann Ney. 1999. Improved alignment models for statistical machine translation. In *Proceedings of the Joint Conference of Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28, Jun.

Ted Pedersen. 2001. A decision tree of bigrams is an accurate predictor of word sense. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, pages 79–86, Jun.

Adwait Ratnaparkhi. 1999. Learning to parse natural language with maximum entropy models. *Machine Learning*, 34(1-3):151–175.

Stuart Shieber and Yves Schabes. 1990. Synchronous tree-adjoining grammars. In *Proceedings of the 13th International Conference on Computational Linguistics*, volume 3, pages 1–6.

Daniel Sleator and Davy Temperley. 1993. Parsing english with a link grammar. In *Third International Workshop on Parsing Technologies*, Aug.

Dekai Wu. 1995. Stochastic inversion transduction grammars, with application to segmentation, bracketing, and alignment of parallel corpora. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence*, pages 1328–1335, Aug.

Fei Xia, Martha Palmer, Nianwen Xue, Mary Ellen Ocurowski, John Kovarik, Fu-Dong Chiou, Shizhe Huang, Tony Kroch, and Mitch Marcus. 2000. Developing guidelines and ensuring consistency for chinese text annotation. In *Proceedings of the Second Language Resources and Evaluation Conference*, June.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In *Proceedings of the Conference of the Association for Computational Linguistics*.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual pos taggers and np bracketers via robust projection across aligned corpora. In *Proceedings of the 2nd Meeting of the North American Chapter of the Association for Computational Linguistics*, Jun.

A Algorithm Pseudocode

The following code does not address what constitutes a legal combination of subspans for an alignment. Legal subspans depend on constraints imposed by an input parse, if available. Otherwise, as in Alshawi and Douglas (2000), all possible combinations of subspans are legal. Regardless of what constitutes a legal subspan, the enumeration of spans must be done in a reasonable way. Small spans must be enumerated before larger spans that are constructed from them.

The variables i_V and j_V denote the span $v_{i_V+1}...v_{j_V}$, and p_V denotes a partition of the span such that $i_V \leq p_V \leq j_V$. The variables i_W, j_W, and p_W are defined analogously on W.

Our data structure is a chart α, which contains cells indexed by i_V, j_V, i_W, and j_W. Each cell contains subfields $phrase$, $modifierPhrase$, and $score$.

Finally, we assume the existence of functions assocScore and score. The assocScore function computes the score of directly aligning to short spans of the sentence pair. In this paper, we use variations on the ϕ^2 metric (Gale and Church., 1991) for this. The score function computes the score of combining two sub-alignments, assuming that the second sub-alignment becomes a modifier of the first. In this paper, we use one score function that simply adds the score of sub-alignments, and one that adds bigram correlation to the score of the sub-alignments. In principle, arbitrary scoring functions can be used.

initialize the chart

for all legal combinations of i_V, j_V,i_W, and j_W

$\alpha(i_V, j_V, i_W, j_W) = assocScore(v_{i_V+1}...v_{j_V}, w_{i_W+1}...w_{j_W})$

complete the chart

for all legal combinations of i_V, j_V, p_V, i_W, j_W, and p_W

consider the case in which aligned subphrases are in the same order in both languages.

$phrase = \alpha(i_V, p_V, i_W, p_W)$

$modifierPhrase = \alpha(p_V, j_V, p_W, j_W)$

$score =$score($phrase, modifierPhrase$)

if $score > \alpha(i_V, j_V, i_W, j_W).score$ then

$\alpha(i_V, j_V, i_W, j_W) =$ new subAlignment($phrase, modifierPhrase, score$)

consider the case in which the dominance relationship between these two phrases is reversed.

swap($phrase, modifierPhrase$)

$score =$score($phrase, modifierPhrase$)

if $score > \alpha(i_V, j_V, i_W, j_W).score$ then

$\alpha(i_V, j_V, i_W, j_W) =$ new subAlignment($phrase, modifierPhrase, score$)

consider the case in which aligned subphrases are in the reverse order in each language.

$phrase = \alpha(i_V, p_V, p_W, j_W)$

$modifierPhrase = \alpha(p_V, j_V, i_W, p_W)$

$cost =$cost($phrase, modifierPhrase$)

$score =$score($phrase, modifierPhrase$)

if $score > \alpha(i_V, j_V, i_W, j_W).score$ then

$\alpha(i_V, j_V, i_W, j_W) =$ new subAlignment($phrase, modifierPhrase, score$)

consider the case in which the dominance relationship between these two phrases is reversed.

swap($phrase, modifierPhrase$)

$score =$score($phrase, modifierPhrase$)

if $score > \alpha(i_V, j_V, i_W, j_W).score$ then

$\alpha(i_V, j_V, i_W, j_W) =$ new subAlignment($phrase, modifierPhrase, score$)

return $\alpha(0, m, 0, n)$

Generating A Parsing Lexicon from an LCS-Based Lexicon

Necip Fazıl Ayan and Bonnie J. Dorr

Department of Computer Science
University of Maryland
College Park, 20742, USA
{nfa, bonnie}@umiacs.umd.edu

Abstract

This paper describes a technique for generating parsing lexicons for a principle-based parser (Minipar). Our approach maps lexical entries in a large LCS-based repository of semantically classified verbs to their corresponding syntactic patterns. A by-product of this mapping is a lexicon that is directly usable in the Minipar system. We evaluate the accuracy and coverage of this lexicon using LDOCE syntactic codes as a gold standard. We show that this lexicon is comparable to the hand-generated Minipar lexicon (i.e., similar recall and precision values). In a later experiment, we automate the process of mapping between the LCS-based repository and syntactic patterns. The advantage of automating the process is that the same technique can be applied directly to lexicons we have for other languages, for example, Arabic, Chinese, and Spanish.

1. Introduction

This paper describes a technique for generating parsing lexicons for a principle-based parser (Minipar (Lin, 1993; Lin, 1998)) using a lexicon that is semantically organized according to Lexical-Conceptual Structure (LCS) (Dorr, 1993; Dorr, 2001)—an extended version of the verb classification system proposed by (Levin, 1993).[1] We aim to determine how much syntactic information we can obtain from this resource, which extends Levin's original classification as follows: (1) it contains 50% more verbs and twice as many verb entries (Dorr, 1997)—including new classes to accommodate previously unhandled verbs and phenomena (e.g., clausal complements); (2) it incorporates theta-roles which, in turn, are associated with a thematic hierarchy for generation (Habash and Dorr, 2001); and (3) it provides a higher degree of granularity, i.e., verb classes are sub-divided according to their aspectual characteristics (Olsen et al., 1997).

More specifically, we provide a general technique for projecting this broader-scale semantic (language-independent) lexicon onto syntactic entries, with the ultimate objective of testing the effects of such a lexicon on parser performance. Each verb in our semantic lexicon is associated with a class, an LCS representation, and a thematic grid.[2] These are mapped systematically into syntactic representations. A by-product of this mapping is a lexicon that is directly usable in the Minipar system.

Several recent lexical-acquisition approaches have produced new resources that are ultimately useful for syntactic analysis. The approach that is most relevant to ours is that of (Stevenson and Merlo, 2002b; Stevenson and Merlo, 2002a), which involves the derivation of verb classes from syntactic features in corpora. Because their approach is unsupervised, it provides the basis for automatic verb classification for languages not yet seen. This work is instrumental in providing the basis for wide-spread applicability of our technique (mapping verb classes to a syntactic parsing lexicon), as verb classifications become increasingly available for new languages over the next several years.

An earlier approach to lexical acquisition is that of (Grishman et al., 1994), an effort resulting in a large resource called Comlex—a repository containing 38K English headwords associated with detailed syntactic patterns. Other researchers (Briscoe and Carroll, 1997; Manning, 1993) have also produce subcategorization patterns from corpora. In each of these cases, data collection is achieved by means of statistical ex-

[1] We focus only on verb entries as they are cross-linguistically the most highly correlated with lexical-semantic divergences.

[2] Although Lexical Conceptual Structure (LCS) is the primary semantic representation used in our verb lexicon, it is not described in detail here (but see (Dorr, 1993; Dorr, 2001). For the purpose of this paper, we rely primarily on the thematic grid representation, which is derived from the LCS. Still we refer to the lexicon as "LCS-based" as we store all of these components together in one large repository: http://www.umiacs.umd.edu/~bonnie/LCS_Database_Documentation.html.

traction from corpora; there is no semantic basis and neither is intended to be used for multiple languages.

The approaches of (Carroll and Grover, 1989) and (Egedi and Martin, 1994) involve acquisition English lexicons from entries in LDOCE and *Oxford Advanced Learner's Dictionary* (OALD), respectively. The work of (Brent, 1993) produces a lexicon from a grammar—the reverse of what we aim to do. All of these approaches are specific to English. By contrast, our goal is to have a unified repository that is transferable to other languages—and from which our parsing (and ultimately generation) grammars may be derived.

For evaluation purposes, we developed a mapping from the codes of Longman's Dictionary of Contemporary English (LDOCE (Procter, 1978))—the most comprehensive online dictionary for syntactic categorization—to a set of syntactic patterns. We use these patterns as our gold standard and show that our derived lexicon is comparable to the hand-generated Minipar lexicon (i.e., similar recall and precision values). In a later experiment, we automate the process of mapping between the LCS-based repository and syntactic patterns—with the goal of portability: We currently have LCS lexicons for English, Arabic, Spanish, and Chinese, so our automated approach allows us to produce syntactic lexicons for parsing in each of these languages.

Section 2. presents a brief description of each code set we use in our experiments. In Section 3., we explain how we generated syntactic patterns from three different lexicons. In Section 4., we discuss our experiments and the results. Section 5. describes ongoing work on automating the mapping between LCS-based representations and syntactic patterns. Finally, we discuss our results and some possible future directions.

2. Code Descriptions

In many online dictionaries, verbs are classified according to the arguments and modifiers that can follow them. Most dictionaries use specific codes to identify transitivity, intransitivity, and ditransitivity. These broad categories may be further refined, e.g., to distinguish verbs with NP arguments from those with clausal arguments. The degree of refinement varies widely.

In the following subsections, we will present three different code sets. As shown in Figure 1, the first of these (OALD) serves as a mediating representation in the mapping between Minipar codes and syntactic patterns. The LCS lexicon and LDOCE codes are mapped directly into syntactic patterns, without an intervening representation. The patterns resulting from the LDOCE are taken as the gold standard, serving as the basis of comparison between the Minipar- and LCS-based lexicons.

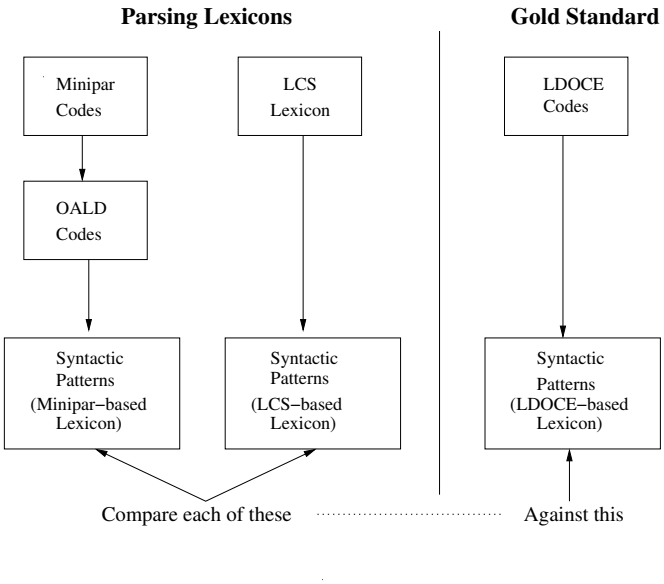


Figure 1: A Comparison between Minipar- and LCS-based Lexicons using LDOCE as the Gold Standard

2.1. OALD Codes

This code set is used in Oxford Advanced Learner's Dictionary, a.k.a OALD (Mitten, 1992). The verbs are categorized into 5 main groups: Intransitive verbs, transitive verbs, ditransitive verbs, complex transitive verbs, and linking verbs. Each code is of the form $Sa_1[.a_2]$ where S is the first letter of the verb categorization ($S \in \{I, T, D, C, L\}$ for the corresponding groups), and $a_1, a_2, \ldots$ are the argument types. If a code contains more than one argument, each argument is listed serially. Possible argument types are n for nouns, f for finite clauses (*that* clauses), g for "-ing" clauses, t for infinitive clauses, w for finite clauses beginning with "-wh", i for bare infinitive clauses, a for adjective phrases, p for prepositions and pr for prepositional phrases.

For example, Tn refers to the verbs followed by a noun ('She read the book'), $Tn.pr$ refers to the verbs followed by a noun and a prepositional phrase ('He opened the door with a latch'), and $Dn.n$ refers to the verbs followed by two nouns ('She taught the children French'). The number of codes in OALD code set is 32 and the codes are listed in Table 1.

OALD codes are simplistic in that they do not include modifiers. In addition, they also do not explicitly specify which prepositions can be used in the PPs.

2.2. Minipar Codes

The Minipar coding scheme is an adaptation of the OALD codes. Minipar extends OALD codes by pro-

Categorization	OALD Codes
Intransitive verbs	{I, Ip, Ipr, In/pr, It}
Transitive verbs	{Tn, Tn.pr, Tn.p, Tf, Tw, Tt, Tg, Tn.t, Tn.g, Tn.i}
Complex Transitive verbs	{Cn.a, Cn.n, Cn.n/a, Cn.t, Cn.g, Cn.i}
Ditransitive verbs	{Dn.n, Dn.pr, Dn.f, Dn.t, Dn.w, Dpr.f, Dpr.w, Dpr.t}
Linking verbs	{La, Ln}

Table 1: OALD Code Set: The Basis of Minipar Codes

viding a facility for specifying prepositions, but only 8 verbs are encoded with these prepositional codes in the official Minipar distribution. In these cases, the codes containing pr are refined to be $pr.prep$, where $prep$ is the head of the PP argument.[3] In addition, Minipar codes are refined in the following ways:

1. Optional arguments are allowed, e.g., $T[n].pr$ describes verbs followed by an optional noun and a PP. This is equivalent to the combination of the OALD codes $Tn.pr$ and Ipr.

2. Two or more codes may be combined, e.g., *Tfgt* describes verbs followed by a clause that is finite, infinitive, or gerundive ("-ing").

3. Prepositions may be specified in prepositional phrases. Some of the codes containing pr as an argument are converted into $pr.prep$ in order to declare that the prepositional phrase can begin with only the specified preposition $prep$.

The set of Minipar codes contain 66 items. We will not list them here since they are very similar to the ones in Table 1, with the modifications described above.

2.3. LDOCE Codes

LDOCE has a more detailed code set than that of OALD (and hence Minipar). The codes include both arguments and modifiers. Moreover, prepositions are richly specified throughout the lexicon. The syntax of the codes is either *CN* or *CN-Prep*, where *C* corresponds to the verb sub-categorization (as in the generic OALD codes) and N is a number, which corresponds to different sets of arguments that can follow the verb. For example, T1-ON refers to verbs that are followed by a noun and a PP with the head *on*. The number of codes included in this set is 179. The meaning of each is described in Table 2.

Number	Arguments
1	one or more nouns
2	bare infinitive clause
3	infinitive clause
4	-ing form
5	-that clause
6	clauses with a wh- word
7	adjective
8	past participle
9	descriptive word or phrase

Table 2: LDOCE Number Description

3. Our Approach

Our goal is evaluate the accuracy and coverage of a parsing lexicon where each verb is classified according to the arguments it takes. We use syntactic patterns as the basis of the comparison between our parsing lexicon and the original lexicon used in Minipar.

Syntactic patterns simply list the type of the arguments one by one, including the subject. Formally, a syntactic pattern is $a_1, a_2, \ldots$ where a_i is an element of NP, AP, PP, FIN, INF, BARE, ING, WH, PREP, corresponding to noun phrases, adjective phrases, prepositional phrases, clauses beginning with "that", infinitive clauses, bare infinitive clauses, "-ing" clauses, "-wh" clauses and prepositions, respectively. Prepositional phrases may be made more specific by including the heads, which is done by PP.$prep$ where $prep$ is the head of the prepositional phrase. The first item in the syntactic pattern gives the type of the subject.

Our initial attempts at comparing the Minipar- and LCS-based lexicons involved the use of the OALD code set instead of syntactic patterns. This approach has two problems, which are closely related. First, using the class number and thematic grids as the basis of mapping from the LCS lexicon to OALD codes is a difficult task because of the high degree of ambiguity. For example, it is hard to choose among four OALD codes (Ln, La, Tn or Ia) for the thematic grid _th_pred, regardless of the Levin class. In general, the grid-to-OALD mapping is so ambiguous that maintaining consistency over the whole LCS lexicon is virtually impossible.

Secondly, even if we are able to find the correct OALD codes, it is not worth the effort because all that is needed for the parsing lexicon is the type and number of arguments that can follow the verb. For example, $Cn.n$ (as in *"appoint him king"*) and $Dn.n$ (as in *"give him a book"*) both correspond to two

[3] This extension is used only for the preposition *as* for the verbs *absolve, accept, acclaim, brand, designate, disguise, fancy,* and *reckon*.

NPs, but the second NP is a direct object in the former case and an indirect object in the latter. Since the parser relies ultimately on syntactic patterns, not codes, we can eliminate this redundancy by mapping any verb in either of these two categories directly into the [NP.NP.NP] pattern. Thus, using syntactic patterns is sufficient for our purposes.

Our experiments revealed additional flexibility in using syntactic patterns. Unlike the OALD codes (which contain at most two arguments or modifiers), the thematic grids consist of up to 4 modifiers. Mapping onto syntactic patterns instead of onto OALD codes allows us to use all arguments in the thematic grids. For example, [NP.NP.PP.from.PP.to] is an example of transitive verb with two prepositional phrases, one beginning with *from* and the other beginning with *to*, as in *"She drove the kids from home to school."*

In the following subsections, we will examine the mapping into these syntactic patterns from: (1) the LCS lexicon; (2) the Minipar codes; and (3) the LDOCE codes.

3.1. Mapping from the LCS Lexicon to Syntactic Patterns

The LCS lexicon consists of verbs grouped into classes based on an adapted version of verb classes (Levin, 1993) along with the thematic grid representations (see (Dorr, 1993; Dorr, 2001)). We automatically assigned syntactic patterns for each verb in the LCS lexicon using its semantic class number and thematic grid. The syntactic patterns we used in our mapping specify prepositions for entries that require them. For example, the grid _ag_th_instr(with) is mapped onto [NP.NP.PP.with] instead of a generic pattern [NP.NP.PP].

More generally, thematic grids contain a list of arguments and modifiers, and they can be obligatory (indicated by an underscore before the role) or optional(indicated by a comma before the role). The arguments can be one of AG, EXP, TH, SRC, GOAL, INFO, PERC, PRED, LOC, POSS, TIME, and PROP. The logical modifiers can be one of MOD-POSS, BEN, INSTR, PURP, MOD-LOC, MANNER, MOD-PRED, MOD-PERC and MOD-PROP. If the argument or the modifier is followed by parenthesis, the corresponding element is a prepositional phrase and its head must be the one specified between the parentheses (if there is nothing between parentheses, PP can begin with any preposition).

Our purpose is to find the set of syntactic patterns for each verb in LCS lexicon using its Levin class and thematic grid. Since each verb can be in many classes and we aim at assigning syntactic patterns based on the semantic classes and thematic grids, there are three possible mapping methodologies:

1. Assign one or more patterns to each class.
2. Assign one or more patterns to each thematic grid.
3. Assign one or more patterns to each pair of class and thematic grid.

The first methodology fails for some classes because the distribution of syntactic patterns over a specific class is not uniform. In other words, attempting to assign only a set of patterns to each class introduces errors because some classes are associated with more than one syntactic frame. For example, class 51.1.d includes three thematic grids: (1) _th,src; (2) _th,src(from); and (3) _th,src(),goal(). We can either assign all patterns for all of these thematic grids to this class or we can choose the most common one. However, both of these approaches introduce errors: The first will generate redundant patterns and the second will assign incorrect patterns to some verbs. (This occurs because, within a class, thematic grids may vary with respect to their optional arguments or the prepositional head associated with arguments or modifiers.)

The second methodology also fails to provide an appropriate mapping. The problem is that some thematic grids correspond to different syntactic patterns in different classes. For example, the thematic grid _th_prop corresponds to 3 different syntactic patterns: (1) [NP.NP] in class 024 and 55.2.a; (2) [NP.ING] in classes 066, 52.b, and 55.2.b; and (3) [NP.INF] in class 005. Although the thematic grid is the same in all of these classes, the syntactic patterns are different.

The final methodology circumvents the two issues presented above (i.e., more than one grid per class and more than one syntactic frame per thematic grid) as follows: If a thematic grid contains an optional argument, we create two mappings for that grid, one in which the optional argument is treated as if it were not there and one in which the argument is obligatory. For example, _ag_th,goal() is mapped onto two patterns [NP.NP] and [NP.NP.PP]. If the number of optional arguments is X, then the maximum number of syntactic patterns for that grid is 2^X (or perhaps smaller than 2^X since some of the patterns may be identical).

Using this methodology, we found the correct mapping for each class and thematic grid pair by examining the verbs in that class and considering all possible syntactic patterns for that pair. This is a many-to-many mapping, i.e. one pattern can be used for different

OALD Code	Syntactic Patterns
I	[NP]
Tn	[NP.NP]
T[n].pr	[NP.NP] and [NP.NP.PP]
Cn.a	[NP.NP.AP]
Cn.n	[NP.NP.NP]
Cn.n/a	[NP.NP.PP.as]
Cn.i	[NP.NP.BARE]
Dn.n	[NP.NP.NP]

Table 3: Mapping From OALD to Syntactic Patterns

LDOCE Code	Syntactic Patterns
I-ABOUT	[NP.PP.about]
I2	[NP.BARE]
L9-WITH	[NP.PP.with]
T1	[NP.NP]
T5	[NP.FIN]
D1	[NP.NP.NP]
D3	[NP.NP.INF]
V4	[NP.NP.ING]

Table 4: Mapping From LDOCE to Syntactic Patterns

pairs and each pair may be associated with more than one pattern. Each verb in each class is assigned the corresponding syntactic patterns according to its thematic grid. Finally, for each verb, we combined all patterns in all classes containing this particular verb in order to generate the lexicon. We will refer to the resulting lexicon as the LCS-based lexicon in Section 4..

3.2. Mapping from Minipar Codes To Syntactic Patterns

Minipar codes are converted straightforwardly into syntactic patterns using the code specification in (Mitten, 1992). An excerpt of the mapping is given in Table 3. This mapping is one-to-many as exemplified by the code $T[n].pr$. Moreover, the set of syntactic patterns extracted from Minipar does not include some patterns such as [NP.PP] (and related patterns) because Minipar does not include modifiers in its code set.

As a result of this mapping, we produced a new lexicon from Minipar entries, where each verb is listed along with the set of syntactic patterns. We will refer to this lexicon as the Minipar-based lexicon in Section 4..

3.3. Mapping from LDOCE Codes to Syntactic Patterns

Similar to the mapping from Minipar to the syntactic patterns, we converted LDOCE codes to syntactic patterns using the code specification in (Procter, 1978). An excerpt of the mapping is given in Table 4.

Each LDOCE code was mapped manually to one or more patterns. LDOCE codes are more refined than the generic OALD codes, but mapping each to syntactic patterns provides an equivalent mediating representation for comparison. For example, LDOCE codes D1-AT and T1-AT are mapped onto [NP.NP.PP.at] by our mapping technique. Again, this is a many-to-many mapping but only a small set of LDOCE codes map to more than one syntactic pattern.

As a result of this mapping, we produced a new lexicon from LDOCE entries, similar to Minipar lexicon. We will refer to this lexicon as the LDOCE-based lexicon in Section 4..

4. Experiments and Results

To measure the effectiveness of our mapping from LCS entries to syntactic patterns, we compared the precision and recall our derived LCS-based syntactic patterns with the precision and recall of Minipar-based syntactic patterns, using LDOCE-based syntactic patterns as our "gold standard".

Each of the three lexicons contains verbs along with their associated syntactic patterns. For experimental purposes, we convert these into pairs. Formally, if a verb v is listed with the patterns $p_1, p_2, \ldots$, we create pairs (v, p_1), (v, p_2) and so on. In addition, we have made the following adjustments to the lexicons, where L is the lexicon under consideration (Minipar or LCS):

1. Given that the number of verbs in each of the two lexicons is different and that neither one completely covers the other, we take only those verbs that occur in both L and LDOCE, for each L, while measuring precision and recall.

2. In the LDOCE- and Minipar-based lexicons, the number of arguments is never greater than 2. Thus, for a fair comparison, we converted the LCS-based lexicon into the same format. For this purpose, we simply omit the arguments after the second one if the pattern contains more than two arguments/modifiers.

3. The prepositions are not specified in Minipar-based lexicon. Thus, we ignore the heads of the prepositions in LCS-based lexicon, i.e., if the pattern includes [PP.$prep$] we take it as a [PP].

Precision and recall are based on the following inputs:

A = Number of pairs in L occurring in LDOCE
B = Number of pairs in L NOT occurring in LDOCE
C = Number of pairs in LDOCE NOT occurring in L

That is, given a syntactic pattern encoded lexicon L, we compute:

(1) The *precision* of $L = \frac{A}{A+B}$;
(2) The *recall* of $L = \frac{A}{A+C}$.

Verbs in LDOCE Lexicon	5648
Verbs in LCS Lexicon	4267
Common verbs in LCS and LDOCE	3757
Pairs in LCS Lexicon	9274
Pairs in LDOCE Lexicon	9200
Pairs in LCS and LDOCE	5654
Verbs fetched completely	1780
Precision	61%
Recall	61%

Table 5: Experiment on LCS-based Lexicon

	All Verbs in Minipar Lexicon	**Common verbs with LCS Lexicon**
Verbs in LDOCE Lexicon	5648	5648
Verbs in Minipar Lexicon	8159	4001
Common verbs in Minipar and LDOCE	5425	3721
Pairs in Minipar Lexicon	10006	7567
Pairs in LDOCE Lexicon	11786	9141
Pairs in Minipar and LDOCE	8014	6124
Verbs fetched completely	3002	1875
Precision	80%	81%
Recall	68%	67%

Table 6: Experiments on Minipar-based Lexicon

We compare two results: one where L is the Minipar-based lexicon and one where L is the LCS-based lexicon. Table 5 gives the number of verbs used in the LCS-based lexicon and the LDOCE-based lexicon, showing the precision and recall. The row showing the number of verbs fetched completely gives the number of verbs in the LCS lexicon which contains all the patterns in the LDOCE entry for the same verb. Both the precision and the recall for LCS-based lexicon with the manually-crafted mapping is 61%.

We did the same experiment for the Minipar-based lexicon in two different ways, first with all the verbs in the Minipar lexicon and then with only the verbs occurring in both the LCS and Minipar lexicons. The second approach is useful for a direct comparison between the Minipar- and LCS-based lexicons. As before, we used the LDOCE-based lexicon as our gold standard. The results are shown in Table 6. The definitions of entries are the same as in Table 5.

The number of Minipar verbs in Minipar occurring in the LCS lexicon is different from the total number of LCS verbs because some LCS verbs (266 of them) do not appear in Minipar lexicon. The results indicate that the Minipar-based lexicon yields much better precision, with an improvement of nearly 25% over the LCS-based lexicon. The recall is low because Minipar does not take modifiers into account most of the time. This results in missing nearly all patterns with PPs, such as [NP.PP] and [NP.NP.PP]. However, the recall achieved is 6% more than the recall for the LCS-based lexicon.

Verbs in LDOCE Lexicon	5648
Verbs in Intersection Lexicon	3623
Common verbs in Int. and LDOCE	3368
Pairs in Intersection Lexicon	4564
Pairs in LDOCE Lexicon	8366
Pairs in Int. and LDOCE	4156
Verbs fetched completely	1265
Precision	91%
Recall	50%

Table 7: Experiment on Intersection Lexicon

Finally, we conducted an experiment to see how the intersection of the Minipar and LCS lexicons compares to the LDOCE-based lexicon. For this experiment, we included only the verbs and patterns occurring in both lexicons. The results are shown in Table 7 in a format similar to previous tables.

The number of common verbs differs from the previous ones because we omit the verbs which do not have any patterns across the two lexicons. The results are not surprising: High precision is achieved because only those patterns that occur in both lexicons are included in the intersection lexicon; thus, the total number of pairs is reduced significantly. For the same reason, the recall is significantly reduced.

The highest precision is achieved by the intersection of two lexicons, but at the expense of recall. We found that the precision was higher for Minipar than for the LCS lexicon, but when we examined this in more detail, we found that this was almost entirely due to "double counting" of entries with optional modifiers in the LCS-based lexicon. For example, the single LCS-based grid _ag_th,instr(with) corresponds to two syntactic patterns, [NP.NP] and [NP.NP.PP], while LDOCE views these as the single pattern [NP.NP]. Specifically, 53% of the non-matching LCS-based patterns are [NP.NP.PP]—and 93% of these co-occur with [NP.NP]. Similarly, 13% of the non-matching LCS-based patterns are pattern [NP.PP]—and 80% of these co-occur with [NP].

This is a significant finding, as it reveals that our precision is spuriously low in our comparison with the "gold standard." In effect, we should be counting the LCS-based pattern [NP.NP.PP]/[NP.NP] to be a match against the LDOCE-based pattern [NP.NP]—which is a fairer comparison since neither LDOCE nor Minipar takes modifiers into account. (We henceforth refer to LCS-based the co-occurring patterns

	Minipar Lexicon (All verbs in Minipar Lexicon)	Minipar Lexicon (Common verbs with LCS Lexicon)	LCS Lexicon	Intersection of Minipar and LCS Lexicons
Precision	80%	81%	61%	91%
Enhanced Precision	81%	82%	80%	91%
Recall	68%	67%	61%	50%

Table 8: Precision and Recall Summary: Minipar- and LCS-based Lexicons

[NP.NP.PP]/[NP.NP] and [NP.PP]/[NP] as overlapping pairs.) To observe the degree of the impact of optional modifiers, we computed another precision value for the LCS-based lexicon by counting overlapping patterns once instead of twice. With this methodology, we achieved 80% (enhanced) precision. This precision value is nearly same as the value achieved with the current Minipar lexicon. Table 8 summarizes all results in terms of precision and recall.

The enhanced precision is an important and accurate indicator of the effectiveness of our approach, given that overlapping patterns arise because of (optional) modifiers. When we ignore those modifiers during our mapping process, we achieve nearly the same precision and recall with the current Minipar lexicon, which also ignores the modifiers in its code set. Moreover, overlapping patterns in our LCS-based lexicon do not affect the performance of the parser, other than to induce a more sophisticated handling of modifiers (which presumably would increase the precision numbers, if we had access to a "gold standard" that includes modifiers). For example, Minipar attaches modifiers at the clausal level instead of at the verbal level even in cases where the modifier is obviously verbal—as it would be in the LCS-based version of the parse in the sentence *She rolled the dough [PP into cookie shapes]*.

5. Ongoing Work: Automatic Generation of Syntactic Patterns

The lexicon derived from the hand-crafted mapping between the LCS lexicon and the syntactic patterns is comparable to the current Minipar lexicon. However, the mapping required a great deal of human effort, since each semantic verb class must be examined by hand in order to identify appropriate syntactic patterns. The process is error-prone, laborious, and time-intensive (approximately 3-4 person-months). Moreover, it requires that the mapping be done again by a human every time the LCS lexicon is updated.

In a recent experiment, we developed an automated mapping (in 2 person-weeks) that takes into account both semantic roles and some additional features stored in the LCS database, without reference to the class number. The mapping is based primarily on the thematic role, however in some situations the thematic roles themselves are not sufficient to determine the type of the argument. In such cases, the correct form is assigned using featural information associated with that specific verb in the LCS database.

Table 10 summarizes the automated mapping rules. The thematic role "prop" is an example of a case where featural information is necessary (e.g., (cform inf)), as there are five different patterns to choose from for this thematic role. Similarly, whether a "pred" role is an NP or AP is determined by featural information. For example, this role becomes an AP for the verb $behave$ in class 29.6.a while it is mapped onto an NP for the verb $carry$ in class 54.2. In the cases where the syntactic pattern is ambiguous and there is no specification for the verbs, default values are used for the mapping: BARE for "prop", AP for "pred" and NP for "perc".

Syntactic patterns for each thematic grid are computed by combining the results of the mapping from each thematic role in the grid to a syntactic pattern, one after another. If the grid includes optional roles, every possibility is explored and the syntactic patterns for each of them is included in the whole list of patterns for that grid. For example, the syntactic patterns for _ag_th,instr(with) include the patterns for both _ag_th and _ag_th_instr(with), which are [NP.NP] and [NP.NP.PP.with].

Note that this approach eliminates the need for using the same syntactic patterns for all verbs in a specific class: Verbs in the same class can be assigned different syntactic patterns with the help of additional features in the database. Thus, we need not rely on the semantic class number at all during this mapping. We can easily update the resulting lexicons when there is any change on the semantic classes or thematic grids of some verbs.

This experiment resulted in a parsing lexicon that has virtually the same precision/recall as that of the manually generated LCS-based lexicon above. (See Table 9.) As in the case of the manually generated mappings, the enhanced precision is 80%, which is

Verbs in LDOCE Lexicon	5648
Verbs in LCS Lexicon	4267
Common verbs in LCS and LDOCE	3757
Pairs in LCS Lexicon	9253
Pairs in LDOCE Lexicon	9200
Pairs in LCS and LDOCE	5634
Verbs fetched completely	1781
Precision	61%
Enhanced Precision	80%
Recall	61%

Table 9: Precision and Recall of Automatic Generation of Syntactic Patterns

Thematic Role	**Syntactic Patterns**
particle	PREP
prop(...), mod-prop(...), info(...)	FIN or INF or ING or PP
all other role(...)	PP
th, exp, info	FIN or INF or ING or NP
prop	NP or ING or INF or FIN or BARE
pred	AP or NP
perc	[NP.ING] or [NP.BARE]
all other roles	NP

Table 10: Syntactic Patterns Corresponding to Thematic Roles

only 1-2% lower than that of the current Minipar-based lexicon.

Our approach demonstrates that examination of thematic-role and featural information in the LCS-based lexicon is sufficient for executing this mapping automatically. Automating our approach gives us the flexibility of re-running the program if the structure of the database changes (e.g., an LCS representation is modified or class membership changes) and of porting to a new language with minimal effort.

6. Discussion

In all experiments reported above, both the LCS- and Minipar-based lexicons yield low recall values. Upon further investigation, we found that LDOCE is too specific in assigning codes to verbs. Most of the patterns associated with the verbs are rare—cases not considered in the LCS- and Minipar-based lexicons. Because of that, we believe that the recall values will improve if we take only a subset of LDOCE-based lexicon, e.g., those associated with the most frequent verb-pattern pairs in a large corpus. This is a future research direction considered in the next section.

The knowledgeable reader may question the mapping of a Levin-style lexicon into syntactic codes, given that Levin's original proposal is to investigate verb meaning through examination of syntactic patterns, or *alternations*, in the first place. As alluded to in Section 1., there are several ways in which this database has become more than just a "semantified" version of a syntactic framework; we elaborate on this further here.

Levin's original framework omitted a large number of verbs—and verb senses for existing Levin verbs—which we added to the database by semi-automatic techniques. Her original framework contained 3024 verbs in 192 classes numbering between 9.1 and 57—a total of 4186 verb entries. These were grouped together primarily by means of syntactic alternations. Our augmented database contains 4432 verbs in 492 classes with more specific numbering (e.g., "51.3.2.a.ii") including additional class numbers for new classes that Levin did not include in her work (between 000 and 026)—a total of 9844 verb entries. These were categorized according to semantic information (using WordNet synsets coupled with syntactic filtering) (Dorr, 1997)—not syntactic alternations.

An example of an entry that we added to the database is the verb *oblige*. We have assigned a semantic representation and thematic grid to this verb, creating a new class 002—which we call *Coerce Verbs*—corresponding to verbs whose underlying meaning corresponds to "force to act". Because Levin's repository omits verbs taking clausal complements, several other verbs with a similar meaning fell into this class (e.g., *coerce*, *compel*, *persuade*) including some that were already included in the original system, but not in this class (e.g., *ask*). Thus, the LCS Database contains 50% more verbs and twice as many verb entries since the original framework of Levin. The result is that we can now parse constructions such as *She compelled him to eat* and *She asked him to eat*, which would not have been analyzable had we compiled our parsing lexicon on the basis of Levin's

classes alone.

Levin's original proposal also does not contain semantic representations or thematic grids. When we built the LCS database, we examined each verb class carefully by hand to determine the underlying components of meaning unifying the members of that class. For example, the LCS representation that we generated for verbs in the *put* class includes components of meaning corresponding to "spatial placement in some manner," thus covering *dangle*, *hang*, *suspend*, etc.

From these hand-generated LCS representations, we derived our thematic grids—the same ones that are mapped onto our syntactic patterns. For example, position 1 (the highest leftmost argument in the LCS) is always mapped into the agent role of the thematic grid. The grids are organized into a thematic hierarchy that provides the basis for determining argument assignments, thus enhancing the generation process in ways that could not have been done previously with Levin's classes alone—e.g., producing constructions like *John sent a book to Paul* instead of constructions like *The book sent John to Paul*. Although the value of the thematic hierarchy seems most relevant to generation, the overall semantic/thematic hierarchical organization enables the automatic construction of lexicons that are equally suitable for both parsing and generation, thus reducing our overall lexical acquisition effort for both processes.

Beyond the above considerations, the granularity of the original Levin framework also was not adequate for our interlingual MT and lexical acquisition efforts. Our augmented form of this repository has brought about a more refined classification in which we are able to accommodate aspectual distinctions. We encode knowledge about aspectual features (e.g., *telicity*) in our LCS representations, thus sub-dividing the classes into more specific sub-classes. The tests used for this sub-division are purely *semantic* in nature, not syntactic. An example is the Dowty-style test "*He was X-ing* entails *He has X-ed*" (Dowty, 1979), where X is atelic (as in *run*) only if this entailment is considered valid by a human—and telic otherwise (as in *win*).

The inclusion of this type of knowledge allows us to refine Levin's classification significantly. An example is Class 35.6—*Ferret Verbs*: In Levin's original framework, this class conflated verbs occurring in different aspectual categories. Using the semantic tests above, we found that, in fact, these verbs should be divided as follows (Olsen et al., 1997):

Ferret Verbs: nose ferret tease (telic); seek (atelic)

The implication of this division for parsing is that the verbal arguments are constrained in a way that was not available to us in the original Levin-style classification—thus easing the job of the parser in choosing attachment points:

Telic:

∗He ferreted the truth from him.

He ferreted the truth out of him

Atelic:

He sought the truth from him.

∗He sought the truth out of him

Finally, Levin makes no claims as to the applicability of the English classes to other languages. Orienting our LCS database more toward semantic (aspectual) features rather than syntactic alternations has brought us closer to an interlingual representation that has now been demonstrably ported (quickly) to multiple languages including Arabic, Chinese, and Spanish. For example, telicity has been shown to be a crucial deciding feature in translating between divergence languages (Olsen et al., 1998), as in the translation of English *run across* as Spanish *cruzar corriendo*.

To summarize, our work is intended to: (1) Investigate the realization of a parsing lexicon from an LCS database that has developed from extensive semantic enhancements to an existing framework of verb classes and (2) Automate this technique so that it is directly applicable to LCS databases in other languages.

7. Future Work and Conclusions

Our ongoing work involves the following:

1. Using a subset of LDOCE-based lexicon by taking only the most frequent verb-pattern pairs in a big corpus: We expect that this approach will produce more realistic recall values.

2. Creating parsing lexicons for different languages: Once we have an automated mapping from the semantic lexicon to the set of syntactic patterns, we can use this method to create parsing lexicons from semantic lexicons that we already have available in other languages (Chinese, Spanish and Arabic).

3. Integration of these parsing lexicons in ongoing machine translation work (Habash and Dorr, 2001): We will feed the created lexicons into a parser and examine how successful the lexicons are. The same lexicons will also be used in our current clustering project.

Some of the ideas mentioned above are explored in detail in (Ayan and Dorr, 2002).

We conclude that it is possible to produce a parsing lexicon by projecting from LCS-based lexical entries—achieving precision and recall on a par with

a syntactic lexicon (Minipar) encoded by hand specifically for English. The consequence of this result is that, as semantic lexicons become increasingly available for multiple languages (ours are now available in English, Chinese, and Arabic), we are able to produce parsing lexicons automatically for each language.

Acknowledgments

This work has been supported, in part, by ONR MURI Contract FCPO.810548265 and Mitre Contract 010418-7712.

8. References

Necip Fazil Ayan and Bonnie J. Dorr. 2002. Creating Parsing Lexicons From Semantic Lexicons Automatically and Its Applications. Technical report, University of Maryland, College Park, MD. Technical Report: LAMP-TR-084, CS-TR-4352, UMIACS-TR-2002-32.

Michael Brent. 1993. From Grammar to Lexicon: Unsupervised Learning of Lexical Syntax. *Computational Linguistics*, 19(2):243–262.

Ted Briscoe and John Carroll. 1997. Automatic extraction of subcategorization from corpora. In *Proceedings of the 5th Conference on Applied Natural Language Processing (ANLP-97)*, Washington, DC.

J. Carroll and C. Grover. 1989. The Derivation of a Large Computational Lexicon for English from LDOCE. In B. Boguraev and Ted Briscoe, editors, *Computational lexicography for natural language processing*, pages 117–134. Longman, London.

Bonnie J. Dorr. 1993. *Machine Translation: A View from the Lexicon*. The MIT Press, Cambridge, MA.

Bonnie J. Dorr. 1997. Large-Scale Dictionary Construction for Foreign Language Tutoring and Interlingual Machine Translation. *Machine Translation*, 12(4):271–322.

Bonnie J. Dorr. 2001. LCS Verb Database. Technical Report Online Software Database, University of Maryland, College Park, MD. http://www.umiacs.umd.edu/~bonnie/-LCS_Database_Documentation.html.

David Dowty. 1979. *Word Meaning in Montague Grammar*. Reidel, Dordrecht.

Dania Egedi and Patrick Martin. 1994. A Freely Available Syntactic Lexicon for English. In *Proceedings of the International Workshop on Sharable Natural Language Resources*, Nara, Japan.

Ralph Grishman, Catherine Macleod, and Adam Meyers. 1994. Comlex Syntax: Building a Computational Lexicon. In *Proceedings of the COLING*, Kyoto.

Nizar Habash and Bonnie Dorr. 2001. Large-Scale Language Independent Generation Using Thematic Hierarchies. In *Proceedings of MT Summit VIII, Santiago de Compostella, Spain*.

Beth Levin. 1993. *English Verb Classes and Alternations: A Preliminary Investigation*. University of Chicago Press, Chicago, IL.

Dekang Lin. 1993. Principle-Based Parsing without Overgeneration. In *Proceedings of ACL-93*, pages 112–120, Columbus, Ohio.

Dekang Lin. 1998. Dependency-Based Evaluation of MINIPAR. In *Proceedings of the Workshop on the Evaluation of Parsing Systems, First International Conference on Language Resources and Evaluation*, Granada, Spain, May.

Christopher D. Manning. 1993. Automatic Acquisition of a Large Subcategorization Dictionary from Corpora. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics*, pages 235–242, Columbus, Ohio.

R. Mitten. 1992. *Computer-Usable Version of Oxford Advanced Learner's Dictionary of Current English*. Oxford Text Archive.

Mari Broman Olsen, Bonnie J. Dorr, and Scott C. Thomas. 1997. Toward Compact Monotonically Compositional Interlingua Using Lexical Aspect. In *Proceedings of the Workshop on Interlinguas in MT, MT Summit, New Mexico State University Technical Report MCCS-97-314*, pages 33–44, San Diego, CA, October. Also available as UMIACS-TR-97-86, LAMP-TR-012, CS-TR-3858, University of Maryland.

Mari Broman Olsen, Bonnie J. Dorr, and Scott C. Thomas. 1998. Enhancing Automatic Acquisition of Thematic Structure in a Large-Scale Lexicon for Mandarin Chinese. In *Proceedings of the Third Conference of the Association for Machine Translation in the Americas, AMTA-98, in Lecture Notes in Artificial Intelligence, 1529*, pages 41–50, Langhorne, PA, October 28–31.

P. Procter. 1978. *Longman Dictionary of Contemporary English*. Longman, London.

Suzanne Stevenson and Paola Merlo. 2002a. A Multilingual Paradigm for Automatic Verb Classification. In *Proceedings of Association of Computational Linguistics*, Philadelphia, PA.

Suzanne Stevenson and Paola Merlo. 2002b. Automatic verb classification using distributions of grammatical features. In *Proceedings of the 9th Conference of the European Chapter of ACL*, pages 45–52, Bergen, Norway.

Building Thematic Lexical Resources by Bootstrapping and Machine Learning

Alberto Lavelli[*], **Bernardo Magnini**[*], **Fabrizio Sebastiani**[†]

[*]ITC-irst
Via Sommarive, 18 – Località Povo
38050 Trento, Italy
{lavelli,magnini}@itc.it

[†]Istituto di Elaborazione dell'Informazione
Consiglio Nazionale delle Ricerche
56124 Pisa, Italy
fabrizio@iei.pi.cnr.it

Abstract

We discuss work in progress in the semi-automatic generation of *thematic lexicons* by means of *term categorization*, a novel task employing techniques from information retrieval (IR) and machine learning (ML). Specifically, we view the generation of such lexicons as an iterative process of learning previously unknown associations between terms and *themes* (i.e. disciplines, or fields of activity). The process is iterative, in that it generates, for each c_i in a set $C = \{c_1, \ldots, c_m\}$ of themes, a sequence $L_0^i \subseteq L_1^i \subseteq \ldots \subseteq L_n^i$ of lexicons, bootstrapping from an initial lexicon L_0^i and a set of text corpora $\Theta = \{\theta_0, \ldots, \theta_{n-1}\}$ given as input. The method is inspired by *text categorization*, the discipline concerned with labelling natural language texts with labels from a predefined set of themes, or categories. However, while text categorization deals with documents represented as vectors in a space of terms, we formulate the task of term categorization as one in which terms are (dually) represented as vectors in a space of documents, and in which terms (instead of documents) are labelled with themes. As a learning device, we adopt *boosting*, since (a) it has demonstrated state-of-the-art effectiveness in a variety of text categorization applications, and (b) it naturally allows for a form of "data cleaning", thereby making the process of generating a thematic lexicon an iteration of generate-and-test steps.

1. Introduction

The generation of *thematic lexicons* (i.e. lexicons consisting of specialized terms, all pertaining to a given theme or discipline) is a task of increased applicative interest, since such lexicons are of the utmost importance in a variety of tasks pertaining to natural language processing and information access.

One of these tasks is to support text search and other information retrieval applications in the context of thematic, "vertical" portals (aka *vortals*)[1]. Vortals are a recent phenomenon in the World Wide Web, and have grown out of the users' needs for directories, services and information resources that are both rich in information and specific to their interests. This has led to Web sites that specialize in aggregating market-specific, "vertical" content and information. Actually, the evolution from the generic portals of the previous generation (such as Yahoo!) to today's vertical portals is just natural, and is no different from the evolution that the publishing industry has witnessed decades ago with the creation of specialized magazines, targeting specific categories of readers with specific needs. To read about the newest developments in ski construction technology, skiers read specialty magazines about skiing, and not generic newspapers, and skiing magazines is also where advertisers striving to target skiers place their ads in order to be the most effective. Vertical portals are the future of commerce and information seeking on the Internet, and supporting sophisticated information access capabilities by means of thematic lexical resources is thus of the utmost importance.

Unfortunately, the generation of thematic lexicons is expensive, since it requires the intervention of specialized manpower, i.e. lexicographers and domain experts working together. Besides being expensive, such a manual approach does not allow for fast response to rapidly emerging needs. In an era of frantic technical progress new disciplines emerge quickly, while others disappear as quickly; and in an era of evolving consumer needs, the same goes for new market niches. There is thus a need of cheaper and faster methods for answering application needs than manual lexicon generation. Also, as noted in (Riloff and Shepherd, 1999), the manual approach is prone to errors of omission, in that a lexicographer may easily overlook infrequent, non-obvious terms that are nonetheless important for many tasks.

Many applications also require that the lexicons be not only thematic, but also tailored to the specific data tackled in the application. For instance, in query expansion (automatic (Peat and Willett, 1991) or interactive (Sebastiani, 1999)) for information retrieval systems addressing thematic document collections, terms synonymous or quasi-synonymous to the query terms are added to the query in order to retrieve more documents. In this case, the added terms should occur in the document collection, otherwise they are useless, and the relevant terms which occur in the document collection should potentially be added. That is, for this application the ideal thematic lexicon should contain all and only the technical terms present in the document

[1]See e.g. `http://www.verticalportals.com/`

collection under consideration, and should thus be generated directly from this latter.

1.1. Our proposal

In this paper we propose a methodology for the semi-automatic generation of thematic lexicons from a corpus of texts. This methodology relies on *term categorization*, a novel task that employs a combination of techniques from information retrieval (IR) and machine learning (ML). Specifically, we view the generation of such lexicons as an iterative process of learning previously unknown associations between terms and *themes* (i.e. disciplines, or fields of activity)[2]. The process is iterative, in that it generates, for each c_i in a set $C = \{c_1, \ldots, c_m\}$ of predefined themes, a sequence $L_0^i \subseteq L_1^i \subseteq \ldots \subseteq L_n^i$ of lexicons, bootstrapping from a lexicon L_0^i given as input. Associations between terms and themes are learnt from a sequence $\Theta = \{\theta_0, \ldots, \theta_{n-1}\}$ of sets of documents (hereafter called *corpora*); this allows to enlarge the lexicon as new corpora from which to learn become available. At iteration y, the process builds the lexicons $L_{y+1} = \{L_{y+1}^1, \ldots, L_{y+1}^m\}$ for all the themes $C = \{c_1, \ldots, c_m\}$ in parallel, from the same corpus θ_y. The only requirement on θ_y is that at least some of the terms in each of the lexicons in $L_y = \{L_y^1, \ldots, L_y^m\}$ should occur in it (if none among the terms in a lexicon L_y^j occurs in θ_y, then no new term is added to L_y^j in iteration y).

The method we propose is inspired by *text categorization*, the activity of automatically building, by means of machine learning techniques, *automatic text classifiers*, i.e. programs capable of labelling natural language texts with (zero, one, or several) thematic categories from a predefined set $C = \{c_1, \ldots, c_m\}$ (Sebastiani, 2002). The construction of an automatic text classifier requires the availability of a corpus $\psi = \{\langle d_1, C_1\rangle, \ldots, \langle d_h, C_h\rangle\}$ of preclassified documents, where a pair $\langle d_j, C_j\rangle$ indicates that document d_j belongs to all and only the categories in $C_j \subseteq C$. A general inductive process (called the *learner*) automatically builds a classifier for the set C by learning the characteristics of C from a *training set* $Tr = \{\langle d_1, C_1\rangle, \ldots, \langle d_g, C_g\rangle\} \subset \psi$ of documents. Once a classifier has been built, its effectiveness (i.e. its capability to take the right categorization decisions) may be tested by applying it to the *test set* $Te = \{\langle d_{g+1}, C_{g+1}\rangle, \ldots, \langle d_h, C_h\rangle\} = \psi - Tr$ and checking the degree of correspondence between the decisions of the automatic classifier and those encoded in the corpus.

While the purpose of text categorization is that of classifying documents represented as vectors in a space of terms, the purpose of term categorization, as we formulate it, is (dually) that of classifying terms represented as vectors in a space of documents. In this task terms are thus items that may belong, and must thus be assigned, to (zero, one, or several) themes belonging to a predefined set. In other words, starting from a set Γ_y^i of preclassified terms, a new set of terms Γ_{y+1}^i is classified, and the terms in Γ_{y+1}^i which are deemed to belong to c_i are added to L_y^i to yield L_{y+1}^i. The set Γ_y^i is composed of lexicon L_y^i, acting as the set of "positive examples", plus a set of terms known not to belong to c_i, acting as the set of "negative examples".

For input to the learning device and to the term classifiers that this will eventually build, we use "bag of documents" representations for terms (Salton and McGill, 1983, pages 78–81), dual to the "bag of terms" representations commonly used in text categorization.

As the learning device we adopt ADABOOST.MH^{KR} (Sebastiani et al., 2000), a more efficient variant of the ADABOOST.MH^{R} algorithm proposed in (Schapire and Singer, 2000). Both algorithms are an implementation of *boosting*, a method for supervised learning which has successfully been applied to many different domains and which has proven one of the best performers in text categorization applications so far. Boosting is based on the idea of relying on the collective judgment of a committee of classifiers that are trained sequentially; in training the k-th classifier special emphasis is placed on the correct categorization of the training examples which have proven harder for (i.e. have been misclassified more frequently by) the previously trained classifiers.

We have chosen a boosting approach not only because of its state-of-the-art effectiveness, but also because it naturally allows for a form of "data cleaning", which is useful in case a lexicographer wants to check the results and edit the newly generated lexicon. That is, in our term categorization context it allows the lexicographer to easily inspect the classified terms for possible misclassifications, since at each iteration y the algorithm, apart from generating the new lexicon L_{y+1}^i, ranks the terms in L_y^i in terms of their "hardness", i.e. how successful have been the generated classifiers at correctly recognizing their label. Since the highest ranked terms are the ones with the highest probability of having been misclassified in the previous iteration (Abney et al., 1999), the lexicographer can examine this list starting from the top and stopping where desired, removing the misclassified examples. The process of generating a thematic lexicon then becomes an iteration of generate-and-test steps.

This paper is organized as follows. In Section 2. we describe how we represent terms by means of a "bag of documents" representation.. For reasons of space we do not describe ADABOOST.MH^{KR}, the boosting algorithm we employ for term classification; see the extended paper for details (Lavelli et al., 2002). Section 3.1. discusses how to combine the indexing tools introduced in Section 2. with the boosting algorithm, and describes the role of the lexicographer in the iterative generate-and-test cycle. Section 3.2. describes the results of our preliminary experiments. In Section 4. we review related work on the automated generation of lexical resources, and spell out the differences between our and existing approaches. Section 5. concludes, pointing to avenues for improvement.

[2] We want to point out that our use of the word "term" is somehow different from the one often used in natural language processing and terminology extraction (Kageura and Umino, 1996), where it often denotes a *sequence* of lexical units expressing a concept of the domain of interest. Here we use this word in a neutral sense, i.e. without making any commitment as to its consisting of a single word or a sequence of words.

2. Representing terms in a space of documents

2.1. Text indexing

In text categorization applications, the process of building internal representations of texts is called *text indexing*. In text indexing, a document d_j is usually represented as a vector of term *weights* $\vec{d_j} = \langle w_{1j}, \ldots, w_{rj} \rangle$, where r is the cardinality of the *dictionary* and $0 \leq w_{kj} \leq 1$ represents, loosely speaking, the contribution of t_k to the specification of the semantics of d_j. Usually, the dictionary is equated with the set of *terms* that occur at least once in at least α documents of Tr (with α a predefined threshold, typically ranging between 1 and 5).

Different approaches to text indexing may result from different choices (i) as to what a term is and (ii) as to how term weights should be computed. A frequent choice for (i) is to use single words (minus stop words, which are usually removed prior to indexing) or their stems, although some researchers additionally consider noun phrases (Lewis, 1992) or "bigrams" (Caropreso et al., 2001). Different "weighting" functions may be used for tackling issue (ii), either of a probabilistic or of a statistical nature; a frequent choice is the *normalized* $tfidf$ function (see e.g. (Salton and Buckley, 1988)), which provides the inspiration for our "term indexing" methodology spelled out in Section 2.2..

2.2. Abstract indexing and term indexing

Text indexing may be viewed as a particular instance of *abstract indexing*, a task in which abstract objects are represented by means of abstract features, and whose underlying metaphor is, by and large, that the semantics of an object corresponds to the *bag of features* that "occur" in it[3]. In order to illustrate abstract indexing, let us define a *token* τ to be a specific occurrence of a given feature $f(\tau)$ in a given object $o(\tau)$, let T be the set of all tokens occurring in any of a set of objects O, and let F be the set of features of which the tokens in T are instances. Let us define the *feature frequency* $ff(f_k, o_j)$ of a feature f_k in an object o_j as

$$ff(f_k, o_j) = |\{\tau \in T \mid f(\tau) = f_k \ \wedge \ o(\tau) = o_j\}| \quad (1)$$

We next define the *inverted object frequency* $iof(f_k)$ of a feature f_k as

$$iof(f_k) = \quad (2)$$
$$= \log \frac{|O|}{|\{o_j \in O \mid \exists \tau \in T : f(\tau) = f_k \ \wedge \ o(\tau) = o_j\}|}$$

and the *weight* $w(f_k, o_j)$ of feature f_k in object o_j as

$$w_{kj} = w(f_k, o_j) = \quad (3)$$
$$= \frac{ff(f_k, o_j) \cdot iof(f_k)}{\sqrt{\sum_{s=1}^{|F|} (ff(f_s, o_j) \cdot iof(f_s))^2}}$$

We may consider the $w(f_k, o_j)$ function of Equation (3) as an *abstract indexing function*; that is, different instances of this function are obtained by specifying different choices for the set of objects O and set of features F.

The well-known text indexing function $tfidf$, mentioned in Section 2.1., is obtained by equating O with the training set of documents and F with the dictionary; T, the set of occurrences of elements of F in the elements of O, thus becomes the set of term occurrences.

Dually, a term indexing function may be obtained by switching the roles of F and O, i.e. equating F with the training set of documents and O with the dictionary; T, the set of occurrences of elements of F in the elements of O, is thus again the set of term occurrences (Schäuble and Knaus, 1992; Sheridan et al., 1997).

It is interesting to discuss the kind of intuitions that Equations (1), (2) and (3) embody in the dual cases of text indexing and term indexing:

- Equation (1) suggests that when a feature occurs multiple times in an object, the feature characterizes the object to a higher degree. In text indexing, this indicates that the more often a term occurs in a document, the more it is representative of its content. In term indexing, this indicates that the more often a term occurs in a document, the more the document is representative of the content of the term.
- Equation (2) suggests that the fewer the objects a feature occurs in, the more representative it is of the content of the objects in which it occurs. In text indexing, this means that terms that occur in too many documents are not very useful for identifying the content of documents. In term indexing, this means that the more terms a document contains (i.e. the longer it is), the less useful it is for characterizing the semantics of a term it contains.
- The intuition ("length normalization") that supports Equation (3) is that weights computed by means of $ff(f_k, o_j) \cdot iof(f_k)$ need to be normalized in order to prevent "longer objects" (i.e. ones in which many features occur) to emerge (e.g. to be scored higher in document-document similarity computations) just because of their length and not because of their content. In text indexing, this means that longer documents need to be deemphasized. In term indexing, this means instead that terms that occur in many documents need to be deemphasized[4].

It is also interesting to note that any program or data structure that implements $tfidf$ for text indexing may be used straightaway, with no modification, for term indexing: one needs only to feed the program with the terms in place of the documents and viceversa.

[3] "Bag" is used here in its set-theoretic meaning, as a synonym of *multiset*, i.e. a set in which the same element may occur several times. In text indexing, adopting a "bag of words" model means assuming that the number of times that a given word occurs in the same document is semantically significant. "Set of words" models, in which this number is assumed not significant, are thus particular instances of bag of words models.

[4] Incidentally, it is interesting to note that in switching from text indexing to term indexing, Equations (2) and (3) switch their roles: the intuition that terms occurring in many documents should be deemphasized is implemented in Equation (2) in text indexing and Equation (3) in term indexing, while the intuition that longer documents need to be deemphasized is implemented in Equation (3) in text indexing and Equation (2) in term indexing.

3. Generating thematic lexicons by bootstrapping and learning

3.1. Operational methodology

We are now ready to describe the overall process that we will follow for the generation of thematic lexicons. The process is iterative: we here describe the y-th iteration. We start from a set of thematic lexicons $L_y = \{L_y^1, \ldots, L_y^m\}$, one for each theme in $C = \{c_1, \ldots, c_m\}$, and from a corpus θ_y. We index the terms that occur in θ_y by means of the term indexing technique described in Section 2.2.; this yields, for each term t_k, a representation consisting of a vector of weighted documents, the length of the vector being $r = |\theta_y|$.

By using $L_y = \{L_y^1, \ldots, L_y^m\}$ as a training set, we then generate m classifiers $\Phi_y = \{\Phi_y^1, \ldots, \Phi_y^m\}$ by applying the ADABOOST.MH^{KR} algorithm. While generating the classifiers, ADABOOST.MH^{KR} also produces, for each theme c_i, a ranking of the terms in L_y^i in terms of how hard it was for the generated classifiers to classify them correctly, which basically corresponds to their probability of being misclassified examples. The lexicographer can then, if desired, inspect L_y and remove the misclassified examples, if any (possibly rerunning, especially if these latter were a substantial number, ADABOOST.MH^{KR} on the "cleaned" version of L_y). At this point, the terms occurring in θ_y that ADABOOST.MH^{KR} has classified under c_i are added (possibly, after being checked by the lexicographer) to L_y^i, yielding L_{y+1}^i. Iteration $y+1$ can then take place, and the process is repeated again.

Note that an alternative approach is to involve the lexicographer only after the last iteration, and not after each iteration. For instance, Riloff and Shepherd (Riloff and Shepherd, 1999) perform several iterations, at each of which they add to the training set (without human intervention) the new items that have been attributed to the category with the highest confidence. After the last iteration, a lexicographer inspects the list of added terms and decides which one to remove, if any. This latter approach has the advantage of requiring the intervention of the lexicographer only once, but has the disadvantage that spurious terms added to lexicon at early iterations can cause, if not promptly removed, new spurious ones to be added in the next iterations, thereby generating a domino effect.

3.2. Experimental methodology

The process we have described in Section 3.1. is the one that we would apply in an operational setting. In an experimental setting, instead, we are also interested in evaluating the effectiveness of our approach on a benchmark. The difference with the process outlined in Section 3.1. is that at the beginning of the process the lexicon L_y is split into a training set and a test set; the classifiers are learnt from the training set, and are then tested on the test set by checking how good they are at extracting the terms in the test set from the corpus θ_y. Of course, in order to guarantee a fair evaluation, the terms that never occur in θ_y are removed from the test set, since there is no way that the algorithm (or any other algorithm that extracts terms from a corpus) could possibly guess them.

Category c_i		expert judgments	
		YES	**NO**
classifier judgments	**YES**	TP_i	FP_i
	NO	FN_i	TN_i

Table 1: The contingency table for category c_i. Here, FP_i (*false positives wrt* c_i) is the number of test terms incorrectly classified under c_i; TN_i (*true negatives wrt* c_i), TP_i (*true positives wrt* c_i) and FN_i (*false negatives wrt* c_i) are defined accordingly.

We will comply with standard text categorization practice in evaluating term categorization effectiveness by a combination of *precision* (π), the percentage of positive categorization decisions that turn out to be correct, and *recall* (ρ), the percentage of positive, correct categorization decisions that are actually taken. Since most classifiers can be tuned to emphasize one at the expense of the other, only combinations of the two are usually considered significant. Following common practice, as a measure combining the two we will adopt their harmonic mean, i.e. $F_1 = \frac{2\pi\rho}{\pi+\rho}$. Effectiveness will be computed with reference to the contingency table illustrated in Table 1. When effectiveness is computed for several categories, the results for individual categories must be averaged in some way; we will do this both by *microaveraging* ("categories count proportionally to the number of their positive training examples"), i.e.

$$\pi^{\mu} = \frac{TP}{TP+FP} = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{|\mathcal{C}|}(TP_i+FP_i)}$$

$$\rho^{\mu} = \frac{TP}{TP+FN} = \frac{\sum_{i=1}^{m} TP_i}{\sum_{i=1}^{m}(TP_i+FN_i)}$$

and by *macroaveraging* ("all categories count the same"), i.e.

$$\pi^{M} = \frac{\sum_{i=1}^{|\mathcal{C}|} \pi_i}{m} \qquad \rho^{M} = \frac{\sum_{i=1}^{m} \rho_i}{m}$$

Here, "μ" and "M" indicate microaveraging and macroaveraging, respectively, while the other symbols are as defined in Table 1. Microaveraging rewards classifiers that behave well on *frequent categories* (i.e. categories with many positive test examples), while classifiers that perform well also on infrequent categories are emphasized by macroaveraging. Whether one or the other should be adopted obviously depends on the application.

3.3. Our experimental setting

We now describe the resources we have used in our experiments.

3.3.1. The corpora

As the corpora $\Theta = \{\theta_1, \ldots, \theta_n\}$, we have used various subsets of the Reuters Corpus Volume I (RCVI), a corpus of documents recently made available by Reuters[5] for text categorization experimentation and consisting of about 810,000 news stories. Note that, although the texts of RCVI

[5] http://www.reuters.com/

are labelled by thematic categories, we have not made use of such labels (not it would have made much sense to use them, given that these categories are different from the ones we are working with); the reasons we have chosen this corpus instead of other corpora of unlabelled texts are inessential.

3.3.2. The lexicons

As the thematic lexicons we have used subsets of an extension of WordNet, that we now describe.

WordNet (Fellbaum, 1998) is a large, widely available, non-thematic, monolingual, machine-readable dictionary in which sets of synonymous words are grouped into synonym sets (or *synsets*) organized into a directed acyclic graph. In this work, we will always refer to WordNet version 1.6.

In WordNet only a few synsets are labelled with thematic categories, mainly contained in the glosses. This limitation is overcome in WordNetDomains, an extension of WordNet described in (Magnini and Cavaglià, 2000) in which each synset has been labelled with one or more from a set of 164 thematic categories, called *domains*[6]. The 164 domains of WordNetDomains are a subset of the categories belonging to the classification scheme of Dewey Decimal Classification (DDC (Mai Chan et al., 1996)); example domains are ZOOLOGY, SPORT, and BASKETBALL.

These 164 domains have been chosen from the much larger set of DDC categories since they are the most popular labels used in dictionaries for sense discrimination purposes. Domains have long been used in lexicography (where they are sometimes called *subject field codes* (Procter, 1978)) to mark technical usages of words. Although they convey useful information for sense discrimination, they typically tag only a small portion of a dictionary. WordNetDomains extends instead the coverage of domain labels to an entire, existing lexical database, i.e. WordNet.

A domain may include synsets of different syntactic categories: for instance, the MEDICINE domain groups together senses from Nouns, such as `doctor#1` (the first among several senses of the word "doctor") and `hospital#1`, and from Verbs, such as `operate#7`. A domain may include senses from different WordNet subhierarchies. For example, SPORT contains senses such as `athlete#1`, which descends from `life_form#1`; `game_equipment#1`, from `physical_object#1`; `sport#1`, from `act#2`; and `playing_field#1`, from `location#1`. Note that domains may group senses of the same word into thematic clusters, with the side effect of reducing word polysemy in WordNet.

The annotation methodology used in (Magnini and Cavaglià, 2000) for creating WordNetDomains was mainly manual, and based on lexico-semantic criteria which take advantage from the already existing conceptual relations in WordNet. First, a small number of high level synsets were manually annotated with their correct domains. Then, an automatic procedure exploiting some of the WordNet relations (i.e. hyponymy, troponymy, meronymy, antonymy and pertain-to) was used in order to extend these assignments to all the synsets reachable through inheritance. For example, this procedure automatically marked the synset {`beak, bill, neb, nib`} with the code ZOOLOGY, starting from the fact that the synset {`bird`} was itself tagged with ZOOLOGY, and following a "part-of" relation (one of the meronymic relations present in WordNet). In some cases the inheritance procedure had to be manually blocked, inserting an "exception" in order to prevent a wrong propagation. For instance, if blocking had not been used, the term `barber_chair#1`, being a "part-of" `barbershop#1`, which is annotated with COMMERCE, would have inherited COMMERCE, which is unsuitable.

For the purpose of the experiments reported in this paper, we have used a simplified variant of WordNetDomains, called WordNetDomains(42). This was obtained from WordNetDomains by considering only 42 highly relevant labels, and tagging by a given domain c_i also the synsets that, in WordNetDomains, were tagged by the domains immediately related to c_i in a hierarchical sense (that is, the parent domain of c_i and all the children domains of c_i). For instance, the domain SPORT is retained into WordNetDomains(42), and labels both the synsets that it originally labelled in WordNetDomains, plus the ones that in WordNetDomains were labelled under its children categories (e.g. VOLLEY, BASKETBALL, ...) or under its parent category (FREE-TIME). Since FREE-TIME has another child (PLAY) which is also retained in WordNetDomains(42), the synsets originally labelled by FREE-TIME will now be labelled also by PLAY, and will thus have multiple labels. However, that a synset may have multiple labels is true in general, i.e. these labels need not have any particular relation in the hierarchy.

This restriction to the 42 most significant categories allows to obtain a good compromise between the conflicting needs of avoiding data sparseness and preventing the loss of relevant semantic information. These 42 categories belong to 5 groups, where the categories in a given group are all the children of the same WordNetDomains category, which is however not retained into WordNetDomains(42); for example, one group is formed by SPORT and PLAY, which are both children of FREE-TIME (not included into WordNetDomains(42)).

3.3.3. The experiment

We have run several experiments for different choices of the subset of RCVI chosen as corpus of text θ_y, and for different choices of the subsets of WordNetDomains(42) chosen as training set Tr_y and test set Te_y. We first describe how we have run a generic experiment, and then go on to describe the sequence of different experiments we have run. For the moment being we have run experiments consisting of one iteration only of the bootstrapping process. In future experiments we also plan to allow for multiple iterations, in which the system learns new terms also from previously learnt ones.

In our experiments we considered only nouns, thereby discarding words tagged by other syntactic categories. We plan to also consider words other than nouns in future ex-

[6] From the point of view of our term categorization task, the fact that more than one domain may be attached to the same synset means that ours is a *multi-label* categorization task (Sebastiani, 2002, Section 2.2).

periments.

For each experiment, we discarded all documents that did not contain any term from the training lexicon Tr_y, since they do not contribute in representing the meaning of training documents, and thus could not possibly be of any help in building the classifiers. Next, we discarded all "empty" training terms, i.e. training terms that were not contained in any document of θ_y, since they could not possibly contribute to learning the classifiers. Also empty test terms were discarded, since no algorithm that extracts terms from corpora could possibly extract them. Quite obviously, we also do not use the terms that occur in θ_y but belong neither to the training set Tr_y nor to the test set Te_y.

We then lemmatized all remaining documents and annotated the lemmas with part-of-speech tags, both by means of the TREETAGGER package (Schmid, 1994); we also used the WordNet morphological analyzer in order to resolve ambiguities and lemmatization mistakes. After tagging, we applied a filter in order to identify the words actually contained in WordNet, including multiwords, and then we discarded all terms but nouns. The final set of terms that resulted from this process was randomly divided into a training set Tr_y (consisting of two thirds of the entire set) and a test set Te_y (one third). As negative training examples of category c_i we chose all the training terms that are not positive examples of c_i.

Note that in this entire process we have not considered the grouping of terms into synsets; that is, the lexical units of interest in our application are the terms, and not the synsets. The reason is that RCVI is not a sense-tagged corpus, and for any term occurrence τ it is not clear to which synset τ refers to.

3.3.4. The results

Our experimental results on this task are still very preliminary, and are reported in Table 2.

Instead of tackling the entire RCVI corpus head on, for the moment being we have run only small experiments on limited subsets of it (up to 8% of its total size), with the purpose of getting a feel for which are the dimensions of the problem that need investigation; for the same reason, for the moment being we have used only a small number of boosting iterations (500). In Table 2, the first three lines concern experiments on the news stories produced on a single day (08.11.1996); the next three lines use the news stories produced in a single week (08.11.1996 to 14.11.1996), and the last six lines use the news stories produced in an entire month (01.11.1996 to 30.11.1996). Only training and test terms occurring in at least x documents were considered; the experiments reported in the same block of lines differ for the choice of the x parameter.

There are two main conclusions we can draw from these still preliminary experiments. The first conclusion is that F_1 values are still low, at least if compared to the F_1 values that have been obtained in *text* categorization research on the same corpus (Ault and Yang, 2001); a lot of work is still needed in tuning this approach in order to obtain significant categorization performance. The low values of F_1 are mostly the result of low recall values, while precision tends to be much higher, often well above the 70% mark. Note that the low absolute performance might also be explained, at least partially, with the imperfect quality of the WordNetDomains(42) resource, which was generated by a combination of automatic and manual procedures and did no undergo extensive checking afterwards.

The second conclusion is that results show a constant and definite improvement when higher values of x are used, despite the fact that higher levels of x mean a higher number of labels per term, i.e. more polysemy. This is not surprising, since when a term occurs e.g. in one document only, this means that only one entry in the vector that represents the term is non-null (i.e. significant). This is in sharp contrast with text categorization, in which the number of non-null entries in the vector representing a document equals the number of distinct terms contained in the document, and is usually at least in the hundreds. This alone might suffice to justify the difference in performance between term categorization and text categorization.

However, one reason the actual F_1 scores are low is that this is a hard task, and the evaluation standards we have adopted are considerably tough. This is discussed in the next paragraph.

No baseline? Note that we present no baseline, either published or new, against which to compare our results, for the simple fact that term categorization as we conceive it here is a novel task, and there are as yet no previous results or known approaches to the problem to compare with.

Only (Riloff and Shepherd, 1999; Roark and Charniak, 1998) have approached the problem of extending an existing thematic lexicon with new terms drawn from a text corpus. However, there are key differences between their evaluation methodology and ours, which makes comparisons difficult and unreliable. First, their "training" terms have not been chosen randomly our of a thematic dictionary, but have been carefully selected through a manual process by the authors themselves. For instance, (Riloff and Shepherd, 1999) choose words that are "frequent in the domain" and that are "(relatively) unambiguous". Of course, their approach makes the task easier, since it allows the "best" terms to be selected for training. Second, (Riloff and Shepherd, 1999; Roark and Charniak, 1998) extract the terms from texts that are known to be about the theme, which makes the task easier than ours; conversely, by using generic texts, we avoid the costly process of labelling the documents by thematic categories, and we are able to generate thematic lexicons for multiple themes at once *from the same unlabelled text corpus*. Third, their evaluation methodology is manual, i.e. subjective, in the sense that the authors themselves manually checked the results of their experiments, judging, for each returned term, how reasonable the inclusion of the term in the lexicon is[7]. This sharply contrasts with our evaluation methodology, which is completely automatic (since we measure the proficiency

[7] For instance, (Riloff and Shepherd, 1999) judged a word classified into a category correct also if they judged that "the word refers to a part of a member of the category", thereby judging the words `cartridge` and `clips` to belong to the domain WEAPONS. This looks to us a loose notion of category membership, and anyway points to the pitfalls of "subjective" evaluation methodologies.

# of docs	# of training terms	# of test terms	# of labels per term	minimum # of docs per term	Precision micro	Recall micro	F_1 micro	Precision macro	Recall macro	F_1 macro
2,689	4,424	2,212	1.96	1	0.542029	0.043408	0.080378	0.584540	0.038108	0.071551
2,689	1,685	842	2.36	5	0.512903	0.079580	0.137782	0.487520	0.078677	0.135489
2,689	1,060	530	2.55	10	0.517544	0.086131	0.147685	0.560876	0.084176	0.146383
16,003	7,975	3,987	1.76	1	0.720165	0.049631	0.092863	0.701141	0.038971	0.073837
16,003	4,132	2,066	2.02	5	0.733491	0.075121	0.136284	0.738505	0.065472	0.120281
16,003	2,970	1,485	2.15	10	0.740260	0.091405	0.162718	0.758044	0.078162	0.141712
67,953	11,313	5,477	1.66	1	0.704251	0.043090	0.081211	0.692819	0.034241	0.065256
67,953	6,829	3,414	1.83	5	0.666667	0.040816	0.076923	0.728300	0.050903	0.095155
67,953	5,335	2,668	1.92	10	0.712406	0.076830	0.138701	0.706678	0.056913	0.105342
67,953	4,521	2,261	1.99	15	0.742574	0.086445	0.154863	0.731530	0.064038	0.117766
67,953	3,317	1,659	2.10	30	0.745455	0.098439	0.173913	0.785371	0.075573	0.137878
67,953	2,330	1,166	2.25	60	0.760417	0.117789	0.203982	0.755136	0.086809	0.155718

Table 2: Preliminary results obtained on the automated lexicon generation task (see Section 3.3. for details).

of our system at discovering terms about the theme, by the capability of the system to replicate the lexicon generation work of a lexicographer), can be replicated by other researchers, and is unaffected by possible experimenter's bias. Fourth, checking one's results for "reasonableness", as (Riloff and Shepherd, 1999; Roark and Charniak, 1998) do, means that one can only ("subjectively") measure precision (i.e. whether the terms spotted by the algorithm do in fact belong to the theme), but not recall (i.e. whether the terms belonging to the theme have actually been spotted by the algorithm). Again, this is in sharp contrast with our methodology, which ("objectively") measures precision, recall, and a combination of them. Also, note that in terms of precision, i.e. the measure that (Riloff and Shepherd, 1999; Roark and Charniak, 1998) subjectively compute, our algorithm fares pretty well, mostly scoring higher than 70% even in these very preliminary experiments.

4. Related work

4.1. Automated generation of lexical resources

The automated generation of lexicons from text corpora has a long history, dating back at the very least to the seminal works of Lesk, Salton and Sparck Jones (Lesk, 1969; Salton, 1971; Sparck Jones, 1971), and has been the subject of active research throughout the last 30 years, both within the information retrieval community (Crouch and Yang, 1992; Jing and Croft, 1994; Qiu and Frei, 1993; Ruge, 1992; Schütze and Pedersen, 1997) and the NLP community (Grefenstette, 1994; Hirschman et al., 1988; Riloff and Shepherd, 1999; Roark and Charniak, 1998; Tokunaga et al., 1995). Most of the lexicons built by these works come in the form of *cluster-based thesauri*, i.e. networks of groups of synonymous or quasi-synonymous words, in which edges connecting the nodes represent semantic contiguity. Most of these approaches follow the basic pattern of (i) measuring the degree of pairwise similarity between the words extracted from a corpus of texts, and (ii) clustering these words based on the computed similarity values. When the lexical resources being built are of a *thematic* nature, the thematic nature of a word is usually established by checking whether its frequency within thematic documents is higher than its frequency in generic documents (Chen et al., 1996; Riloff and Shepherd, 1999; Schatz et al., 1996; Sebastiani, 1999) (this property is often called *salience* (Yarowsky, 1992)).

In the approach described above, the key decision is how to tackle step (i), and there are two main approaches to this. In the first approach the similarity between two words is usually computed in terms of their degree of co-occurrence and co-absence within the same document (Crouch, 1990; Crouch and Yang, 1992; Qiu and Frei, 1993; Schäuble and Knaus, 1992; Sheridan and Ballerini, 1996; Sheridan et al., 1997); variants of this approach are obtained by restricting the context of co-occurrence from the document to the paragraph, or to the sentence (Schütze, 1992; Schütze and Pedersen, 1997), or to smaller linguistic units (Riloff and Shepherd, 1999; Roark and Charniak, 1998). In the second approach this similarity is computed from head-modifier structures, by relying on the assumption that frequent modifiers of the same word are semantically similar (Grefenstette, 1992; Ruge, 1992; Strzalkowski, 1995). The latter approach can also deal with indirect co-occurrence[8], but the former is conceptually simpler, since it does not even need any parsing step.

This literature (apart from (Riloff and Shepherd, 1999; Roark and Charniak, 1998), which are discussed below) has thus taken an *unsupervised* learning approach, which can be summarized in the recipe "from a set of documents about theme t and a set of generic documents (i.e. mostly not about t), extract the words that mostly characterize t". Our work is different, in that its underlying *supervised* learning approach requires a starting kernel of terms about t, but does not require that the corpus of documents from which

[8]We say that words w_1 and w_2 *co-occur directly* when they both occur in the same document (or other linguistic context), while we say that they *co-occur indirectly* when, for some other word w_3, w_1 and w_3 co-occur directly and w_2 and w_3 co-occur directly. Perfect synonymy is not revealed by direct co-occurrence, since users tend to consistently use either one or the other synonym but not both, while it is obviously revealed by indirect co-occurrence. However, this latter also tends to reveal many more "spurious" associations than direct co-occurrence.

the terms are extracted be labelled. This makes our supervised technique particularly suitable for *extending* a previously existing thematic lexical resource, while the previously known unsupervised techniques tend to be more useful for generating one from scratch. This suggests an interesting methodology of (i) generating a thematic lexical resource by some unsupervised technique, and then (ii) extending it by our supervised technique. An intermediate approach between these two is the one adopted in (Riloff and Shepherd, 1999; Roark and Charniak, 1998), which also requires a starting kernel of terms about t, but also requires a set of documents about theme t from which the new terms are extracted.

As anyone involved in applications of supervised machine learning knows, labelled resources are often a bottleneck for learning algorithms, since labelling items by hand is expensive. Concerning this, note that our technique is advantageous, since it requires an initial set of labelled terms *only in the first bootstrapping iteration*. Once a lexical resource has been extended with new terms, extending it further only requires a new *unlabelled* corpus of documents, but no other labelled resource. This is different from the other techniques described earlier, which require, for extending a lexical resource that has just been built by means of them, a new *labelled* corpus of documents.

A work which is closer in spirit to ours than the abovementioned ones is (Tokunaga et al., 1997), since it deals with using previously classified terms as training examples in order to classify new terms. This work exploits a naive Bayesian model for classification in conjunction with another learning method, chosen among nearest neighbour, "category-based" (by which the authors basically mean a Rocchio method – see e.g. (Sebastiani, 2002, Section 6.7)) and "cluster-based" (which does not use category labels of training examples). However, these latter learning methods and (especially) the nature of their integration with the naive Bayesian model are not specified in mathematical detail, which does not allow us to make a precise comparison between the model of (Tokunaga et al., 1997) and ours. Anyway, our model is more elegant, in that it just assumes a single learning method (for which we have chosen boosting, although we might have chosen any other supervised learning method), and in that it replaces the ad-hoc notion of "co-occurrence" with a theoretically sounder "dual" theory of text indexing, which allows one, among other things, to bring to bear any kind of intuitions on term weighting, or any kind of text indexing theory, that are known from information retrieval.

4.2. Boosting

Boosting has been applied to several learning tasks related to text analysis, including POS-tagging and PP-attachment (Abney et al., 1999), clause splitting (Carreras and Màrquez, 2001b), word segmentation (Shinnou, 2001), word sense disambiguation (Escudero et al., 2000), text categorization (Schapire and Singer, 2000; Schapire et al., 1998; Sebastiani et al., 2000; Taira and Haruno, 2001), e-mail filtering (Carreras and Márquez, 2001a), document routing (Iyer et al., 2000; Kim et al., 2000), and term extraction (Vivaldi et al., 2001). Among these works, the one somehow closest in spirit to ours is (Vivaldi et al., 2001), since it is concerned with extracting medical terms from a corpus of texts. A key difference with our work is that the features by which candidate terms are represented in (Vivaldi et al., 2001) are not simply the documents they occur in, but the results of term extraction algorithms; therefore, our approach is simpler and more general, since it does not require the existence of separate term extraction algorithms.

5. Conclusion

We have reported work in progress on the semi-automatic generation of thematic lexical resources by the combination of (i) a dual interpretation of IR-style text indexing theory and (ii) a boosting-based machine learning approach. Our method does not require pre-existing semantic knowledge, and is particularly suited to the situation in which one or more preexisting thematic lexicons need to be extended and no corpora of texts classified according to the themes are available. We have run only initial experiments, which suggest that the approach is viable, although large margins of improvement exist. In order to improve the overall performance we are planning several modifications to our currently adopted strategy.

The first modification consists in performing *feature selection*, as commonly used in text categorization (Sebastiani, 2002, Section 5.4). This will consist in individually scoring (by means of the *information gain* function) all documents in terms of how indicative they are of the occurrence or non-occurrence of the categories we are interested in, and to choose only the best-scoring ones out of a potentially huge corpus of available documents.

The second avenue we intend to follow consists in trying alternative notions of what a document is, by considering as "documents" paragraphs, or sentences, or even smaller, syntactically characterized units (as in (Riloff and Shepherd, 1999; Roark and Charniak, 1998)), rather than full-blown Reuters news stories.

A third modification consists in selecting, as the negative examples of a category c_i, all the training examples that are not positive examples of c_i *and* are at the same time positive examples of (at least one of) the siblings of c_i. This method, known as the *query-zoning method* or as the *method of quasi-positive examples*, is known to yield superior performance with respect to the method we currently use (Dumais and Chen, 2000; Ng et al., 1997).

The last avenue for improvement is the optimization of the parameters of the boosting process. The obvious parameter that needs to be optimized is the number of boosting iterations, which we have kept to a minimum in the reported experiments. A less obvious parameter is the form of the initial distribution on the training examples (that we have not described here for space limitations); by changing it with respect to the default value (the uniform distribution) we will be able to achieve a better compromise between precision and recall (Schapire et al., 1998), which for the moment being have widely different values.

Acknowledgments

We thank Henri Avancini for help with the coding task and Pio Nardiello for assistance with the

ADABOOST.MH^{KR} code. Above all, we thank Roberto Zanoli for help with the coding task and for running the experiments.

6. References

Steven Abney, Robert E. Schapire, and Yoram Singer. 1999. Boosting applied to tagging and PP attachment. In *Proceedings of EMNLP-99, 4th Conference on Empirical Methods in Natural Language Processing*, pages 38–45, College Park, MD.

Thomas Ault and Yiming Yang. 2001. kNN, Rocchio and metrics for information filtering at TREC-10. In *Proceedings of TREC-10, 10th Text Retrieval Conference*, Gaithersburg, US.

Maria Fernanda Caropreso, Stan Matwin, and Fabrizio Sebastiani. 2001. A learner-independent evaluation of the usefulness of statistical phrases for automated text categorization. In Amita G. Chin, editor, *Text Databases and Document Management: Theory and Practice*, pages 78–102. Idea Group Publishing, Hershey, US.

Xavier Carreras and Lluís Márquez. 2001a. Boosting trees for anti-spam email filtering. In *Proceedings of RANLP-01, 4th International Conference on Recent Advances in Natural Language Processing*, Tzigov Chark, BG.

Xavier Carreras and Lluís Màrquez. 2001b. Boosting trees for clause splitting. In *Proceedings of CONLL-01, 5th Conference on Computational Natural Language Learning*, Toulouse, FR.

Hsinchun Chen, Chris Schuffels, and Rich Orwing. 1996. Internet categorization and search: A machine learning approach. *Journal of Visual Communication and Image Representation, Special Issue on Digital Libraries*, 7(1):88–102.

Carolyn J. Crouch and Bokyung Yang. 1992. Experiments in automated statistical thesaurus construction. In *Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval*, pages 77–87, Kobenhavn, DK.

Carolyn J. Crouch. 1990. An approach to the automatic construction of global thesauri. *Information Processing and Management*, 26(5):629–640.

Susan T. Dumais and Hao Chen. 2000. Hierarchical classification of Web content. In *Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pages 256–263, Athens, GR. ACM Press, New York, US.

Gerard Escudero, Lluís Màrquez, and German Rigau. 2000. Boosting applied to word sense disambiguation. In *Proceedings of ECML-00, 11th European Conference on Machine Learning*, pages 129–141, Barcelona, ES.

Christiane Fellbaum, editor. 1998. *WordNet: An Electronic Lexical Database*. The MIT Press, Cambridge, US.

Gregory Grefenstette. 1992. Use of syntactic context to produce term association lists for retrieval. In *Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval*, pages 89–98, Kobenhavn, DK.

Gregory Grefenstette. 1994. *Explorations in automatic thesaurus discovery*. Kluwer Academic Publishers, Dordrecht, NL.

Lynette Hirschman, Ralph Grishman, and Naomi Sager. 1988. Grammatically-based automatic word class formation. *Information Processing and Management*, 11(1/2):39–57.

Raj D. Iyer, David D. Lewis, Robert E. Schapire, Yoram Singer, and Amit Singhal. 2000. Boosting for document routing. In *Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management*, pages 70–77, McLean, US.

Yufeng Jing and W. Bruce Croft. 1994. An association thesaurus for information retrieval. In *Proceedings of RIAO-94, 4th International Conference "Recherche d'Information Assistee par Ordinateur"*, pages 146–160, New York, US.

Kyo Kageura and Bin Umino. 1996. Methods of automatic term recognition: a review. *Terminology*, 3(2):259–289.

Yu-Hwan Kim, Shang-Yoon Hahn, and Byoung-Tak Zhang. 2000. Text filtering by boosting naive Bayes classifiers. In *Proceedings of SIGIR-00, 23rd ACM International Conference on Research and Development in Information Retrieval*, pages 168–75, Athens, GR.

Alberto Lavelli, Bernardo Magnini, and Fabrizio Sebastiani. 2002. Building thematic lexical resources by term categorization. Technical report, Istituto di Elaborazione dell'Informazione, Consiglio Nazionale delle Ricerche, Pisa, IT. Forthcoming.

Michael E. Lesk. 1969. Word-word association in document retrieval systems. *American Documentation*, 20(1):27–38.

David D. Lewis. 1992. An evaluation of phrasal and clustered representations on a text categorization task. In *Proceedings of SIGIR-92, 15th ACM International Conference on Research and Development in Information Retrieval*, pages 37–50, Kobenhavn, DK.

Bernardo Magnini and Gabriela Cavaglià. 2000. Integrating subject field codes into WordNet. In *Proceedings of LREC-2000, 2nd International Conference on Language Resources and Evaluation*, pages 1413–1418, Athens, GR.

Lois Mai Chan, John P. Comaromi, Joan S. Mitchell, and Mohinder Satija. 1996. *Dewey Decimal Classification: a practical guide*. OCLC Forest Press, Albany, US, 2nd edition.

Hwee T. Ng, Wei B. Goh, and Kok L. Low. 1997. Feature selection, perceptron learning, and a usability case study for text categorization. In *Proceedings of SIGIR-97, 20th ACM International Conference on Research and Development in Information Retrieval*, pages 67–73, Philadelphia, US. ACM Press, New York, US.

Helen J. Peat and Peter Willett. 1991. The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of the American Society for Information Science*, 42(5):378–383.

Paul Procter, editor. 1978. *The Longman Dictionary of Contemporary English*. Longman, Harlow, UK.

Yonggang Qiu and Hans-Peter Frei. 1993. Concept-based query expansion. In *Proceedings of SIGIR-93, 16th ACM International Conference on Research and Devel-*

opment in Information Retrieval, pages 160–169, Pittsburgh, US.

Ellen Riloff and Jessica Shepherd. 1999. A corpus-based bootstrapping algorithm for semi-automated semantic lexicon construction. *Journal of Natural Language Engineering*, 5(2):147–156.

Brian Roark and Eugene Charniak. 1998. Noun phrase cooccurrence statistics for semi-automatic semantic lexicon construction. In *Proceedings of ACL-98, 36th Annual Meeting of the Association for Computational Linguistics*, pages 1110–1116, Montreal, CA.

Gerda Ruge. 1992. Experiments on linguistically-based terms associations. *Information Processing and Management*, 28(3):317–332.

Gerard Salton and Christopher Buckley. 1988. Term-weighting approaches in automatic text retrieval. *Information Processing and Management*, 24(5):513–523.

Gerard Salton and Michael J. McGill. 1983. *Introduction to modern information retrieval*. McGraw Hill, New York, US.

Gerard Salton. 1971. Experiments in automatic thesaurus construction for information retrieval. In *Proceedings of the IFIP Congress*, volume TA-2, pages 43–49, Ljubljana, YU.

Robert E. Schapire and Yoram Singer. 2000. BOOSTEXTER: a boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168.

Robert E. Schapire, Yoram Singer, and Amit Singhal. 1998. Boosting and Rocchio applied to text filtering. In *Proceedings of SIGIR-98, 21st ACM International Conference on Research and Development in Information Retrieval*, pages 215–223, Melbourne, AU.

Bruce R. Schatz, Eric H. Johnson, Pauline A. Cochrane, and Hsinchun Chen. 1996. Interactive term suggestion for users of digital libraries: Using subject thesauri and co-occurrence lists for information retrieval. In *Proceedings of DL-96, 1st ACM Digital Library Conference*, pages 126–133, Bethesda, US.

Peter Schäuble and Daniel Knaus. 1992. The various roles of information structures. In *Proceedings of the 16th Annual Conference of the Gesellschaft für Klassifikation*, pages 282–290, Dortmund, DE.

Helmut Schmid. 1994. Probabilistic part-of-speech tagging using decision trees. In *Proceedings of the International Conference on New Methods in Language Processing*, pages 44–49, Manchester, UK.

Hinrich Schütze and Jan O. Pedersen. 1997. A cooccurrence-based thesaurus and two applications to information retrieval. *Information Processing and Management*, 33(3):307–318.

Hinrich Schütze. 1992. Dimensions of meaning. In *Proceedings of Supercomputing'92*, pages 787–796, Minneapolis, US.

Fabrizio Sebastiani, Alessandro Sperduti, and Nicola Valdambrini. 2000. An improved boosting algorithm and its application to automated text categorization. In *Proceedings of CIKM-00, 9th ACM International Conference on Information and Knowledge Management*, pages 78–85, McLean, US.

Fabrizio Sebastiani. 1999. Automated generation of category-specific thesauri for interactive query expansion. In *Proceedings of IDC-99, 9th International Database Conference on Heterogeneous and Internet Databases*, pages 429–432, Hong Kong, CN.

Fabrizio Sebastiani. 2002. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47.

Páraic Sheridan and Jean-Paul Ballerini. 1996. Experiments in multilingual information retrieval using the SPIDER system. In *Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval*, pages 58–65, Zürich, CH.

Páraic Sheridan, Martin Braschler, and Peter Schäuble. 1997. Cross-language information retrieval in a multilingual legal domain. In *Proceedings of ECDL-97, 1st European Conference on Research and Advanced Technology for Digital Libraries*, pages 253–268, Pisa, IT.

Hiroyuki Shinnou. 2001. Detection of errors in training data by using a decision list and AdaBoost. In *Proceedings of the IJCAI-01 Workshop on Text Learning: Beyond Supervision*, Seattle, US.

Karen Sparck Jones. 1971. *Automatic keyword classification for information retrieval*. Butterworths, London, UK.

Tomek Strzalkowski. 1995. Natural language information retrieval. *Information Processing and Management*, 31(3):397–417.

Hirotoshi Taira and Masahiko Haruno. 2001. Text categorization using transductive boosting. In *Proceedings of ECML-01, 12th European Conference on Machine Learning*, pages 454–465, Freiburg, DE.

Takenobu Tokunaga, Makoto Iwayama, and Hozumi Tanaka. 1995. Automatic thesaurus construction based on grammatical relations. In *Proceedings of IJCAI-95, 14th International Joint Conference on Artificial Intelligence*, pages 1308–1313, Montreal, CA.

Takenobu Tokunaga, Atsushi Fujii, Makoto Iwayama, Naoyuki Sakurai, and Hozumi Tanaka. 1997. Extending a thesaurus by classifying words. In *Proceedings of the ACL-EACL Workshop on Automatic Information Extraction and Building of Lexical Semantic Resources*, pages 16–21, Madrid, ES.

Jordi Vivaldi, Lluís Màrquez, and Horacio Rodríguez. 2001. Improving term extraction by system combination using boosting. In *Proceedings of ECML-01, 12th European Conference on Machine Learning*, pages 515–526, Freiburg, DE.

David Yarowsky. 1992. Word-sense disambiguation using statistical models of Roget's categories trained on large corpora. In *Proceedings of COLING-92, 14th International Conference on Computational Linguistics*, pages 454–460, Nantes, FR.

Learning Grammars for Noun Phrase Extraction by Partition Search

Anja Belz

ITRI
University of Brighton
Lewes Road
Brighton BN2 4GJ, UK
Anja.Belz@itri.brighton.ac.uk

Abstract

This paper describes an application of Grammar Learning by Partition Search to noun phrase extraction, an essential task in information extraction and many other NLP applications. Grammar Learning by Partition Search is a general method for automatically constructing grammars for a range of parsing tasks; it constructs an optimised probabilistic context-free grammar by searching a space of nonterminal set partitions, looking for a partition that maximises parsing performance and minimises grammar size. The idea is that the considerable time and cost involved in building new grammars can be avoided if instead existing grammars can be automatically adapted to new parsing tasks and new domains. This paper presents results for applying Partition Search to the tasks of (i) identifying flat NP chunks, and (ii) identifying all NPs in a text. For NP chunking, Partition Search improves a general baseline result by 12.7%, and a method-specific baseline by 2.2%. For NP identification, Partition Search improves the general baseline by 21.45%, and the method-specific one by 3.48%. Even though the grammars are nonlexicalised, results for NP identification closely match the best existing results for lexicalised approaches.

1. Introduction

Grammar Learning by Partition Search is a computational learning method that constructs probabilistic grammars optimised for a given parsing task. Its main practical application is the adaptation of grammars to new tasks, in particular the adaptation of conventional, "deep" grammars to the shallow parsing tasks involved in many NLP applications. The parsing tasks investigated in this paper are NP identification and NP chunking both of which involve the detection of NP boundaries, a task which is fundamental to information extraction and retrieval, text summarisation, document classification, and other applications.

The ability to automatically adapt an existing grammar to a new parsing task saves time and expense. Furthermore, adapting deep grammars to shallow parsing tasks has a specific advantage. Existing approaches to NP extraction are mostly completely flat. They do not carry out any structural analysis above the level of the chunks and phrases they are meant to detect. Using Partition Search to adapt deep grammars for shallow parsing permits those parts of deeper structural analysis to be retained that are useful for the detection of more shallow components.

The remainder of this paper is organised in two main sections. Section 2. describes Grammar Learning by Partition Search. Section 3. reports experiments and results for NP identification and NP chunking.

2. Learning PCFGs by Partition Search

Partition Search Grammar Learning starts from the idea that new context-free grammars can be created from old simply by modifying the nonterminal sets, *merging* and *splitting* subsets of nonterminals. For example, for certain parsing tasks it is useful to *split* a single verb phrase category into verb phrases that are headed by a modal verb and those that are not, whereas for other parsing tasks, the added grammar complexity is avoidable. In another context, it may not be necessary to distinguish noun phrases in subject position from first objects and second objects, making it possible to *merge* the three categories into one.

The usefulness of such split and merge operations can be objectively measured by their effect on a grammar's size (number of rules and nonterminals) and performance (parsing accuracy on a given task). Grammar Learning by Partition Search automatically tries out different combinations of merge and split operations and therefore can automatically optimise a grammar's size and performance.

2.1. Preliminary definitions

Definition 1 Set Partition

A partition of a nonempty set A is a subset Π of 2^A such that $\emptyset$ is not an element of Π and each element of A is in one and only one set in Π.

The partition of A where all elements are singleton sets is called the *trivial partition* of A.

Definition 2 Probabilistic Context-Free Grammar

A Probabilistic Context-Free Grammar (PCFG) is a 4-tuple (W, N, N^S, R), where W is a set of terminal symbols, N is a set of nonterminal symbols, $N^S = \{(s_1, p(s_1)), \ldots (s_l, p(s_l))\}, \{s_1, \ldots s_l\} \subseteq N$ is a set of start symbols with associated probabilities summing to one, and $R = \{(r_1, p(r_1)), \ldots (r_m, p(r_m))\}$ is a set of rules with associated probabilities. Each rule r_i is of the form $n \rightarrow \alpha$, where n is a nonterminal, and α is a string of terminals and nonterminals. For each nonterminal n, the values of all $p(n \rightarrow \alpha_i)$ sum to one, or: $\sum_{i:(n \rightarrow \alpha_i, p(n \rightarrow \alpha_i)) \in R} p(n \rightarrow \alpha_i) = 1$.

2.2. Generalising and Specialising PCFGs through Nonterminal Set Operations

2.2.1. Nonterminal merging

Consider two PCFGs G and G':

$G = (W, N, N^S, R)$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N =$ { S, NP-SUBJ, VP, NP-OBJ }
$N^S =$ { (S, 1) }
$R =$ { (S -> NP-SUBJ VP, 1),
(NP-SUBJ -> NNS, 0.5),
(NP-SUBJ -> DET NN, 0.5),
(VP -> VBD NP-OBJ, 1),
(NP-OBJ -> NNS, 0.75),
(NP-OBJ -> DET JJ NNS, 0.25) }

$G' = (W, N', N^S, R')$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N' =$ { S, NP, VP }
$N^S =$ { (S, 1) }
$R' =$ { (S -> NP VP, 1),
(NP -> NNS, 0.625),
(NP -> DET NN, 0.25),
(VP -> VBD NP, 1),
(NP -> DET JJ NNS, 0.125) }

Intuitively, to derive G' from G, the two nonterminals NP-SUBJ and NP-OBJ are merged into a single new nonterminal NP. This merge results in two rules from R becoming identical in R': both NP-SUBJ -> NNS and NP-OBJ -> NNS become NP -> NNS. One way of determining the probability of the new rule NP -> NNS is to sum the probabilities of the old rules and renormalise by the number of nonterminals that are being merged[1]. In the above example therefore $p(\text{NP -> NNS}) = (0.5 + 0.75)/2 = 0.625$[2].

An alternative would be to reestimate the new grammar on some corpus, but this is not appropriate in the current context: merge operations are used in a search process (see below), and it would be expensive to reestimate each new candidate grammar derived by a merge. It is better to use any available training data to estimate the original grammar's probabilities, then the probabilities of all derived grammars can simply be calculated as described above without expensive corpus reestimation.

The new grammar G' derived from an old grammar G by merging nonterminals in G is a generalisation of G: the language of G', or $L(G')$, is a superset of the language of G, or $L(G)$. E.g., det jj nns vbd det jj nns is in $L(G')$ but not in $L(G)$. The set of parses assigned to a sentence s by G' differs from the set of parses assigned to s by G. The probabilities of parses for s can change, and so can the probability ranking of the parses, i.e. the most likely parse for s under G may be different from the most likely parse for s under G'. Finally, G' has the same number of rules as G or fewer.

[1] Reestimating the probabilities on the training corpus would of course produce identical results.

[2] Renormalisation is necessary because the probabilities of all rules expanding the same nonterminal sum to one, therefore the probabilities of all rules expanding a new nonterminal resulting from merging n old nonterminals will sum to n.

2.2.2. Nonterminal splitting

Deriving a new PCFG from an old one by splitting nonterminals in the old PCFG is not quite the exact reverse of deriving a new PCFG by merging nonterminals. The difference lies in determining probabilities for new rules. Consider the following grammars G and G':

$G = (W, N, N^S, R)$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N =$ { S, NP, VP }
$N^S =$ { (S, 1) }
$R =$ { (S -> NP VP, 1),
(NP -> NNS, 0.625),
(NP -> DET NN, 0.25),
(VP -> VBD NP, 1),
(NP -> DET JJ NNS, 0.125) }

$G' = (W, N', N^S, R')$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N' =$ { S, NP-SUBJ, VP, NP-OBJ }
$N^S =$ { (S, 1) }
$R' =$ { (S -> NP-SUBJ VP, ?),
(S -> NP-OBJ VP, ?),
(NP-SUBJ -> NNS, ?),
(NP-SUBJ -> DET NN, ?),
(NP-SUBJ -> DET JJ NNS, ?) }
(VP -> VBD NP-SUBJ, ?),
(VP -> VBD NP-OBJ, ?),
(NP-OBJ -> NNS, ?),
(NP-OBJ -> DET NN, ?),
(NP-OBJ -> DET JJ NNS, ?) }

To derive G' from G, the single nonterminal NP is split into two nonterminals NP-SUBJ and NP-OBJ. This split results in several new rules. For example, for the old rule NP -> NNS, there now are two new rules NP-SUBJ -> NNS and NP-OBJ -> NNS. One possibility for determining the new rule probabilities is to redistribute the old probability mass evenly among them, i.e. $p(\text{NP -> NNS}) = p(\text{NP-SUBJ -> NNS}) = p(\text{NP-SUBJ -> NNS})$. However, then there would be no benefit at all from performing such a split: the resulting grammar would be larger, the most likely parses remain unchanged, and for each parse p under G that contains a nonterminal participating in a split operation, there would be at least two equally likely parses under G'.

The new probabilities cannot be calculated directly from G. The redistribution of the probability mass has to be motivated from a knowledge source outside of G. One way to proceed is to estimate the new rule probabilities on the original corpus — provided that it contains the information on the basis of which a split operation was performed in extractable form. For the current example, a corpus in which objects and subjects are annotated could be used to estimate the probabilities of the rules in G', and might yield the following result (which reflects the fact that in English, the NP in a sentence NP VP is usually a subject, whereas the NP in a VP consisting of a verb followed by an NP is an object):

$G' = (W, N', N^S, R')$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N' =$ { S, NP-SUBJ, VP, NP-OBJ }
$N^S =$ { (S, 1) }
$R' =$ {
(S -> NP-SUBJ VP, 1),
(S -> NP-OBJ VP, 0),
(NP-SUBJ -> NNS, 0.5),
(NP-SUBJ -> DET NN, 0.5),
(NP-SUBJ -> DET JJ NNS, 0) }
(VP -> VBD NP-SUBJ, 0),
(VP -> VBD NP-OBJ, 1),
(NP-OBJ -> NNS, 0.75),
(NP-OBJ -> DET NN, 0),
(NP-OBJ -> DET JJ NNS, 0.25) }

With rules of zero probability removed, G' is identical to the original grammar G in the example in the previous section.

2.3. Partition Search

A PCFG together with nonterminal merge and split operations defines a space of derived grammars which can be searched for a new PCFG that optimises some given objective function. The disadvantage of this search space is that it is infinite, and each split operation requires the reestimation of rule probabilities from a training corpus, making it computationally much more expensive than a merge operation.

However, there is a simple way to make the search space finite, and at the same time to make split operations redundant. The resulting method, Grammar Learning by Partition Search, is summarised in this section (Partition Search is described in more detail, including formal definitions and algorithmic details, in Belz (2002)).

2.3.1. PCFG Partitioning

An arbitrary number of merges can be represented by a partition of the set of nonterminals. For the example presented in Section 2.2.1. above, the partition of the nonterminal set N in G that corresponds to the nonterminal set N' in G' is { {S}, {NP-SBJ, NP-OBJ}, {VP} }. The original grammar G together with a partition of its nonterminal set fully specifies the new grammar G': the new rules and probabilities, and the entire new grammar G' can be derived from the partition together with the original grammar G. The process of obtaining a new grammar G', given a base grammar G and a partition of the nonterminal set N of G will be called PCFG Partitioning[3].

2.3.2. Search space

The search space for Grammar Learning by Partition Search can be made finite and searchable entirely by merge operations (grammar partitions).

Making the search space finite: The number of merge operations that can be applied to a nonterminal set is finite, because after some finite number of merges there remains only one nonterminal. On the other hand, the number of split operations that can sensibly be applied to a nonterminal *NT* has an upper bound in the number of different terminals strings dominated by *NT* in a corpus of evidence (e.g. the corpus the PCFG was trained on). For example, when splitting the nonterminal NP into subjects and objects, there would be no point in creating more new nonterminals than the number of different subjects and objects found in the corpus.

Given these (generous) bounds, there is a finite number of distinct grammars derivable from the original grammar by different combinations of merge and split operations. This forms the basic space of candidate solutions for Grammar Learning by Partition Search.

Making the search space searchable by grammar partitioning only: Imposing an upper limit on the number and kind of split operations permitted not only makes the search space finite but also makes it possible to directly derive this *maximally split nonterminal set* (Max Set). Once the Max Set has been defined, the single grammar corresponding to it — the *maximally split Grammar* (Max Grammar) — can be derived and retrained on the training corpus.

The set of points in the search space corresponds to the set of partitions of the Max Set. Search for an optimal grammar can thus be carried out directly in the partition space of the Max Grammar.

Structuring the search space: The finite search space can be given hierarchical structure as shown in Figure 1 for an example of a very simple base nonterminal set {NP, VP, PP}, and a corpus which contains three different NPs, three different VPs and two different PPs.

At the top of the graph is the Max Set. The sets at the next level down (level 7) are created by merging pairs of nonterminals in the Max Set, and so on for subsequent levels. At the bottom is the *maximally merged nonterminal set* (Min Set) consisting of a single nonterminal *NT*. The sets at the level immediately above it can be created by splitting *NT* in different ways. The sets at level 2 are created from those at level 1 by splitting one of their elements. The original nonterminal set ends up somewhere in between the top and bottom (at level 3 in this example).

While this search space definition results in a finite search space and obviates the need for the expensive split operation, the space will still be vast for all but trivial corpora. In Section 3.3. below, alternative ways for defining the Max Set are described that result in much smaller search spaces.

2.3.3. Search task and evaluation function

The input to the Partition Search procedure consists of a base grammar G_0, a base training corpus C, and a task-specific training corpus D^T. G_0 and C are used to create the Max Grammar G. The **search task** can then be defined as follows:

> Given the maximally split PCFG $G = (W, N, N^S, R)$, a data set of sentences D, and a set of target parses D^T for D, find a partition Π_N of N that derives a grammar $G' = (W, \Pi_N, N^{S'}, R')$, such that $|R'|$ is minimised, and $f(G', D, D^T)$ is maximised, where f scores the performance of G' on D as compared to D^T.

[3]The concept of context-free grammar partitioning in this paper is not directly related to that in (Korenjak, 1969; Weng and Stolcke, 1995), and later publications by Weng et al. In these previous approaches, a non-probabilistic CFG's *set of rules* is partitioned into subsets of rules. The partition is drawn along a specific nonterminal NT, which serves as an interface through which the subsets of rules (hence, subgrammars) can communicate after partition (one grammar calling the other).

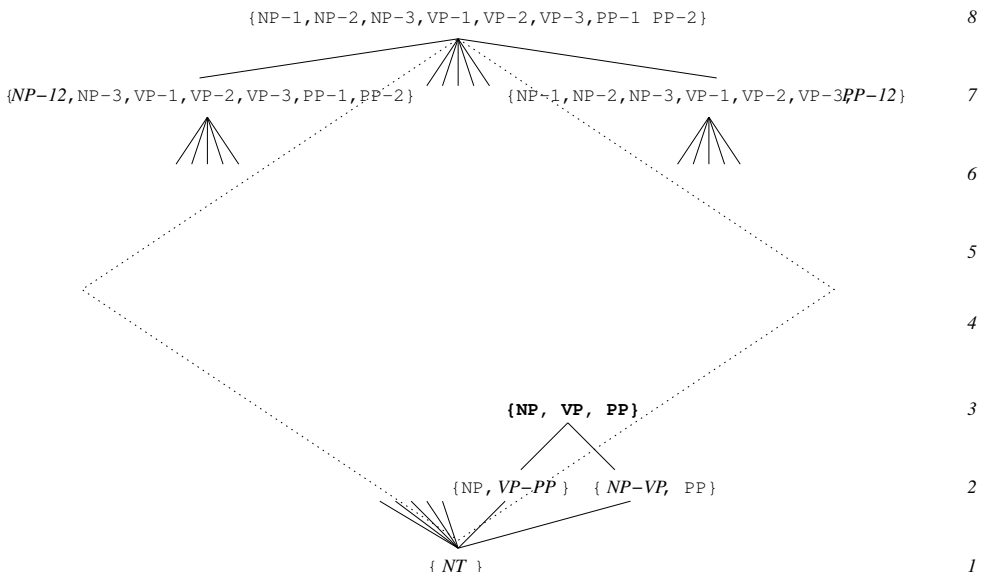


Figure 1: Simple example of a partition search space.

The size of the nonterminal set and hence of the grammar decreases from the top to the bottom of the search space. Therefore, if the partition space is searched top-down, grammar size is minimised automatically and does not need to be assessed explicitly.

In the current implementation, the **evaluation function** f simply calculates the F-Score achieved by a candidate grammar on D as compared to D^T. The F-Score is obtained by combining the standard PARSEVAL evaluation metrics *Precision* and *Recall*[4] as follows: $2 \times Precision \times Recall/(Precision + Recall)$.

An existing parser[5] was used to obtain Viterbi parses. If the parser failed to find a complete parse for a sentence, a simple grammar extension method was used to obtain partial parses instead (based on Schmid and Schulte im Walde (2000, p. 728)).

2.3.4. Search algorithm

Since each point in the search space can be accessed directly by applying the corresponding nonterminal set partition to the Max Grammar, the search space can be searched in any direction by any search method using partitions to represent candidate grammars.

In the current implementation, a variant of beam search is used to search the partition space top down. A list of the n current best candidate partitions is maintained (initialised to the Max Set). For each of the n current best partitions a random subset of size b of its children in the hierarchy is generated and evaluated. From the union of current best partitions and the newly generated candidate partitions, the n best elements are selected and form the new current best set. This process is iterated until either no new partitions can be generated that are better than their parents, or the lowest level of the partition tree is reached. In each iteration the size of the nonterminal set (partition) decreases by one.

The size of the search space grows exponentially with the size i of the Max Set. However, the complexity of the Partition Search algorithm is only $O(nbi)$, because only up to $n \times b$ partitions are evaluated in each of up to i iterations[6].

3. Learning NP Extraction Grammars

3.1. Data and Parsing Tasks

Sections 15–18 of WSJC were used for deriving the base grammar and as the base training corpus, and different randomly selected subsets of Section 1 from the same corpus were used as task-specific training corpora during search. Section 20 was used for final performance tests.

Results are reported in this paper for the following two parsing tasks. In **NP identification** the task is to identify in the input sentence all noun phrases[7], nested and otherwise, that are given in the corresponding WSJC parse. **NP chunking** was first defined by (Abney, 1991), and involves the identification of flat noun phrase chunks. Target parses were derived from WSJC parses by an existing conversion procedure[8].

The Brill Tagger was used for POS tagging testing data, and achieved an average accuracy of 97.5% (as evaluated by `evalb`).

3.2. Base grammar

A simple treebank grammar[9] was derived from Sections 15–18 of the WSJ corpus by the following procedure:

1. Iteratively edit the corpus by deleting (i) brackets and labels that correspond to empty category expansions; (ii) brackets

[4] I used the `evalb` program by Sekine and Collins (`http://cs.nyu.edu/cs/projects/proteus/evalb/`) to obtain Precision and Recall figures.

[5] LoPar (Schmid, 2000) in its non-head-lexicalised mode. Available from `http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/LoPar-en.html`.

[6] As before, n is the number of current best candidate solutions, b is the width of the beam, and i is the size of the Max Set.

[7] Corresponding to the WSJC categories `NP, NX, WHNP` and `NAC`.

[8] Devised by Erik Tjong Kim Sang for the TMR project *Learning Computational Grammars*.

[9] The term was coined by Charniak (1996).

and labels containing a single constituent that is not labelled with a POS-tag; (iii) cross-indexation tags; (iv) brackets that become empty through a deletion.

2. Convert each remaining bracketting in the corpus into the corresponding production rule.
3. Collect sets of terminals W, nonterminals N and start symbols N^S from the corpus. Probabilities p for rules $n \rightarrow \alpha$ are calculated from the rule frequencies C by Maximum Likelihood Estimation: $p(n \rightarrow \alpha) = \frac{C(n \rightarrow \alpha)}{\sum_i C(n \rightarrow \alpha^i)}$.

This procedure creates the base grammar *BARE* which has $10,118$ rules and 147 nonterminals.

3.3. Restricting the search space further

The simple method described in Section 2.3.2. for defining the maximally split nonterminal set (Max Set) tends to result in vast search spaces. Using parent node (PN) information to create the Max Set is much more restrictive and linguistically motivated. The Max Grammar *PN* used in the experiments reported below can be seen as making use of *Local Structural Context* (Belz, 2001): the independence assumptions inherent in PCFGs are weakened by making the rules' expansion probabilities dependent on part of their immediate structural context (here, its parent node). To obtain the grammar *PN*, the base grammar's nonterminal set is maximally split on the basis of the *parent node* under which rules are found in the base training corpus[10]. Several previous investigations have demonstrated improvement in parsing results due to the inclusion of parent node information (Charniak and Carroll, 1994; Johnson, 1998; Verdú-Mas et al., 2000).

Another possibility is to use the base grammar *BARE* itself as the Max Grammar. This is a very restrictive search space definition and amounts to an attempt to optimise the base grammar in terms of its size and its performance on a given task without adding any information. Results are given below for both *BARE* and *PN* as Max Grammars.

In the current implementation of the algorithm, the search space is reduced further by avoiding duplicate partitions, and by only allowing merges of nonterminals that have the same phrase prefix `NP-*, VP-*` etc.

The Max Grammars end up having sets of nonterminals that differ from the bracket labels used in the WSJC: while the phrase categories (e.g. `NP`) are the same, the tags (e.g. `*-S, *-3`) on the phrase category labels may differ. In the evaluation, all labels starting with the same phrase category prefix are considered equivalent.

3.4. NP chunking results

Baseline Results. Base grammar *BARE* (see Section 3.2. achieves an F-Score of 88.25 on the NP chunking task. This baseline result compares as follows with existing results:

	NP chunking
Chunk Tag Baseline	79.99
Grammar *BARE*	88.25
Current Best: nonlexicalised	90.12
lexicalised	93.25 (93.86)

The chunk tag baseline F-Score is the standard baseline for the NP chunking task and is obtained by tagging each POS tag in a sentence with the label of the phrase that it most frequently appears in, and converting these phrase tags into labelled brackettings (Nerbonne et al., 2001, p. 102). The best nonlexicalised result was achieved with the decision-tree learner C5.0 (Tjong Kim Sang et al., 2000), and the current overall best result for NP chunking is for memory-based learning and a lexicalised chunker (Tjong Kim Sang et al., 2000)[11].

Table 1 shows results for Partition Search applied to the NP chunking task. The first column shows the Max Grammar used in a given batch of experiments. The second column indicates the type of result, where the Max Grammar result is the F-Score, grammar size and number of nonterminals of the Max Grammar itself, and the remaining results are the average and single best results achieved by Partition Search. The third and fourth columns show the number of iterations and evaluations carried out before search stopped. Columns 5–8 show details of the final solution grammars: column 5 shows the evaluation score on the training data, column 6 the overall F-Score on the testing data, column 7 the size, and the last column gives the number of nonterminals.

The best result (boldface) was an F-Score of 90.24% (compared to the base result of 88.25%), and 95 nonterminals (147 in the base grammar), while the number of rules increased from 10,118 to 11,972. This result improves the general baseline by 12.7% and the performance by grammar *BARE* by 2.2%. It also outperforms the best existing result of 90.12% for nonlexicalised NP chunking by a small margin.

3.5. NP identification results

Baseline Results. Base grammar *BARE* achieves an F-Score of 79.29 on the NP identification task. This baseline result compares as follows with existing results:

	NP identification
Chunk Tag Baseline	67.56
Grammar *BARE*	79.29
Current Best: nonlexicalised	80.15
lexicalised	83.79

All results in this table (except for that for grammar *BARE*) are reported in Nerbonne et al. (2001, p. 103). The task definition used there was slightly different in that it omitted two minor NP categories (WSJC brackets labelled `NAC` and `NX`). The slightly different task definition has only a very small effect on F-Scores, so the above results are comparable. The chunk tag baseline F-Score was again obtained by tagging each POS tag in a sentence with the label of the phrase that it most frequently appears in. The best lexicalised result was achieved with a cascade of memory-based learners. The same paper also included two results for nonlexicalised NP identification.

Table 2 (same format as Table 1) contains results for Partition Search and the NP identification task. The smallest nonterminal set had 63 nonterminals (147 in the base

[10] The parent node of a phrase is the category of the phrase that immediately contains it.

[11] Nerbonne et al. (2001) report a slightly better result of 93.86 achieved by combining seven different learning systems.

Max Grammar		Iter.	Eval.	F-Score (subset)	F-Score (WSJC S 1)	Size (rules)	Nonterms
BARE	Max Grammar result:				88.25	10,118	147
	Average:	116.8	2,749.6	89.64	88.57	7,849.6	32.2
	Best (size):	119	2,806	89.79	88.51	7,541	30
	Best (F-score):	114	2,674	87.93	88.70	7,777	35
PN	Max Grammar result:				89.86	16,480	970
	Average:	526	13,007.75	94.85	89.83	14,538.25	446
	Best (size and F-score):	**877**	**21,822**	**93.85**	**90.24**	**11,972**	**95**

Table 1: Partition tree search results for NP chunking task, WSJC Section 1 (averaged over 5 runs, variable parameters: $x = 50$, $b = 5$, $n = 5$).

Max Grammar		Iter.	Eval.	F-Score (subset)	F-Score (WSJC S 1)	Size (rules)	Nonterms
BARE	Max Grammar result:				79.29	10,118	147
	Average	111.4	2,629	87.831	79.10	8,655	37.6
	Best (size):	113	2,679	86.144	78.9	8,374	36
	Best (F-score):	114	2,694	90.246	79.51	8,541	41
PN	Max Grammar result:				82.01	16,480	970
	Average:	852.6	21,051	91.2098	81.41308	13,202.8	119.4
	Best (size):	909	22,474	91.881	80.9830	12,513	63
	Best (F-score):	**658**	**16,286**	**89.572**	**82.0503**	**15,305**	**314**

Table 2: Partition tree search results for NP identification task, WSJC Section 1 (averaged over 5 runs, variable parameters: $x = 50$, $b = 5$, $n = 5$).

grammar). The best result (boldface) was an F-Score of 82.05% (base result was 79.29%), while the number of rules increased from 10,118 to 15,305. This improves the general baseline by 21.45% and grammar *BARE* by 3.48%. It also outperforms the other two results for nonlexicalised NP chunking by a significant margin, and even comes close to the best lexicalised result (83.79%).

3.6. General comments

Partition Search is able to reduce grammar size by merging groups of nonterminals (hence groups of rules) that do not need to be distinguished for a given task. It is able to improve parsing performance firstly by grammar generalisation (partitioned grammars parse a superset of the sentences parsed by the base grammar), and secondly by reranking parse probabilities (the most likely parse for a sentence under a partitioned grammar can differ from its most likely parse under the base grammar).

The margins of improvement over baseline results were bigger for the NP identification task than for NP chunking. The results reported here for NP chunking are no match for the best lexicalised results, whereas the results for NP identification come close to the best lexicalised results. This indicates that the two characteristics that most distinguish the grammars used here from other approaches — some non-shallow structural analysis and parent node information — are more helpful for NP identification.

Preliminary tests revealed that results were surprisingly constant over different combinations of variable parameter values, although training subset size of less then 50 meant unpredictable results for the complete WSJC Section 1. For a random subset of size 50 and above, there is an almost complete correspondence between subset F-Score and Section 1 F-Score, i.e. higher subset F-Score almost always means higher Section 1 F-Score.

The results presented in the previous section also show what happens if Partition Search is used as a grammar compression method (when existing grammars are used as Max Grammars). In Table 1, for example, when applied to the base grammar *BARE* (four top rows), it maximally reduces the number of nonterminals from 147 to 30 and the number of rules from $10,118$ to $7,541$, while *improving* the overall F-Score. The size reductions on the *PN* grammar are even bigger: 970 nonterminals down to 95, and $16,480$ rules down to $11,972$, again with a slight improvement in the F-Score (even though on average, the F-Score remained about the same). Unlike other grammar compression methods (Charniak, 1996; Krotov et al., 2000), Partition Search achieves lossless compression, in the sense that the compressed grammars are guaranteed to be able to parse all of the sentences parsed by the original grammar.

Compared to other approaches using parent node information (Charniak and Carroll, 1994; Johnson, 1998; Verdú-Mas et al., 2000), the approach presented here has the advantage of being able to select a subset of all parent node information on the basis of its usefulness for a given parsing task. This saves on grammar complexity, hence parsing cost.

3.7. Nonterminal distinctions preserved/eliminated

The base grammar *BARE* has 26 different phrase category prefixes (S, NP, etc.). The additional tags encoding grammatical function and parent node information results in much larger numbers of nonterminals. One of the aims

of partition search is to reduce this number, preserving only useful distinctions. This section looks at nonterminal distinctions that were preserved and eliminated for each task and grammar.

3.7.1. Base grammar *BARE* (functional tags only)

Twelve of the 26 phrase categories are not annotated with functional tags in the WSJC. The remaining 14 phrase categories have between 2 and 28 grammatical function subcategories[12].

In the *BARE* grammar, more nonterminals were merged on average in the NP chunking task (32.2 remaining) than in the NP identification task (37.6 remaining). This is as might be expected since the NP identification task looks the more complex.

Results for NP chunking show a very strong tendency to merge the subcategories of all phrase categories except for two: NP and PP. With only the rare exception, the distinction between different grammatical functions is eliminated for the other 12 out of 14 phrase categories. By contrast, for NP, between 2 and 5 different categories remain (average 2.8), and for PP, between 2 and 4 remain (average 3.6). This implies that for NP chunking only the different grammatical functions of NPs and PPs are useful.

Results for NP identification show a tendency to perserve distinctions among the subcategories of SBAR, NP and PP and to a lesser extent among those of ADVP and ADJP. Other distinctions tend to be eliminated. All subcategories of `SBARQ`, `NX`, `NAC`, `INTJ` and `FRAG` are always merged, `UCP` and `SINV` nearly always.

3.7.2. Grammar *PN* (parent node tags)

The *PN* grammar has 970 phrase subcategories for the 26 basic phrase categories of which only those with the largest numbers of subcategories are examined here: NP (173), PP (173), ADVP (118), S (76), and VP (62).

Surprisingly, far fewer nonterminals were merged on average in the NP chunking task (446 remaining) than in the NP identification task (only 119.4 remaining).

In both tasks, although more so in the NP chunking task, the strongest tendency was that far more NP subcategories were preserved than any other.

In the NP identification task, the different NAC and NX subcategories were always merged into a single one, whereas in the NP chunking task, at least 4 different NAC and 3 different NX subcategories remained.

In both tasks equally, ADVP and PP distinctions were mostly eliminated. The same goes for VP distinctions although VPs with parent node S, SBAR and VP had a higher tendency to remain unmerged.

These results indicate that by far the most important parent node information for both NP identification and chunking are the parent nodes of the NPs themselves. More detailed analysis of merge sets would be needed to see what exactly this means.

[12] `ADJP`: 6, `ADVP`: 18, `FRAG`: 2, `INTJ`: 2, `NAC`: 4, `NP`: 23, `NX`: 2, `PP`: 28, `S`: 14, `SBAR`: 20, `SBARQ`: 3, `SINV`: 2, `UCP`: 8, `VP`: 3.

4. Conclusions and Further Research

Grammar Learning by Partition Search was shown to be an efficient method for constructing PCFGs optimised for a given parsing task. In the nonlexicalised applications reported in this paper, the performance of the base grammar was improved by up to 3.48%. This corresponds to an improvement of up to 21.45% over the standard baseline. The result for NP chunking is slightly better than the best existing result for nonlexicalised NP chunking, whereas the result for NP identification closely matches the best existing result for lexicalised NP identification.

Partition Search can also be used to simply reduce grammar size, if an existing grammar is used as the Max Grammar. In the experiments reported in this paper, Partition Search reduced the size of nonterminal sets by up to 93.5%, and the size of rule sets by up to 27.4%. Compared to other grammar compression techniques, it has the advantage of being lossless.

Further research will look at additionally incorporating lexicalisation, other search methods, and other variable parameter combinations.

5. Acknowledgements

The research reported in this paper was in part funded under the European Union's TMR programme (Grant No. ERBFMRXCT980237).

6. References

Steven Abney. 1991. Parsing by chunks. In R. Berwick, S. Abney, and C. Tenny, editors, *Principle-Based Parsing*, pages 257–278. Kluwer Academic Publishers, Boston.

A. Belz. 2001. Optimising corpus-derived probabilistic grammars. In *Proceedings of Corpus Linguistics 2001*, pages 46–57.

A. Belz. 2002. Grammar learning by partition search. In *Proceedings of LREC Workshop on Event Modelling for Multilingual Document Linking*.

Eugene Charniak and Glenn Carroll. 1994. Context-sensitive statistics for improved grammatical language models. Technical Report CS-94-07, Department of Computer Science, Brown University.

Eugene Charniak. 1996. Tree-bank grammars. Technical Report CS-96-02, Department of Computer Science, Brown University.

Mark Johnson. 1998. PCFG models of linguistic tree representations. *Computational Linguistics*, 24(4):613–632.

A. J. Korenjak. 1969. A practical method for constructing LR(k) processors. *Communications of the ACM*, 12(11).

A. Krotov, M. Hepple, R. Gaizauskas, and Y. Wilks. 2000. Evaluating two methods for treebank grammar compaction. *Natural Language Engineering*, 5(4):377–394.

J. Nerbonne, A. Belz, N. Cancedda, Hervé Déjean, J. Hammerton, R. Koeling, S. Konstantopoulos, M. Osborne, F. Thollard, and E. Tjong Kim Sang. 2001. Learning computational grammars. In *Proceedings of CoNLL 2001*, pages 97–104.

H. Schmid and S. Schulte Im Walde. 2000. Robust German noun chunking with a probabilistic context-free grammar. In *Proceedings of COLING 2000*, pages 726–732.

H. Schmid. 2000. LoPar: Design and implementation. Bericht des Sonderforschungsbereiches "Sprachtheoretische Grundlagen für die Computerlinguistik" 149, Institute for Computational Linguistics, University of Stuttgart.

E. Tjong Kim Sang, W. Daelemans, H. Déjean, R. Koeling, Y. Krymolowski, V. Punyakanok, and D. Roth. 2000. Applying system combination to base noun phrase identification. In *Proceedings of COLING 2000*, pages 857–863.

Jose Luis Verdú-Mas, Jorge Calera-Rubio, and Rafael C. Carrasco. 2000. A comparison of PCFG models. In *Proceedings of CoNLL-2000 and LLL-2000*, pages 123–125.

F. L. Weng and A. Stolcke. 1995. Partitioning grammars and composing parsers. In *Proceedings of the 4th International Workshop on Parsing Technologies*.

An integration of Vector-Based Semantic Analysis and Simple Recurrent Networks for the automatic acquisition of lexical representations from unlabeled corpora

Fermín Moscoso del Prado Martín*, **Magnus Sahlgren**†

*Interfaculty Research Unit for Language and Speech (IWTS)
University of Nijmegen & Max Planck Institute for Psycholinguistics
P.O. Box 310, NL-6500 AH Nijmegen, The Netherlands
fermin.moscoso-del-prado@mpi.nl

†Swedish Institute for Computer Science (SICS)
Box 1263, SE-164 29 Kista, Sweden
mange@sics.se

Abstract

This study presents an integration of Simple Recurrent Networks to extract grammatical knowledge and Vector-Based Semantic Analysis to acquire semantic information from large corpora. Starting from a large, untagged sample of English text, we use Simple Recurrent Networks to extract morpho-syntactic vectors in an unsupervised way. These vectors are then used in place of random vectors to perform Vector-Space Semantic Analysis. In this way, we obtain rich lexical representations in the form of high-dimensional vectors that integrate morpho-syntactic and semantic information about words. Apart from incorporating data from the different levels, we argue how these vectors can be used to account for the particularities of each different word token of a given word type. The amount of lexical knowledge acquired by the technique is evaluated both by statistical analyses comparing the information contained in the vectors with existing 'hand-crafted' lexical resources such as CELEX and WordNet, and by performance in language proficiency tests. We conclude by outlining the cognitive implications of this model and its potential use in the bootstrapping of lexical resources

1. Introduction

Collecting word-use statistics from large text corpora has proven to be a viable method for automatically acquiring knowledge about the structural properties of language. The perhaps most well-known example is the work of George Zipf, who, in his famous *Zipf's laws* (Zipf, 1949), demonstrated that there exist fundamental statistical regularities in language. Although the useability of statistics for extracting structural information has been widely recognized, there has been, and still is, much scepticism regarding the possibility of extracting semantic information from word-use statistics. We believe that part of the reason for this scepticism is the conception of meaning as something external to language — as something *out there* in the world, or as something *in here* in the mind of a language user. However, if we instead adopt what we may call a "Wittgensteinian" perspective, in which we do not demand any rigid definitions of word meanings, but rather characterize them in terms of their use and their "family resemblance" (Wittgenstein, 1953), we may argue that word-use statistics provide us with exactly the right kind of data to facilitate semantic knowledge acquisition. The idea, first explicitly stated in Harris (1968), is that the meaning of a word is related to its distributional pattern in language. This means that if two words frequently occur in similar context, we may assume that they have similar meanings. This assumption is known as "the Distributional Hypothesis," and it is the ultimate rationale for statistical approaches to semantic knowledge acquisition, such as Simple Recurrent Networks or Vector-Based Semantic Analysis.

1.1. Simple Recurrent Networks

Simple Recurrent Networks (SRN; Elman, 1990) are a class of Artificial Neural Networks consisting of the three traditional 'input', 'hidden' and 'output' layers of units, to which one additional layer of 'context' units is added. The basic architecture of an SRN is shown in Figure 1. The outputs of the 'context' units are connected to the inputs of the 'hidden' layer as if they formed and additional 'input' layer. However instead of receiving their activation from outside, the activations of the 'context' layer at time step n are a copy of the activations of the 'hidden' layer at time step $n-1$. This is achieved by adding simple, one-to-one 'copy-back' connections from the 'hidden' layer into the 'context' layer. In contrast to all the other connections in the network, these are special in that they are not trained (their weights are fixed at 1), and in that they perform a raw copy operation from a hidden unit into a context unit, that is to say, they employ the identity function as the activation function. Networks of this kind combine the advantages of recurrent networks, their capability of maintaining a history of past events, with the simplicity of multilayer perceptrons as they can be trained by the backpropagation algorithm.

Elman (1993) trained an SRN on predicting the next word in a sequence of words, using sentences generated by an artificial grammar, with a very limited vocabulary (24 words). He showed that a network of this class, when trained on a word prediction task and given the right training strategy (see (Rohde and Plaut, 2001) for further discussion of this issue), acquired various grammatical properties such as verbal inflection , plural inflection of nouns, argumental structure of verbs or grammatical category. More-

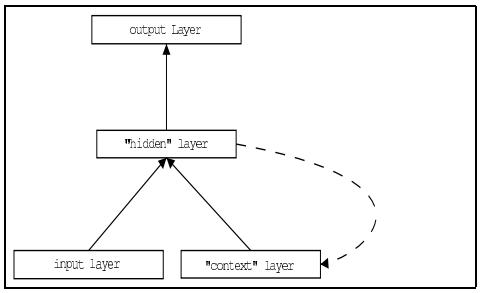


Figure 1: Modular architecture of a Simple Recurrent Network. The boxes correspond to layers of units. The solid arrows represent sets of trainable 'all-to-all' connections between the units in two layers. The dashed arrow stands a for fixed 'one-to-one' not trainable connection between two layers. These connections have the function of copying the activation of the hidden units into the context units at every time step.

over, the activations of the hidden units of the network provided detailed, token-specific characterizations of the morpho-syntactic properties of a word. Moscoso del Prado and Baayen (2001) showed how this method can be extended to deal with the large vocabulary sizes of realistic corpora. They trained a network akin to those of (Rohde and Plaut, 1999; Elman, 1993) on a word prediction task, using moderately large corpora of written English and Dutch (approximately 700,000 tokens of English and 4,500,000 of Dutch). The hidden units again provided rich representations of the morpho-syntactic properties of the words, containing information ranging from grammatical category, to subtle inflectional details such as verbal inflection or adjective gender. Moreover, the network had also captured some semantic properties of words, namely semantic properties that can be inferred from syntactic properties such as argumental structure.

1.2. Vector-Based Semantic Analysis

While SRN's appear to be more sensitive towards syntactic features, vector-space models have been used for over a decade to acquire and represent semantic information about words, documents and other linguistic units. This is done by collecting co-occurrence information in a words-by-contexts matrix F where each row F_w represents a unique word and each column F_c represents a context, which can either be a multi-word segment such as a document or another word. Latent Semantic Analysis/Indexing (LSA/LSI; Deerwester et al., 1990; Landauer and Dumais, 1997) uses document-based co-occurrence statistics, while Hyperspace Analogue to Language (HAL; Lund et al., 1995) and Schutze (1992) use word-based statistics. The cells of the matrix indicate the (weighted and/or normalized) frequency of ocurrence in, or co-occurrence with, the co-occurrence context (i.e. documents or words). Vector-space models generally also use some form of dimension reduction to reduce the computational strains of dealing with the rather ungainly co-occurrence matrix. LSA uses Singular Value Decomposition (SVD) and HAL uses a "column variance method," which consists in discarding the columns with lowest variance. This reduces the dimensionality of the co-occurrence matrix to a fraction of its original size. linguistic units are thus represented in the final reduced matrix by semantic vectors of n dimensionality. LSA is reported to be optimal at $n = 300$ (Landauer and Dumais, 1997), HAL at $n = 200$ (Lund et al., 1995), while Schutze (1992) use $n = 20$. A different approach to create the vector-space is Random Indexing (Kanerva et al., 2000; Karlgren and Sahlgren, 2001), which avoids the inefficient and inflexible dimensionality reduction phase by using high-dimensional sparse, random *index vectors* to accumulate a words-by-contexts matrix in which words are represented by high-dimensional (i.e. n is in the order of thousands) *context vectors*.

Vector-space methodology has been empirically validated in a number of experiments as a viable technique for the automatic extraction of semantic information from raw, unstructured text data. For example, Landauer and Dumais (1997) report a result on a standardized vocabulary test (TOEFL; Test of English as a Foreign Language) that is comparable to the average performance of foreign (non-English speaking) applicants to U.S. colleges (64.4% vs. 64.5% correct answers to the TOEFL). Sahlgren (2001), showed that similar performance (64.5% – 67%) may be obtained by using distributed representations in the Random Indexing technique that eliminates the need for the computationally expensive SVD, and he also demonstrated that the performance may be further enhanced (72% correct answers) by taking advantage of explicit linguistic information (morphology). Further empirical evidence can be found in, for example, (Lund and Burgess, 1996), who used semantic vectors to model reaction times from lexical priming studies, and from (Landauer and Dumais, 1997), who used LSA for evaluating the quality of content of student essays on given topics. Thus, it appears to be beyond doubt that the vector-space methodology really is able to form high-quality semantic representations by using such a simple souce of information as plain co-occurrence statistics. In the remainder of this paper, we will use the label *Vector-Based Semantic Analysis* to denote the practice of using co-occurrence information to construct vectors representing linguistic units in a high-dimensional semantic space.

2. Goal of the paper

In this study, we integrate two techniques to automatically obtain distributed lexical representations from corpora encoding morpho-syntactic and semantic information simultaneously. A hybrid technique such as the one that we describe here has several advantages. First, it requires a minimum of preexisting lexical resources, as it depends only on raw corpora. There is no need for taggers or parsers which, for many languages, may be unavailable.

Second, in contrast to other approaches that exploit word co-occurrences, our method keeps computational costs under control, as we avoid having to deal with huge co-occurrence matrices and we do not need to apply dimensional reduction techniques such as Singular Value Decomposition or Principal Component Analysis. The use of such dimensional reduction techniques imposes important limi-

tations on the extension of existing resources, as the addition of a new item would requires that a new reduced similarity space is calculated. In contrast, both SRN and the VBSA technique allow for the direct inclusion of new data. Another important advantage of our approach is that lexical representations become dynamic in nature: each token of a given type will have a slightly different representation.

We produce explicit measures of reliability that are directly associated to each distance calculated by our method. This is particularly useful for extending existing lexical resources such as computational thesauri.

In what follows, we introduce the corpus employed in the experiment, together with the SRN and VBSA techniques that we used. We then evaluate the grammatical knowledge encoded in the distributed representations obtained by the model. We subsequently evaluate the semantic knowledge contained in the system by means of scores on language proficiency tests (TOEFL), comparison with synonyms in WordNet, and a comparison of the properties of morphological variants. We conclude by discussing the possible application of this technique to bootstrap lexical resources from untagged corpora and the cognitive implications of these results.

3. The Experiment

3.1. Corpus

For the training of the SRN network, we used the texts corresponding to the first 20% of the British National Corpus; by first we mean that we selected the files following the order of directories, and we included the first two directories in the corpus. This corresponds to roughly 20 million tokens. To allow for comparison with the results from (Sahlgren, 2001), which were based on a 10 million word corpus, only the first half of this subset was used in the application of the VSBA technique.

Only a naive preprocessing stage was performed on the original SGML files. This included removing all SGML labels from the corpus, converting all words to lower case, substituting all numerical tokens for a $[num]$ token and separating hyphenated compound words into three different tokens ($first\ word + [hyphen] + second\ word$). All tokens containing non alphabetic characters different from the common punctuation marks were removed from the corpus. Finally, to reduce the vocabulary size, all tokens that were below a frequency threshold of two, were substituted by an $[unknown]$ token.

3.2. Design and training of the SRN

The Simple Recurrent Network followed the basic design shown in Figure 1. We used a network with 300 units in the input and output layers, and 150 units in the hidden and context layers. To allow for representation of a very large number of tokens, we used the semi-localist approach described in (Moscoso del Prado and Baayen, 2001) with a code of three random active units per word. On the one hand, this approach is close to a traditional style one-bit-per-word localistic representation in that the vectors of two different words will be nearly orthogonal. The small deviation from full orthogonality between representations has an effect similar to the introduction of a small amount of random noise, which actually speeds up the learning process. On other the hand, using semi-distributed input/output representations allows us to represent a huge number of types (a maximum of $\binom{300}{3} = 4,455,100$ types), while keeping the size of the network moderately small.

The sentences of the corpus were grouped into 'examples' of five consecutive sentences. At each time step, a word was presented to the input layer and the network would be trained to predict the following word in the output units. The corpus sentences were presented word by word in the order in which they appear. After every five sentences (a full 'example'), the activation of the context units was reset to 0.5. Imposing limitations on the network's memory on the initial stages of training is a pre-requisite for the networks to learn long distance syntactic relations (Elman, 1993; cf., Rohde and Plaut, 2001; Rohde and Plaut, 1999). We implemented this 'starting small' strategy by introducing a small amount of random noise (0.15) in the output of the hidden units, and by gradually reducing to zero during training. At the same time that the random noise in the context units was being reduced, we also gradually reduced the learning rate, starting with a learning rate of 0.1 and finished training with a learning rate of 0.4. Throughout training, we used a momentum of 0.9.

Although the experiments in (Elman, 1993) used the traditional backpropagation algorithm, using the mean square error as the error measure to minimize, following (Rohde and Plaut, 1999) we substituted the training algorithm for a modified momentum descent using cross-entropy as our error measure,

$$\sum_i \left[t_i \log \left(\frac{t_i}{o_i} \right) + (1 - t_i) \log \left(\frac{1 - t_i}{1 - o_i} \right) \right] \qquad (1)$$

Modified momentum descent enables stable learning with very aggressive learning rates as the ones we use. The network was trained on the whole corpus of 20 million for one epoch using the *Light Efficient Network Simulator* (LENS; Rohde, 1999).

3.3. Application of VBSA technique

Once the SRN had been trained, we proceeded to apply the Vector Based Semantic Analysis technique. Sahlgren (2001) used what he called 'random labels'. These were sparse 1800 element vectors, in which, for a given word type, only a small set of randomly chosen elements would be active (± 1.0), while the rest would be inactive. Once these initial labels had been created, the corpus was processed in the following way. For each token in the corpus, the labels of the s immediately preceding or following tokens were added to the vector of the word (all vectors were initialized to a set of 0's). The addition would be weighted giving more importance to the closer word in the window. Words outside a frequency range of $(3 - 14,000)$ are not included in these sums. This range excludes both the very frequent types, typically function words, and the least frequent types, about which there is not enough information to provide reliable counts. Optimal results are obtained with a window size ($s = 3$), that is, by taking into account the three preceeding and following words to a given token.

In order to reduce sparsity, Sahlgren used a lemmatizer to unify tokens representing inflectional variants of the same root. Sahlgren had also observed that the inclusion of explicit syntactic information extracted by a parser did not improve the results, but led to lower performance. We believe that this can be partly due to the *static* character of the syntactic information that was used. We therefore use a *dynamic* coding of syntactic information, which is more sensitive to the subtle changes in grammatical properties of each different instance of a word.

In our study, we substituted the knowledge-free random labels of (Sahlgren, 2001) by the dynamic context-sensitive representations of the individual tokens as coded in the patterns of activations of our SRN. Thus each type is represented by a slightly different vector for each different grammatical context in which it appears. To obtain these representation, we presented the text to the SRN and used the activation of the hidden units to provide the dynamic labels for VBSA

We then used a symmetric window of three words to the left and right of every word. We fed the text again through the neural network in test mode (no weight updating), and we summed the activation of the hidden units of the network for each of the words in the context window that fall within a frequency range of 8 and $30,000$ in the original corpus (the one that was used for the training of the neural network). In this way we excluded low frequency words about which the network might be extremely uncertain, and extremely high frequency function words. We used as weighting schema $w = 2^{1-d}$, were w is the weight for a certain position in the window, and d is the distance in tokens from that position to the center of the window. For instance, the label of the word following the target would be added with a weight $w = 2^{1-1} = 1$ and the label of the word occupying the leftmost position in the window would have a weight $w = 2^{1-3} = 0.25$. When a word in the window was out of the frequency range, its weight was set to 0.0. Punctuation marks were not included in window positions.

4. Results

4.1. Overview of semantics by nearest neighbors

We begin our analysis by inspecting the five nearest neighbors for a given word. Some examples can be found in Table 4.1. To calculate the distances between words, we use normalized cosines (Schone and Jurafsky, 2001). Traditionally, high dimensional lexical vectors have been compared using metrics such as the cosine of the angle between the vectors or the classical Euclidean distance metric or the city-block distance metric. However, using a fixed metric on the components of the vectors induces undesirable effects pertaining to the centrality of representations. More frequent words tend to appear in a much wider range of contexts. When the vectors are calculated as an average of all the tokens of a given type, the vectors or more frequent words will tend to occupy more central positions in the representational space. They will tend to be nearer to all other words, thus introducing an amount of relativity in the distance values. In fact, we believe that this relativity actually reflects people's understanding of word meaning. For example, if we considered the most similar words to a frequent word such as "bird", we would find words as "pigeon" to be very related in meaning. A word such as "penguin" would be considered a more distantly related word. However, if we examined the nearest neighbors of "penguin", we would probably find "bird" among them, although the standard distance measure would still be high. A way to overcome this problem is to place word distances inside a normal distribution, taking into account the distribution of distances of both words. Consider the classical cosine distance between two vectors $\mathbf{v}$ and $\mathbf{w}$:

$$d_{\cos}(\mathbf{v}, \mathbf{w}) = 1 - \frac{\mathbf{v} \cdot \mathbf{w}}{||\mathbf{v}|| \, ||\mathbf{w}||}. \quad (2)$$

For each vector $\mathbf{x} \in \{\mathbf{v}, \mathbf{w}\}$ we calculate the mean ($\mu_{\mathbf{x}}$) and standard deviation ($\sigma_{\mathbf{x}}$) of its cosine distance to 500 randomly chosen vectors of other words. This provides us with an estimate of the mean and standard deviation of the distances between $\mathbf{x}$ and all other words. We can now define the normalized cosine distance between two vectors $\mathbf{v}$ and $\mathbf{w}$ as:

$$d_{norm}(\mathbf{v}, \mathbf{w}) = \max_{\mathbf{x} \in \{\mathbf{v}, \mathbf{w}\}} \left(\frac{d_{\cos}(\mathbf{v}, \mathbf{w}) - \mu_{\mathbf{x}}}{\sigma_{\mathbf{x}}} \right). \quad (3)$$

To speed up this process, the cosine distance means and standard deviation for all words were pre-calculated in advance and stored as part of the representation. The use of normalized cosine distance has the effect of allowing for direct comparisons of the distances between words. In our previous example the distance between "bird" and "penguin", according to a non-normalized metric would suffer from the eccentricity of "penguin"; with the normalization, as the value of the distance would be normalized with respect to "penguin" (the maximum), it would render a value similar to the distance between "bird" and "pigeon".

4.2. Grammatical knowledge

Moscoso del Prado and Baayen (2001) showed that the hidden unit representations of SRN's similar to the one we used here contain information about morpho-syntactic characteristics of the words. In the present technique this information is implicitly available in the input labels for the VBSA technique. The VBSA component however, does not guarantee the preservation of such syntactic information. We therefore need to ascertain whether the grammatical knowledge contained in the SRN vectors is preserved after the application of VBSA.

Note that in Table 4.1., the nearest neighbors of a given word tend to have similar grammatical attributes. For example, plural nouns have other plural nouns as nearest neighbors, e.g., "foreigners" - "others", "outsiders", etc., and verbs tend to have other verbs as nearest neighbors, e.g., "render" - "expose", "reveal", etc. Although the nearest neighbors in Table 4.1. clearly suggest that morpho-syntactic information is coded in the representations, we need to ascertain how much morpho-syntactic information is present and, more importantly, how easily it might be made more explicit. We do this using the techniques proposed in (Moscoso del Prado and Baayen, 2001), that is we employ a machine learning technique using our vectors

Word	Nearest neighbors
hall	centre, theatre, chapel, landscape*, library
half	period, quarter, phase, basis, breeze*
foreigners	others, people, doctors, outsiders, unnecessary*
legislation	orders, contracts, plans, losses, governments
positive	splendid, vital, poetic, similar*, bad
slightly	somewhat, distinctly, little, fake*, supposedly
subjects	issues, films, tasks, substances, materials
taxes	debts, rents, imports, investors, money
render	expose, reveal, extend, ignoring*, develop
re-	anti-, non-, pro-, ex-, pseudo-
omitted	ignored, despised, irrelevant, exploited*, theirs*
Bach	Newton, Webb, Fleming, Emma, Dante

Table 1: Sample of 5 nearest neighbors to some words according to normalized cosine distance. Semantically unrelated words are marked by an asterisk

as input and symbolic grammatical information extracted from the CELEX database (Baayen et al., 1995) as output. A machine learning system is trained to predict the labels from the vectors. The rationale behind this method is very straightforward: If there is a distributed coding of the morpho-syntactic features hidden inside our representation, a standard machine learning technique should be able to detect it.

We begin by assessing whether the grammatical category of a word can be extracted from its vector representation. We randomly selected 500 words that were classified by CELEX as being unambiguously nouns or verbs, that is, they did not have any other possible label. The nouns were sampled evenly between singular an plural nouns, and the verbs were sampled evenly between infinitive, third person singular and gerund forms. Using TiMBL (Daelemans et al., 2000), we trained a memory based learning system on predicting whether a vector corresponded to a noun or a verb. We performed ten-fold cross-validation on the 500 vectors. The systems were trained using 7 nearest neighbors according to a city-block distance metric, the contribution of each component of the vectors weighted by Information Gain Feature Weighting (Quinlan, 1993). To provide a baseline against which to compare the results, we use a second set of files consisting of the same vectors but with random assignment of grammatical category labels to words. The average performance of the system of the Noun-Verb distinction was 68% (randomized averaged 56%). We compared the performance of the system with that of the randomized labels system using a paired two-tailed t-test on the result of each of the runs in the cross-validation, which revealed that the performance of the system was significantly higher than that of the randomized one ($t = 5.63$, $df = 9$, $p = 0.0003$).

We also tested for more subtle inflectional distinctions. We randomly selected 300 words that were unambiguously nouns according to CELEX, sampling evenly from singular and plural nouns. We repeated the test described in the previous paragraph, with the classification task this time being the differentiation between singular and plural. The average performance of the machine learning system was 65% (randomized averaged 48%). A paired two-tailed t-test comparing the results of the systems with the results of systems with the labels randomized revealed again a significant advantage for the non-random system ($t = 5.80$, $df = 9, p = 0.0003$). The same test was performed on a group of 300 randomly chosen unambiguous verbs sampled evenly among infinitive, gerund and third person singular forms, with these labels being the ones the system should learn to predict from the vectors. Performance in differentiating these verbal inflections was of 55% on average while the average of randomized runs was 33%, and significantly above randomized performance accoriding to a paired two-tailed t-test ($t = 4.25$, $df = 9$, $p = 0.0021$).

4.3. Performance in TOEFL synonyms test

Previous studies (Sahlgren, 2001; Landauer and Dumais, 1997) evaluated knowledge about semantic similarity contained in co-occurrence vectors by assessing their performance in a vocabulary test from the Test of English as a Foreign Language (TOEFL). This is a standardized vocabulary test employed by, for instance, American universities, to assess foreign applicants' knowledge of English. In the synonym finding part of the test, participants are asked to select which word is a synonym of another given word, given a choice of four candidates that are generally very related in meaning to the target. In the present experiment, we used the selection of 80 test items described in (Sahlgren, 2001), with the removal of seven test items which contained at least one word that was not present in our representation. This left us with 73 test items consisting of a target word and four possible synonyms. To perform the test, for each test item, we calculated the normalized cosine distance between the target word and each of the candidates, and chose as a synonym the candidate word that showed the smallest cosine distance to the target. The model's performance on the test was 51% of correct responses.

4.3.1. Reliability scores

The results of this test can be improved once we have a measure of the certainty with which the system considers the chosen answer to be a synonym of the target. What we need is a reliability score, according to which, in cases

where the chosen word is not close enough in meaning, i.e., its distance to the target is below a certain probabilistic threshold, the system would refrain from answering. In other words, the system would be allowed to give an answer such as: "I'm not sure about this one". Given that the values of the distances between words in our system, follow a normal distribution $\mathcal{N}(0,1)$, it is quite straightforward to obtain an estimate of the probability of the distance between two words being smaller than a given value, by just using the Normal distribution function $F(x)$. However, while the *general* distribution of distances between any two given words follows $\mathcal{N}(0,1)$, the distribution of the distances from a *particular* word to the other words does not necessarily follow this distribution. In fact they generally do not do so. This difference in the distributions of distances of words is due to effects of prototypicality and probably also word frequency (McDonald and Shillcock, 2001).

To obtain probability scores on how likely it is that a given word is at a certain distance from the target, we need to see the distance of this word *relative* to the distribution of distances from the target word to all other words in the representation. We therefore slightly modify 3, which takes the normalized distance between two words to be the maximum of the cosine distance normalized according to the distribution of distances to the first word, and the cosine distance normalized to the distribution of distances to the second word. We now define the cosine distance between two vectors $\mathbf{v}$ and $\mathbf{w}$ normalized relative to $\mathbf{v}$ as:

$$d^{\mathbf{v}}_{norm} = \frac{d_{\cos}(\mathbf{v},\mathbf{w}) - \mu_{\mathbf{v}}}{\sigma_{\mathbf{v}}}, \qquad (4)$$

which provides us with distances that follow $\mathcal{N}(0,1)$ for each particular word represented by a vector $\mathbf{v}$.

Using 4, we calculated the distance between the target words in the synonym test and the word that the system had selected as most similar, counting only those answers for which the system outputs a probability value below 0.18. The performance on the test increases from 51% to 71%, but the number of items reduced to 45. If we choose probability values below 0.18, the percentage correct continues to rise, but the number of items in the test drops dramatically. Having such a reliability estimator is useful for real-world applications.

4.4. Performance for WordNet synonyms

We can also use the WordNet (Miller, 1990) lexical database to further assess the amount of word similarity knowledge contained in our representations. We randomly selected synonym pairs from each of the four grammatical categories contained in WordNet: nouns, verbs, adjectives and adverbs. We calculated the normalized cosine distance for each of the synonym pairs. As expected, the median distances between synonymous words were clearly smaller than average distance. The median distances were -0.59 for verb synonyms, -0.53 for noun synonyms, -0.49 for adjective synonyms and -0.62 for adverbial synonyms. However, as we have already seen, our vectors contain a great deal of information about morpho-syntactic properties. Hence the fact that synonyms share the same grammatical category could by itself explain the small distances obtained for WordNet synonyms. To check whether this is the case, each synonym pair from our set was coupled with a randomly chosen baseline word of the same grammatical category, and we calculated the distance between one of the synonyms and the baseline word. In this case, as we were interested in the distance of the word relative only to one of the words in the pair, we calculated distances using 4. We compared the series of distances obtained for the true WordNet synonym pairs with the baseline distances by means of two-tailed t-tests. We found that WordNet synonyms were clearly closer in all the cases: nouns ($t = -5.30$, $df = 197$, $p < 0.0001$), verbs ($t = -4.60$, $df = 190$, $p < 0.0001$), adjectives ($t = -3.09$, $df = 195$, $p = 0.0023$) and adverbs ($t = -4.06$, $df = 188$, $p < 0.0001$). This shows that true synonyms were significantly closer in distance space than baseline words.

4.5. Morphology as a measure of meaning

Morphologically related words tend to be related both in form and meaning. This is true both for inflectionally related words, and derivationally related words. As morphological relations tend to reflect regular correspondences to slight changes in the meaning and syntax, they can be used for assessing the amount of semantic knowledge that has been acquired by our system. In what follows, we investigate whether our system is able to recognize inflectional variants of the same word, and whether the vectors of words belonging to the same suffixation class cluster together.

4.5.1. Inflectional morphology

We randomly selected 500 roots that were unambiguously nominal (they did not appear in the CELEX database under any other grammatical category) and for which both the singular and the plural form were present in our dataset. For each of the roots, we calculated the normalized cosine distance between the singular and plural forms. The median of the distance between singular and plural forms was -0.39, which already indicates that inflectional variants of the same noun are represented by similar vectors. As in the case of the WordNet synonyms, it could be argued that this below average distance is completely due to all these word pairs sharing the "noun" property. To ascertain that the observed effect on the distances was at least partly due to real similarities in meaning, each stem r_1 in our set was paired with another stem r_2 also chosen from the original set of 500 nouns. We calculated the normalized cosine distance between the singular form of r_1 and the plural form of r_2. In this way we constructed a data set composed of word pairs plus their normalized cosine distance. A linear mixed effect model (Pinheiro and Bates, 2000) fit to the noun data with normalized cosine distance as dependent variable, the 'stem' (same v. other) as independent variable and the root of the present tense form as random effect, revealed a main effect for stem-sharing pairs ($F(1,499) = 44.42$, $p < 0.0001$). The coefficient of the effect was -0.29 ($\hat{\sigma} = 0.043$). This indicates that the distances between pairs of nouns that share the same stem are in general smaller than the distance between pairs of words that do not share the same root but have the same number. Interestingly, according to a Pearson correlation, 65% of

the variance in the distances is explained by the model.

In the same way, we randomly selected 500 unambiguously verbal roots for which we had the present tense, past tense, gerund and third person singular present tense in our representation. The median normalized cosine distance between the present tense and the other forms of the verb was -0.48, so verbs seem to be clustered together somewhat more tightly than nouns. We repeated the test described above by random pairing of stems, but now we calculated the distances between the present tense form of r_1 and the rest of the inflected forms of r_2. We fit a linear mixed effect model with the normalized cosine distance between the pairs as dependent variable, the pair of inflected forms, i.e., *present-past*, *present-gerund*, or *present-third person singular*, and the 'stem' (same versus different) as independent variables and the root of the first verb as random effect. We found significant, independent effects for type of inflectional pair ($F(1, 2495) = 289.06$, $p < 0.0001$) and stem-sharing ($F(1, 2495) = 109.76$, $p < 0.0001$). The interaction between both independent variables was not significant ($F < 1$). The coefficient for the effect of sharing a root was -0.18 ($\hat{\sigma} = 0.017$), which again indicates that words that share a root have smaller distances than words that do not. It is also interesting to observe that the coefficients for the pairs of inflected forms also provide us with information of how similarly these forms are used in natural language, or, phrased in another way, how similar their meanings are. So, the value of the coefficient for pairs of present tense (uninflected) and past tense forms was -0.48 ($\hat{\sigma} = 0.21$) and the coefficient for pairs composed of a present tense uninflected form and a past tense was -0.38 ($\hat{\sigma} = 0.21$), which suggests that the contexts in which an un-inflected form is used are more similar to the contexts where a past tense form is used than to the contexts of a gerund. The model explained 43% of the variance according to a Pearson correlation.

4.5.2. Derivational morphology

Derivational morphology also captures regular meaning changes, although these changes are often not as regular as the ones that are carried out by inflectional morphology. We tested whether our system captures derivational semantics using the Memory-Based Learning technique that we used for evaluating grammatical knowledge in the system (see section 4.2.). Concentrating on morphological categories, i.e. on words that share the same outer affix. For instance "compositionality" belongs to the morphological category "-ity" and not to the category "-al", although it also contains the suffix "-al". Derivational suffixes generally effect both syntactic and semantic changes. To test whether our vectors reflect semantic regularities, we selected all words ending in the two derivational suffixes "-ist" and "-ness". Both of these suffixes produce nouns, but while the first one generates nouns that are considered agents of actions, the second generates abstract ideas. These affixes generate words with the same grammatical category, but with different semantics. We trained a TiMBL system on predicting the morphological category of the vectors, that is, to predict "-ist" or "-ness". The average performance of the system in predicting these labels in a ten-fold cross-validation was of 78% (compared to an average of 51% obtained when randomizing the affix labels). A paired two-sided t-test between the system performance at each run and the performance of a randomized system on the same run, revealed a significant improvement for the non random system ($t = 10.95$, $df = 9$, $p < 0.0001$).

Although performance was very good for these two nominal affixes, a similar comparison between the adjectival affixes "-able" and "-less", did not render significant differences between randomized and non-randomized labels, indicating that the memory-based learning system was not able to discriminate these two affixes on the sole basis of their semantic vectors. This indicates that, although some of the semantic variance produced by derivational affixes can be captured, many subtler details are being overlooked.

5. Discussion

The analyses that we have performed on the vectors indicate that a high amount of lexical information has been captured by the combination of an SRN with VBSA. On the one hand, the results reported in section 4.2. indicate that the morpho-syntactic information that is coded in the hidden units of a SRN is maintained after the application of VSBA. Moreover, it is clear that the coding of the morpho-syntactic features can be extracted using a standard machine-learning technique such as Memory-Based Learning. This, by itself can be of great use in the bootstrapping of language resources. Given a fairly small set of words that have received morpho-syntactic tags, it is possible to train a machine learning system to identify these labels from their vectors, and then apply this to the vectors of words that are yet to receive morpho-syntactic tagging. Importantly, our technique relies only on word-external order and co-occurrence information, but does not make use of word-internal form information. As it it is evident that word-form information such as presence of inflectional affixes is crucial for morpho-syntactic tagging, our technique can be used to provide a confirmation of possible inflectional candidates. For instance, suppose that two words such as "rabi" and "rabies" are found in a corpora, one would be inclined to classify them as singular and plural version of the same word, when in fact they are both singular forms. The inflectional information in our vectors could be used to disconfirm this hypothesis. In this same aspect, the fact that inflectional variants of the same root tend to be very related in meaning could be used as additional evidence to reject this pair as being inflectional variants.

On the other hand, the nearest neighbors, the TOEFL scores, the results on detecting inflectionally and derivationally related words, and the results on the WordNet synonyms, provide solid evidence that the vectors have succeeded in capturing a great deal of semantic information. Although it is clear to us that our technique needs further fine-tuning, the results are already surprising given the constraints that have been imposed on the system. For instance, the performance on the TOEFL test (51% without the use of the Z scores) is certainly lower than many results that have been reported in the literature. Sahlgren (2001), using the Random Indexing approach to VBSA with random vectors reports 72% correct responses on the same test items. However, he was using a tagged corpus

where all inflectional variants had been unified under the same type. Without the use of stemming, the best performance he reports is of 68%. In the current approach we have used vectors of 150 elements, that is, less than 10% of the size of the vectors used by Sahlgren, and much smaller than the vectors needed to apply techniques such Hyperspace Analog to Language (Lund et al., 1995; Lund and Burgess, 1996) or Latent Semantic Analysis (Landauer and Dumais, 1997) which need to deal with huge co-occurrence matrices. Given the computational requirements of using such huge vectors, we consider that our method provides a good alternative. Our result of 51% on the TOEFL test is clearly above chance performance (25%) and not that far from the results obtained by average foreign applicants to U.S. universities (64.5%). Interestingly, Landauer and Dumais (1997) reported a 64.4% performance on these test items using LSA, but this was only after the application of a dimensional reduction technique (SVD) to their original document co-occurrence vectors. Before the application of SVD, they report a performance of 36% on the plain normalized vectors. Of course, a technique such as SVD could be subsequently applied to the vectors obtained by our method, probably leading to some improvement in our results. However, given that our vectors already have a moderate size, and especially, given that, in their current state, one does not need to re-compute them to add information contained in new corpora, we do not favor the use of such techniques.

Regarding the evaluation of the system against synonym pairs extracted from the WordNet database, although the vectors represent synonyms as being more related than average, it still seems that most of the similarity in these cases was due to morpho-syntactic properties (the average difference in distances between the synonym and baseline conditions was always smaller than 0.1). We believe this is due to several factors. WordNet synonym sets (*synsets*) contain an extremely rich amount of information, that may be too rich for the purposes of evaluating our current vectors. First, many WordNet synonyms correspond to plain spelling variants of the same word in British and American English, e.g., "analyze"-"analise". Our whole training corpus was composed of British English, so the representation of words in American spelling is probably not very accurate. Second, and more importantly, given that the synsets encoded in WordNet reflect in many cases rare or even metaphoric uses of words, we think that the evaluation based on the average type representations provided by our system are not the most appropriate to detect these relations. Possibly, evaluating these synonyms against the vectors corresponding to the particular tokens referring to those senses might be more appropriate. An indication of this is also given by the TOEFL scores, which reflect that the meaning differences can still be detected in many cases. This is important because the synonyms pairs chosen in the TOEFL test, generally reflect the more standard senses of the words involved.

Another important issue is the difference between meaning *relatedness* and meaning *similarity*. These are two different concepts that appear to be somewhat confounded. While our representations reflect in many cases similarity relations, e.g, synonymy, they also appear to capture many relatedness and general world knowledge relations, for instance, the three nearest neighbors of "student" are "university" "pub" and "study", none of which is similar in meaning to "student", but all of them bearing a strong relationship to it. Sahlgren (2001) argues that using a small window to compute the co-occurrences (3 elements to each side, as compared to the 10 elements used in (Burgess and Lund, 1998)), has the effect of concentrating on similarity relations instead of relatedness, which would need much larger contexts such as the full documents used in LSA. The motivation to use very small context windows was to provide an estimation of the syntactic context of words. However, since syntactic information is already made more explicit by our SRN this may not be necessary in our case, and using larger window sizes might actually improve our performance both in similarity and relatedness. A further improvement that should be added to our vectors should come from the inclusion of word internal information. In a pilot experiment we have used the VBSA technique using (automatically constructed) distributed representations of the formal properties of words instead of the random labels. Performance on the TOEFL test were in the same range that was reported here (49%). This suggest that a combination of the technique described here with the formal vectors could probably provide much more precise semantic representations, exploiting both word internal and internal sources of information. This is also in line with the improvement of results found by (Sahlgren, 2001) when using a stemming technique. The use of formal vectors provides an interesting alternative, as it would supply implicit stemming information to the system.

In this paper, we have presented a representation that encodes jointly morpho-syntactic and semantic aspects of words. We have also provided evidence on how morphology is an important cue to meaning, and vice-versa, meaning is also an important cue to morphology. This corroborates previous results from (Schone and Jurafsky, 2001). The idea of integrating formal, syntactic and semantic knowledge about words in one single representation is currently gaining strength within the psycholinguistic community (Gaskell and Marslen-Wilson, 2001; Plaut and Booth, 2000). Some authors are considering morphology as the "convergence of codes", that is, as a set of quasi-regular correspondences between form and meaning, that would probably be linked at a joint representation level (Seidenberg and Gonnerman, 2000). Clear evidence of this strong link has also been put forward by (Ramscar, 2001) showing that the choice of regular or non-regular past tense inflection of a nonce verb is strongly influenced by the context in which the nonce verb appears. If the word appears in a context which entails a meaning similar to that of an irregular verb that is also similar in form to the nonce word, e.g. "frink" - "drink", participants form its past tense in the same manner as the irregular form, e.g., "frank" from "drank". If it appears in a context alike to a similar regular verb, e.g, "wink", participants inflect in regularly, e.g. "frinked" from "winked". Crucially, the meaning of this form is totally determined by context. This in line with the results of (McDonald and Ramscar, 2001), which show how

the meaning of a nonce word is modulated by the context in which it appears. In this respect, our vectors constitute a first approach to such kind of representation: they include contextual and syntactic information. A further step will be the inclusion of word form information in this system, which is left for future research. Our lexical representations are formed by accumulation of predictions. On the one hand, several authors are currently investigating the strong role played by anticipation and prediction in human cognitive processing (e.g., Altmann, 2001). On the other hand, some current models of human lexical processing include the notion of accumulation, generally by recurrent loops in the semantic representations (e.g., Plaut and Booth, 2000).

Acknowledgments We are indebted to Harald Baayen and Rob Schreuder for helpful discussion of the ideas and techniques described in this paper.

The first author was supported by the Dutch Research Council (NWO) through a PIONIER grant awarded to R. Harald Baayen. The second author is funded through the DUMAS project, supported by the European Union IST Programme (contract IST-2000-29452).

6. References

Gerry Altmann. 2001. Grammar learning by adults, infants, and neural networks: A case study. In *7th Annual Conference on Architectures and Mechanisms for Language Processing AMLaP-2001*, Saarbrücken, Germany.

R. Harald Baayen, Richard Piepenbrock, and Léon Gulikers. 1995. *The CELEX lexical database (CD-ROM)*. Linguistic Data Consortium, University of Pennsylvania, Philadelphia, PA.

C. Burgess and K. Lund. 1998. The dynamics of meaning in memory. In E. Dietrich and A. B. Markman, editors, *Cognitive dynamics: Conceptual change in humans and machines*. Lawrence Erlbaum Associates, Mahwah, NJ.

Walter Daelemans, J. Zavrel, K. Van der Sloot, and A. Van den Bosch. 2000. TiMBL: Tilburg Memory Based Learner Reference Guide. Version 3.0. Technical Report ILK 00-01, Computational Linguistics Tilburg University, March.

S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the Society for Information Science*, 41(6):391–407.

J. L. Elman. 1990. Finding structure in time. *Cognitive Science*, 14:179–211.

J. L. Elman. 1993. Learning and development in neural networks: The importance of starting small. *Cognition*, 48:71–99.

M. Gareth Gaskell and William D. Marslen-Wilson. 2001. Representation and competition in the perception of spoken words. *(in press) Cognitive Psychology*.

Z. Harris. 1968. *Mathematical Structures of Language*. New York: Interscience publishers.

P. Kanerva, J. Kristofersson, and A. Holst. 2000. Random indexing of text samples for latent semantic analysis. In *Proceedings of the 22nd Annual Conference of the Cognitive Science Society*, page 1036. Mahwah, New Jersey: Erlbaum.

J. Karlgren and M. Sahlgren. 2001. From words to understanding. In Y. Uesaka, P. Kanerva, and H. Asoh, editors, *Foundations of real-world intelligence*, pages 294–308. Stanford: CSLI Publications.

T. K. Landauer and S. T. Dumais. 1997. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction and representation of knowledge. *Psychological Review*, 104(2):211–240.

K. Lund and C. Burgess. 1996. Producing high-dimensional semantic spaces from lexical co-occurrence. *Behaviour Research Methods, Instruments, and Computers*, 28(2):203–208.

K. Lund, C. Burgess, and R. A. Atchley. 1995. Semantic and associative priming in high-dimensional semantic space. In *Proceedings of the 17th Annual Conference of the Cognitive Science Society*, pages 660–665, Hillsdale, NJ. Erlbaum.

Scott McDonald and Michael Ramscar. 2001. Testing the distributional hypothesis: The influence of context judgements of semantic similarity. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society*.

Scott A. McDonald and Richard C. Shillcock. 2001. Rethinking the word frequency effect: The neglected role of distributional information in lexical processing. *Language and Speech*, 44(3):295–323.

G. A. Miller. 1990. Wordnet: An on-line lexical database. *International Journal of Lexicography*, 3:235–312.

Fermín Moscoso del Prado and R. Harald Baayen. 2001. Unsupervised extraction of high-dimensional lexical representations from corpora using simple recurrent networks. In Alessandro Lenci, Simonetta Montemagni, and Vito Pirrelli, editors, *The Acquisition and Representation of Word Meaning*. Kluwer Academic Publishers (*forthcoming*).

J. C. Pinheiro and D. M. Bates. 2000. *Mixed-effects models in S and S-PLUS*. Statistics and Computing. Springer, New York.

D. C. Plaut and J. R. Booth. 2000. Individual and developmental differences in semantic priming: Empirical and computational support for a single mechanism account of lexical processing. *Psychological Review*, 107:786–823.

J. R. Quinlan. 1993. *Programs for Machine Learning*. Morgan Kauffmann, San Mateo, CA.

Michael Ramscar. 2001. The role of meaning in inflection: Why past tense doesn't require a rule. *(in press) Cognitive Psychology*.

Douglas L. T. Rohde and David C. Plaut. 1999. Language acquisition in the absence of explicit negative evidence: how important is starting small? *Cognition*, 72(1):67–109.

Douglas L. T. Rohde and David C. Plaut. 2001. Less is less in language acquisition. In P. Quinlan, editor, *Connectionist Modelling of Cognitive Development*. (in press) Psychology Press, Hove, U.K.

Douglas L. T. Rohde. 1999. LENS: The light, efficient network simulator. Technical Report CMU-CS-99-164, Carnegie Mellon University, Pittsburg, PA.

Magnus Sahlgren. 2001. Vector-based semantic analysis: Representing word meanings based on random labels. In Alessandro Lenci, Simonetta Montemagni, and Vito Pirrelli, editors, *The Acquisition and Representation of Word Meaning*. Kluwer Academic Publishers (*Forthcoming*).

Patrick Schone and Daniel Jurafsky. 2001. Knowledge free induction of inflectional morphologies. In *Proceedings of the North American Chapter of the Association for Computational Linguistics NAACL-2001*.

Hinrich Schütze. 1992. Dimensions of meaning. In *Proceedings of Supercomputing '92*, pages 787–796.

Mark S. Seidenberg and Laura M. Gonnerman. 2000. Explaining derivational morphology as the convergence of codes. *Trends in the Cognitive Sciences*, 4(9):353–361.

Ludwig Wittgenstein. 1953. *Philosophical Investigations*. Oxford, Blackwell.

G. K. Zipf. 1949. *Human Behavior and the Principle of Least Effort*. Addison-Wesley.

Detection of Errors in PoS-Tagged Corpora by Bootstrapping Generalized Negative *n*-grams

Pavel Květoň and Karel Oliva
Austrian Research Institute for Artificial Intelligence (ÖFAI)
Schottengasse 3, A-1010 Wien, Austria
{pavel,karel}@oefai.at

Abstract

This paper presents two simple yet in practice very efficient bootstrapping techniques serving for automatic detection of those positions in a part-of-speech tagged corpus where an error is to be suspected. The first approach is based on the idea of stepwise learning and application of "negative bigrams", i.e. on the search for pairs of adjacent tags which constitute an incorrect configuration in a text of a particular language (in English, e.g., the bigram *ARTICLE - FINITE VERB*). As the second technique, the paper describes the stepwise generalization of the "negative bigrams" into "negative *n*-grams" (for increasing $n \in N$) which indeed provides a powerful tool for error detection in a corpus. The evaluation of results of the approach when used for error detection in the NEGRA corpus of German and the general implications for the quality of results of statistical taggers are also discussed. Illustrative examples in the text are taken from German, and hence at least a basic command of this language would be helpful for their understanding - due to the complexity of the necessary accompanying explanation, the examples are neither glossed nor translated. However, the central ideas of the paper should be understandable also without any knowledge of German.

1 Errors in PoS-Tagged Corpora

The importance of correctness (error-freeness) of language resources in general and of tagged corpora in particular cannot probably be overestimated. However, the definition of what constitutes an error in a tagged corpus depends on the intended usage of this corpus.

If we consider a quite typical case of a Part-of-Speech (PoS) tagged corpus used for training statistical taggers, then an error is defined naturally as any deviation from the regularities which the system is expected to learn; in this particular case this means that the corpus should contain neither errors in assignment PoS-tags nor ungrammatical constructions in the corpus body[1], since if any of the two cases is present in the corpus, then the learning process necessarily:

- gets a confused view of probability distribution of configurations (e.g., trigrams) in a correct text

and/or, even worse (and, alas, much more likely)

- gets positive evidence also about configurations (e.g., trigrams) which should not occur as the output of tagging linguistically correct texts, while simultaneously getting less evidence about correct configurations.

If we consider PoS-tagged corpora destinated for testing NLP systems, then obviously they should not contain any errors in tagging (since this would be detrimental to the validity of results of the testing) but on the other hand they should contain a certain amount of ungrammatical constructions, in order to test the behaviour of the tested system on a realistic input.

Both these cases share the quiet presupposition that the tagset used is linguistically adequate, i.e. it is sufficient for unequivocal and consistent assignment of tags to the source text[2].

As for using annotated corpora for linguistic research, it seems that even inadequacies in the tagset are tolerable provided they are marked off properly - in fact, these spots in the corpus might well be quite an important source of linguistic investigation since, more often than not, they constitute direct pointers to occurrences of linguistically "interesting" (or at least "difficult") constructions in the text.

2 Representativity

In corpus linguistics, the term representativity is understood as the representativity of a corpus wrt. kind of text or some phenomenon.

In this section, we intend to scrutinize the issue of representativity of a part-of-speech (PoS) tagged corpus wrt. to bigrams[3]. In this case, the phenomena[4] whose presence and relative frequency are at stake are:

- bigrams, i.e. pairs [First,Second] of tags of words occurring in the corpus adjacently and in this order
- unigrams, i.e. the individual tags.

We shall define the *qualitative representativity* wrt. bigrams as the kind of representativity meeting the following two complementary parts:

[1] In this paper we on purpose do not distinguish between "genuine" ungrammaticality, i.e. one which was present already in the source text, and ungrammaticality which came into being as a result of faulty conversion of the source into the corpus-internal format, e.g., incorrect tokenization, OCR-errors, etc.

[2] This problem might be – in a very simplified form – illustrated on an example of a tagset introducing tags for NOUNS and VERBS only, and then trying to tag the sentence *John walks slowly* - whichever tag is assigned to the word *slowly*, it is obviously an incorrect one. Natural as this requirement might seem, it is in fact not met fully satisfactorily in any tagset we know of; for more, cf. (Květoň and Oliva in prep.).

[3] The case of a trigrams, used more usual in tagging practice, would be almost identical but require more lengthy explanations. For the conciseness of argument, we limit the discussion to bigrams in most parts of the text.

[4] In an indeed broadly understood sense of the word "phenomenon".

- the representativity wrt. *the presence of all valid bigrams* of the language in the corpus, which means that if any bigram [First,Second] is a bigram in a correct sentence of the language, then such a bigram occurs also in the corpus - this part might be called *positive representativity*
- the representativity wrt. *the absence of all invalid bigrams* of the language in the corpus, which means that if any bigram [First,Second] is a bigram which cannot occur in a correct (i.e. grammatical) sentence of the language, then such a bigram does not occur in the corpus - this part might be called *negative representativity*.

If a corpus is both positively and negatively representative, then indeed it can be said to be a qualitatively representative corpus[5]. In our particular example this means that a bigram occurs in a qualitatively representative (wrt. bigrams) corpus if and only if it is a possible bigram in the language (and from this it already follows that any unigram occurs in such a corpus if and only if it is a possible unigram[6]). From this formulation, it is also clear that the qualitative representativity depends on the notion of grammaticality, that is, on the "language competence" – on the ability of distinguishing between a grammatical and an ungrammatical sentence.

The *quantitative representativity* of a corpus wrt. bigrams can then be approximated as the requirement that the frequency of any bigram and any unigram occurring in the corpus be in the proportion "as in the language performance" to the frequency of occurrence of all other bigrams or unigrams, respectively[7]. However, even when its basic idea is quite intuitive and natural, it is not entirely clear whether quantitative representativity can be formalized rigorously. At stake is measuring the occurrence of a bigram (and of a unigram) within the "complete language performance", understood as set of utterances of a language. This set, however, is infinite if considered theoretically (i.e. as set of all possible utterances in the language) and finite but practically unattainable if considered as a set of utterances realized within a certain time span (also, due to immanent language change, it is questionable whether the concept of set of utterances over a time span is a true performance of a single language). Notwithstanding these problems, the frequencies are used in practice (e.g., for the purpose of training statistical taggers), and hence it is useful to state openly what they really mean: in our example, it is the relative frequencies of the bigrams (and unigrams) in a particular (learning or otherwise referential) corpus. For this reason, since we would not like to be bound to a particular corpus, we refrain from quantitative representativity in the following and we shall deal only with qualitative representativity.

3 PoS-Tagging Errors Detection

In this (core) section, we shall concentrate on methods and techniques of generating "almost error-free" corpora, or, more precisely, on the possibilities of (semi-)automatic detection (and hence correction) of errors in a PoS-tagged corpus. Due to this, i.e. to the aim of achieving an "error-free" corpus, we shall not distinguish between errors due to incorrect tagging, faulty conversion or ill-formed input, and we shall treat them on a par.

The approach as well as its impact on the correctness of the resulting corpus will be demonstrated on the version 2 of the NEGRA corpus of German (for the corpus itself see www.coli.uni-sb.de/sfb378/negra-corpus, for description cf. (Skut et al. 1997)). However, we believe the solutions developed and presented in this paper are not bound particularly to correcting this corpus or to German, but hold generally.

The error search we use has several phases which differ in the amount of context that has to be taken into consideration during the error detection process. Put plainly, the extent of context mirrors the linguistic complexity of the detection, or, in other words, at the moment when the objective is to search for "complex" errors, the "simple(r)" errors should be already eliminated. The first, preliminary phase, is thus the search for errors which are detectable in the minimal local context of one neighbouring word.

3.1 Bootstrapping Impossible Bigrams

Our starting point is the search for "impossible bigrams". These as a rule occur in a realistic large-scale PoS-tagged corpus, for the following reasons:

- in a hand tagged corpus, an "impossible bigram" results from (and unmistakeably signals) either an ill-formed text in the corpus body (including wrong conversion) or a human error in tagging
- in a corpus tagged by a statistical tagger, an "impossible bigram" may result also from an ill-formed source text, as above, and further either from incorrect tagging of the training data (i.e. the error was seen as a "correct configuration (bigram)" in the training data, and was hence learned by the tagger) or from the process of so-called "smoothing", i.e. of assignment of non-zero probabilities also to configurations (bigrams, in the case discussed) which were not seen in the learning phase[8].

For learning the process of detecting errors in PoS-tagging, let us make a provisional and in practice unrealistic assumption (which we shall correct immediately) that we have a qualitatively representative (wrt. bigrams) corpus of sentences of a certain language at our disposal.

Given such a (hypothetical) corpus, all the bigrams in the corpus are to be collected to a set **CB** (correct bigrams), and then the complement of **CB** to the set of all possible bigrams is to be computed; let this set be called **IB** (incorrect bigrams). The idea is now that if any element of

[5] The definitions of positive and negative representativity are obviously easily transferable to cases with other definitions of a phenomenon. Following this, the definition of qualitative representativity holds of course generally, not only in the particular case of a corpus representative wrt. bigrams.

[6] This assertion holds only on condition that each sentence of the language is of length two (measured in words) or longer. Similarly, a corpus qualitatively representative wrt. trigrams is qualitatively representative wrt. bigrams and wrt. unigrams only on condition that each sentence is of length three at least, etc.

[7] From this it easily follows that any quantitatively representative corpus is also a qualitatively representative corpus.

[8] This "smoothing" is necessary in any purely statistical tagger since - put very simply - otherwise configurations (bigrams) which were not seen during the learning phase cannot be processed if they occur in the text to be tagged.

IB occurs in a PoS-tagged corpus whose correctness is to be checked, then the two adjacent corpus positions where this happened must contain an error (which then can be corrected).

When implementing this approach to error detection, it is first of all necessary to realize that learning the "impossible bigrams" is extremely sensible to both aspects of the qualitative representativity of the learning corpus:

- *the lack of negative representativity:* The presence of an erroneous bigram in the set of **CB** causes that the respective error cannot be detected in the corpus whose correctness is to be checked (even a single occurrence of a bigram in the learning corpus means correctness of the bigram),
- *the lack of positive representativity:* The absence of a correct bigram from the **CB** set causes this bigram to occur in **IB**, and hence any of its occurrences in the checked corpus to be marked as a possible error (absence of a bigram in the learning corpus means incorrectness of the bigram).

However, the available corpora are[9] - at least as a rule - not (qualitatively) representative. Therefore, in practice this deficiency has to be compensated for by appropriate means. When applying the approach to NEGRA, we employed

- bootstrapping for achieving positive representativity as good as possible on a given "training" corpus
- manual pruning of the **CB** and **IB** sets for achieving negative representativity.

We started by very careful hand-cleaning errors in a very small sub-corpus of about 80 sentences (about 1.200 words). From this small corpus, we generated the **CB** set, and pruned it manually, using linguistic knowledge (as well as linguistic imagination) about German syntax. Based on the **CB** set achieved, we generated the corresponding **IB** set and pruned it manually again. The resulting **IB** set was then used for automatic detection of "suspect spots" in the sample of next 500 sentences from the corpus, and for hand-elimination of errors in this sample where appropriate (obviously, not all **IB** violations were genuine errors !). Thus we arrived at a cleaned sample of 580 sentences, which we used just in the same way for generating **CB** set, pruning it, generating **IB** set and pruning this set, arriving at an **IB** set which we used for detection of errors in the whole body of the corpus (about 20.500 sentences, 350.000 positions).

The procedure was then re-applied to the whole corpus. For this purpose, we divided the corpus into four parts of approximately 5.000 sentences each. Then, proceeding in four rounds, first the **IB** set was generated (without manual checking) out of 15.000 sentences and then the **IB** set was applied to the rest of the corpus (on the respective 5.000-sentence partition). The corrections based on the results improved the corpus to such an extent that we made the final round, this time dividing the corpus into 20 partitions with approximately 1.000 sentences each and then reapplying the whole process 20 times.

[9] and will hardly ever become, disregarding their size: e.g., in the body of the 100.000.000 positions of the Czech National Corpus, we easily dicovered a case of a missing trigram (and there are most probably many more missing - we just did not search for them)

3.2 Bootstrapping Impossible *n*-grams

The "impossible bigrams" are a powerful tool for checking the correctness of a corpus, however, a tool which works on a very local scale only, since it is able to detect solely errors which are detectable as deviations from the set of possible pairs of adjacently standing tags. Thus, obviously, quite a number of errors remain undetected by such a strategy. As an example of such an as yet "undetectable" error in German we might take the configuration where two words tagged as finite verbs are separated from each other by a string consisting of nouns, adjectives, articles and prepositions only. In particular, such a configuration is erroneous since the rules of German orthography require that some kind of clause separator (comma, dash, coordinating conjunction) occur inbetween two finite verbs[10].

In order to be able to detect also such kind of errors, the above "impossible bigrams" have to be extended substantially. Searching for the generalization needed, it is first of all necessary to get a linguistic view on the "impossible bigrams", in other words, to get a deeper insight into the impossibility for a certain pair of PoS-tags to occur immediately following each other in any linguistically correct and correctly tagged sentence. The point is that this indeed does not happen by chance, that any "impossible bigram" comes into being as a violation of a certain - predominantly syntactic[11] - rule(s) of the language. Viewed in more detail, these violations might be of the following nature:

- *violation of constituency:* The occurrence of an "impossible bigram" in the text signals that - if the tagging were correct - there is a basic constituency relation violated (resulting in the occurrence of the "impossible bigram"); as an example of such configuration, we might consider the bigram PREPOSITION - FINITE VERB (possible German example string: *...für-PREP reiche-VFIN...*). From this it follows that either there is indeed an error in the source text (in our example, probably a missing word, e.g., *Der Sprecher der UNO-Hilfsorganisation teilte mit, für Arme reiche diese Hilfe nicht.*) or there was a tagging error detected (in the example, e.g., an error as in the sentence *... für reiche Leute ist solche Hilfe nicht nötig...*). The source of the error is in both cases violation of the linguistic rule postulating that, in German, a preposition must always be followed by

[10] At stake are true regular finite forms, exempted are words occurring in fixed collocations which do not function as heads of clauses. As an example of such usage of a finite verb form, one might take the collocation *wie folgt*, e.g., in the sentence *Diese Übersicht sieht wie folgt aus: ...* Mind that in this sentence, the verb *folgt* has no subject, which is impossible with any active finite verb form of a German verb subcategorizing for a subject (and possible only marginally with passive forms, e.g., in *Gestern wurde getanzt*, or – obviously – with verbs which do not subcategorize for a subject, such as *frieren, grauen* in *Mich friert, Mir graut vor Statistik*).

[11] Examples of other such violations are rare and are related mainly to phonological rules. In English, relevant cases would be the word pairs *an table, a apple*, provided the tagset were so fine-grained to express such a distinction, better examples are to be found in other languages, e.g. the case of the Czech ambiguous word *se*, cf. (Oliva to appear).

a corresponding noun (NP) or at least by an adjectival remnant of this NP[12].

- *violation of feature cooccurrence rules* (such as agreement, subcategorization, etc.): The point here is that there exist configurations such that if two wordforms (words with certain morphological features) occur next to each other, they necessarily stand in such a configuration, and because of this also in a certain grammatical relation. This relation, in turn, poses further requirements on the (morphological) features of the two wordforms, and if these requirements are not met, the tags of the two wordforms result in an "impossible bigram". Let us take an example again, this time with tags expressing also morphological characteristics: if the words ... *Staaten schickt* ... are tagged as *Staaten*-NOUN-MASC-PL-NOM and *schickt*-MAINVERB-PRES-ACT-SG, then the respective tags NOUN-MASC-PL-NOM and MAINVERB-PRES-ACT-SG (in this order) create an "impossible bigram". The reason for this bigram being impossible is that if a noun in nominative case occurs in a German clause headed by a finite main verb different from *sein/werden* (which, however, are not tagged as main verbs in the STTS tagset used in NEGRA), then either this noun must be the verb's subject, which in turn requires that the noun and the verb agree in number, or that the noun is a part of coordinated subject, in which case the verb must be in plural. The configuration from the example meets neither of these conditions, and hence it generates an "impossible bigram".

The central observation lies then in the fact that the property of being an impossible configuration can often be retained also after the components of the "impossible bigram" get separated by material occurring inbetween them. Thus, for example, in both our examples the property of being an impossible configuration is conserved if an adverb is placed inbetween, creating thus an "impossible trigram". In particular, in the first example, the configuration PREP ADV VFIN cannot be a valid trigram, exactly for the same reasons as PREP VFIN was not a valid bigram: ADV is not a valid NP remnant. In the second case, the configuration NOUN-MASC-PL-NOM ADV MAINVERB-PRES-ACT-SG is not a valid trigram either, since obviously the presence (or absence) of an adverb in the sentence does not change the subject-verb relation in the sentence. In fact, due to recursivity of language, also two, three and in fact any number of adverbs would not make the configuration grammatical and hence would not disturb the error detection potential of the "extended impossible bigrams" from the examples.

These linguistic considerations have a straightforward practical application. Provided a qualitatively representative (in the above ideal sense) corpus is available, it is possible to construct the **IB** set. Then, for each bigram *[First,Second]* from this set, it is possible to collect all trigrams of the form *[First,Between,Second]* occurring in the corpus, and collect all the possible tags *Between* in the set *Possible_Inner_Tags*. Furthermore, given the impossible bigram *[First,Second]* and the respective set *Possible_Inner_Tags*, the learning corpus is to be searched for all tetragrams *[First,Middle_1,Middle_2, Second]*. In case one of the tags *Middle_1, Middle_2* occurs already in the set *Possible_Inner_Tags*, no action is to be taken, but in case the set *Possible_Inner_Tags* contains neither of *Middle_1, Middle_2*, both the tags *Middle_1* and *Middle_2* are to be added into the set *Possible_Inner_Tags*. The same action is then to be repeated for pentagrams, hexagrams, etc., until the maximal length of sentence in the learn corpus prevents any further prolongation of the *n*-grams and the process terminates.

If now the set *Impossible_Inner_Tags* is constructed as the complement of *Possible_Inner_Tags* relatively to the whole tagset, then any *n*-gram consisting of the tag *First*, of any number of tags from the set *Impossible_Inner_Tags* and finally from the tag *Second* is very likely to be an *n*-gram impossible in the language and hence if it occurs in the corpus whose correctness is to be checked, it is to be signalled as a "suspect spot". Obviously, this idea is again based on the assumption of qualitative representativity of the learning corpus, so that for training on a realistic corpus the correctness of the resulting "impossible *n*-grams" has to be hand-checked. This, however, is well-worth the effort, since the resulting "impossible *n*-grams" are an extremely efficient tool for error detection. The implementation of the idea is a straightforward extension of the above approach to "impossible bigrams". The respective algorithm in a semi-formal coating looks like as in Fig 1.

The above approach does not guarantee, however, that all "impossible *n*-grams" are considered. In particular, any "impossible trigram" *[First,Second,Third]* cannot be detected as such (i.e. as impossible) if the *[First,Second]*, *[Second,Third]* and *[First,Third]* are all possible bigrams (i.e. they all belong to the set **CB**). Such an "impossible trigram" in German is, e.g., *[nominative-noun, main_verb,nominative-noun]* - this trigram is impossible[13] since no German verb apart from *sein/werden* (which, as said above, are not tagged as main verbs in NEGRA) can occur in a context where a nominative noun stands both to its right and to its left, however, all the respective bigrams occur quite commonly (e.g., *Johann schläft, Jetzt schläft Johann, König Johann schläft*). Here, an obvious generalization of the approach from "impossible bigrams" to "impossible trigrams" (and "impossible tetragrams", etc.) is possible, however, we did not perform this in full due to the amount of possible trigrams as well as to the data sparseness problem which, taken together, would make the manual work on checking the results unfeasible in practice. We rather applied only about 20 "impossible trigrams" and 6 "impossible tetragrams" stemming from "linguistic invention" (such as the trigram discussed above).As above, this empirical (performance-based) result has to be checked manually (through a human language competence) for correctness, since the performance results might be distorted by tagging errors or by lack of representativity of the corpus.

[12] unlike English, (standard) German has no preposition stranding and similar phenomena - we disregard the colloquial examples like *Da weiss ich nix von*.

[13] Exempted are quotations and other metalinguistic contexts, such as *Der Fluss heisst Donau, Peter übersetzte Faust - eine Tragödie ins Englische als Fist - one tragedy,* which, however, are as a rule lexically specific and hence can be coped with as such.

```
forall invalid_bigram [First, Second]
{ n := 3;
  possible_i_t := Ø;
  while n =< maximal_sentence_length_in_corpus
                do { find all inner-sentential n-grams [First, V1, V2, .., Vn-2, Second];
                     for each n-gram found
                         do if {V1, V2, .., Vn-2} ∩ possible _i_t = Ø
                              then allowed_i_t := possible _i_t U {V1, V2, .., Vn-2};
                     n := n + 1;
                   };
       impossible_i_t([First, Second]) := tagset - possible _i_t;
}
```

Figure 1: Algorithm for Bootstrapping Negative *n*-grams

The above approach does not guarantee, however, that all "impossible *n*-grams" are considered. For example, any "impossible trigram" *[First,Second,Third]* cannot be detected as such (i.e. as impossible) if the *[First,Second]*, *[Second,Third]* and *[First,Third]* are all possible bigrams (i.e. they all belong to the set **CB**). Such an "impossible trigram" in German is, e.g., *[nominative-noun, main_verb,nominative-noun]* - this trigram is impossible[14] since no German verb apart from *sein/werden* (which, as said above, are not tagged as main verbs via the STTS tagset used in NEGRA) can occur in a context where a nominative noun stands both to its right and to its left, however, all the respective bigrams occur quite commonly (e.g., *Johann schläft, Jetzt schläft Johann, König Johann schläft*). Here, an obvious generalization of the approach from "impossible bigrams" to "impossible trigrams" (and "impossible tetragrams", etc.) is possible, however, we did not perform this in full due to the amount of possible trigrams as well as to the data sparseness problem which, taken together, would make the manual work on checking the results unfeasible in practice. We rather applied only about 20 "impossible trigrams" and 6 "impossible tetragrams" stemming from "linguistic invention" (such as the trigram discussed above).

4 Evaluation of the Results

By means of the error-detection techniques described above, we were able to correct 2.661 errors in the NEGRA corpus. These errors were of all sorts mentioned in Sect. 1, however the prevailing part was that of incorrect tagging (only less than 8% were genuine source errors, about 26% were errors in segmentation). The whole resulted in changes on 3.774 lines of the corpus; the rectification of errors in segmentation resulted in reducing the number of corpus positions by over 700, from 355.096 to 354.354

After finishing the corrections, we experimented with training and testing the TnT tagger (Brants, 2000) on the "old" and on the "corrected" version of NEGRA. We used the same testing as described by Brants, i.e. dividing each of the corpus into ten contiguous parts of equal size, each part having parallel starting and end position in each of the versions, and then running the system ten times, each time training on nine parts and testing on the tenth part, and finally computing the mean of the quality results. In doing so, we arrived at the following results:

- if both the training and the testing was performed on the "old" NEGRA, the tags assigned by the TnT tagger differed from the hand-assigned tags within the test sections on (together) 11.138 positions (out of the total of 355.096), which yields the error rate of 3,14%
- if both the training and the testing was performed on the "correct" NEGRA, the tags assigned by the TnT tagger differed from the hand-assigned tags of the test sections on (together) 10.889 positions (out of the total of 354.354), which yields the error rate of 3,07%
- in the most interesting final experiment, the training was performed on the "old" and the testing on the "correct" NEGRA; in the result, the tags assigned by TnT differed from the hand-assigned tags in the test sections on (together) 12.075 positions (out of the total of 354.354), yielding the error rate of 3,41%.

These results show that there was only a negligible (and, according to the χ^2 test, statistically insignificant) difference between the results in the cases when the tagger was both trained and tested on "old" corpus and both trained and tested on the "corrected" corpus. However, the difference in the error rate when the tagger was once trained on the "old" and once on the "corrected" version, and then in both cases tested on the "corrected" version[15], brought up a relative error improvement of 9,97%. This improvement documents the old and hardly surprising truth that - apart from the size - also the correctness of the training data is absolutely essential for the results of a statistical tagger.

Conclusions

The main contribution of this paper lies in the presentation of a method for detecting errors in part-of-speech tagged corpus which is both quite powerful (as to coverage of errors) and - due to bootstrapping - easy to apply, and hence it offers a relatively low-cost means for achieving high-quality PoS-tagged corpora. The main advantage is that the approach described is based on the combination of focussed search for errors of a particular, specific type with bootstrapping of the search, which makes it possible to detect errors even in a very large corpus where manual checking would not be feasible (at least in practice), since it requires passing through the

[14] Exempted are quotations and other metalinguistic contexts, such as *Der Fluss heisst Donau, Peter übersetzte Faust - eine Tragödie ins Englische als Fist - one tragedy,* which, however, are as a rule lexically specific and hence can be coped with as such.

[15] For obvious reasons, we did not even consider training on the "corrected" corpus and testing on the "old" one.

whole of the text and paying attention to all kinds of possible violations - while the approach described concentrates on violations of particular phenomena on particular spots. Hence, it allows for straightforward checking whether an error really occurs - and if so, for a direct correction.

As a side-effect, it should be also mentioned that the method allows not for detecting errors only, but also for detecting inconsistencies in hand-tagging (i.e. differences in application of a given tagging scheme by different human annotators and/or in different time), and even inconsistencies in the tagging guidelines. A particular issue is further the area of detecting and tagging idioms and collocations, in the particular case when these take a form which makes them deviate from the rules of standard syntax (i.e. they are detected as "suspect spots" by the method). For details on all these points, including the particular problems encountered in NEGRA, cf. (Květoň and Oliva in prep.).

Acknowledgement

This work has been sponsored by the *Fonds zur Förderung der wissenschaftlichen Forschung (FWF), Grant No. P12920.* The *Austrian Research Institute for Artificial Intelligence (ÖFAI)* is supported by the *Austrian Federal Ministry of Education, Science and Culture.*

References

Brants T. (2000). *TnT – A Statistical part-of-speech tagger*, in: Proceedings of the 6th Applied Natural Language Processing conference, Seattle

Hirakawa H., Ono K. and Yoshimura Y. (2000). *Automatic refinement of a PoS tagger using a reliable parser and plain text corpora*, in: Proceedings of the 18th Coling conference, Saarbrücken

Květoň P. and Oliva K. (in prep.). *Correcting the NEGRA Corpus: Methods, Results, Implications*, ÖFAI Technical Report

Müller F.H. and Ule T. (2001). *Satzklammer annotieren und tags korrigieren: Ein mehrstufiges top-down-bottom-up System zur flachen, robusten Annotierung von Sätzen im Deutschen*, in: Proceedings der GLDV-Frühjahrstagung 2001, Gießen

NEGRA. www.coli.uni-sb.de/sfb378/negra-corpus

Oliva K. (2001). *The possibilities of automatic detection/correction of errors in tagged corpora: a pilot study on a German corpus*, in: 4th International conference "Text, Speech and Dialogue" TSD 2001, Lecture Notes in Artificial Intelligence 2166, Springer, Berlin 2001

Oliva K. (to appear). *Linguistics-based tagging of Czech: disambiguation of 'se' as a test case*, in: Proceedings of 4th European Conference on Formal Description of Slavic Languages held in Potsdam from 28th till 30th November 2001

Petkevič V. (2001). *Grammatical agreement and automatic morphological disambiguation of inflectional languages*, in: 4th International conference "Text, Speech and Dialogue" TSD 2001, Lecture Notes in Artificial Intelligence 2166, Springer, Berlin 2001

Schiller A., Teufel S., Stöckert C. and Thielen C. (1999). *Guidelines für das Tagging deutscher Text corpora*, University of Stuttgart / University of Tübingen

Skut W., Krenn B., Brants T. and Uszkoreit H. (1997). *An annotation scheme for free word order languages*, in: Proceedings of the 3rd Applied Natural Language Processing Conference, Washington D.C.

A Comparison Of Efficacy And Assumptions Of Bootstrapping Algorithms For Training Information Extraction Systems

Rayid Ghani* **and Rosie Jones†**

*Accenture Technology Labs
Chicago, IL 60601, USA
rayid.ghani@accenture.com

†School of Computer Science
Carnegie Mellon University, Pittsburgh PA 15213, USA
rosie.jones@cs.cmu.edu

Abstract

Information Extraction systems offer a way of automating the discovery of information from text documents. Research and commercial systems use considerable training data to learn dictionaries and patterns to use for extraction. Learning to extract useful information from text data using only minutes of user time means that we need to leverage unlabeled data to accompany the small amount of labeled data. Several algorithms have been proposed for bootstrapping from very few examples for several text learning tasks but no systematic effort has been made to apply all of them to information extraction tasks. In this paper we compare a bootstrapping algorithm developed for information extraction, meta-bootstrapping, with two others previously developed or evaluated for document classification; cotraining and coEM. We discuss properties of these algorithms that affect their efficacy for training information extraction systems and evaluate their performance when using scant training data for learning several information extraction tasks. We also discuss the assumptions underlying each algorithm such as that seeds supplied by a user will be present and correct in the data, that noun-phrases and their contexts contain redundant information about the distribution of classes, and that syntactic co-occurrence correlates with semantic similarity. We examine these assumptions by assessing their empirical validity across several data sets and information extraction tasks.

1. Introduction

Information Extraction systems offer a way of automating the discovery of information from text documents. Both research and commercial systems for information extraction need large amounts of labeled training data to learn dictionaries and extraction patterns. Collecting these labeled examples can be very expensive, thus emphasizing the need for algorithms that can provide accurate classifications with only a a few labeled examples. One way to reduce the amount of labeled data required is to develop algorithms that can learn effectively from a small number of labeled examples augmented with a large number of unlabeled examples.

Several algorithms have been proposed for bootstrapping from very few examples for several text learning tasks. Using Expectation Maximization to estimate maximum a posteriori parameters of a generative model for text classification (Nigam et al., 2000), using a generative model built from unlabeled data to perform discriminative classification (Jaakkola and Haussler, 1999), and using transductive inference for support vector machines to optimize performance on a specific test set (Joachims, 1999) are some examples that have shown that unlabeled data can significantly improve classification performance, especially with sparse labeled training data. For information extraction, Yangarber et al. used seed information extraction template patterns to find target sentences from unlabeled documents, then assumed strongly correlated patterns are also relevant, for learning new templates. They used an unlabeled corpus of 5,000 to 10,000 documents, and suggest extending the size of the corpus used, as many initial patterns are very infrequently occurring (Yangarber et al., 2000a; Yangarber et al., 2000b).

A related set of research uses labeled and unlabeled data in problem domains where the features naturally divide into two disjoint sets. Blum and Mitchell (Blum and Mitchell, 1998) presented an algorithm for classifying web pages that builds two classifiers: one over the words that appear on the page, and another over the words appearing in hyperlinks pointing to that page. Datasets whose features naturally partition into two sets, and algorithms that use this division, fall into the co-training setting (Blum and Mitchell, 1998). Meta-Bootstrapping (Riloff and Jones, 1999) is an approach to learning dictionaries for information extraction starting only from a handful of phrases which are examples of the target class. It makes use of the fact that noun-phrases and the partial-sentences they are embedded in can be used as two complementary sources of information about semantic classes. Similar methods have been used for named entity classification (Collins and Singer, 1999).

Although a lot of effort has been devoted to developing bootstrapping algorithms for text learning tasks, there has been very little work in systematically applying these algorithms for information extraction and evaluating them on a common set of documents. All of the previously mentioned techniques have been tested on different types of problems, with different sets of documents, under different experimental conditions, thus making it difficult to objectively evaluate the applicability and effectiveness of these algorithms. In this paper, we first describe a range of bootstrapping approaches that fall into the cotraining setting and lay out the underlying assumptions for each. We then experimentally compare the performance of each algorithm on a common set of information extraction tasks and docu-

ments and relate it to the degree to which the assumptions are satisfied in the data sets and semantic learning tasks.

2. The Information Extraction Task

The information extraction tasks we tackle in this paper involve extracting noun phrases that fall into the following three semantic classes: organizations, people and locations. It is important to note that although named entity recognizers are usually used to extract these classes, the distinction we make in this paper is to extract all noun phrases (including "construction company", "jail warden", and "far-flung ports") instead of restricting our task to only proper nouns (which is the case in standard named entity recognizers). Because our focus is extraction of general semantic classes, we have not used many of the features common in English-language named entity recognition, including ones based on sequences of charactes in upper case, and matches to dictionaries, though adding these could improve the accuracy for these classes. This is important to note since that makes it likely that our results will translate to other semantic classes which are not found in online lists or written in capital letters.

The techniques we compare here are similar to those that have been used for semantic lexicon induction (eg (Riloff and Jones, 1999)). However, we believe that the noun-phrases we extract should be taken "in context". Thus, terms we generally consider unambiguous, such as place-names or dictionary terms, can now have different meanings depending on the context that they occur in. For example, the word "Phoenix" usually refers to a location, as in the following sentence:

> *A scenic drive from Phoenix lies a place of legendary beauty.*

but can also refer to the "Phoenix Land Company", as in this sentence:

> *Phoenix seeks to divest non-strategic properties if alternate uses cannot de monstrate sustainable 20% returns on capital investment.*

We can group these types of occurences in three broad categories:

General Polysemy: many words have multiple meanings. For example, "company" can refer to a commercial entity or to companionship.

General Terms: many words have a broad meaning that can refer to entities of various types. For example, "customer" can refer to a person or a company.

Proper Name Ambiguity: proper names can be associated with entities of different types. For example, "John Hancock" can refer to a person or a company, sicne companies are often named after people.

In general, we belive that the context determines whether the meaning of the word can be further determined and that we can correctly classify the noun phrase into the semantic class by examining the immediate context, in addition to the words in the noun phrase. Therefore we approach this problem as an information extraction task, where the goal is to extract and label noun phrase instances that correspond to semantic categories of interest.

3. Data Set and Representation

As our data set, we used 4392 corporate web pages collected for the WebKB project (Craven et al., 1998) of which 4160 were used for training and 232 were set aside as a test set. We preprocessed the web pages by removing HTML tags and adding periods to the end of sentences when necessary.[1] We then parsed the web pages using a shallow parser.

We marked up the held out test data by labeling each noun phrase as one or more of (NP) instance as an organization, person, location, or none. We addressed each task as a binary classification task. Each *noun phrase context* consists of two items: (1) the noun phrase itself, and (2) and the context (an extraction pattern). We used the AutoSlog (Riloff, 1996) system to generate extraction patterns.

By using both the noun phrases and the contexts surrounding them, we provide two different types of features to our classifier. In many cases, the noun phrase itself will be unambiguous and clearly associated with a semantic category (e.g., "the corporation" will nearly always be an organization). In these cases, the noun phrase alone would be sufficient for correct classification. In other cases, the context itself is a dead give-away. For example, the context containing the pattern "subsidiary of <np>" nearly always extracts an organization. In those cases, the context alone is sufficient. However, we suspect that both the noun phrase and the context often play a role in determining the correct classification.

4. Bootstrapping Algorithms

In this section we give a brief overview of each of the algorithms we will be using for bootstrapping. We analyze how the properties and assumptions of each may affect accuracy.

4.1. Baseline Methods

Since our bootstrapping algorithms all use seed noun-phrases for an initial labeling of the training data, we should look at how much of their accuracy is based on the use of those seeds, and how much is derived from bootstrapping using those seeds. To this end, we implemented two baselines which use *only* the seeds, or noun-phrases containing the seeds, but use no bootstrapping.

4.1.1. Extraction Using Seeds Only

All the algorithms we describe use seeds as their source of information about the target class. A useful way of assessing what we gain by using a bootstrapping algorithm is to use the seeds as our sole model of information about the target class. The seeds we use for bootstrapping all algorithms are shown in Table 1.

[1] Web pages pose a problem for parsers because separate lines do not always end with a period (e.g., list items and headers). We used several heuristics to insert periods whenever an independent line or phrase was suspected.

The algorithm for seed extraction is: any noun-phrase in the test set exactly matching a word on the seed list is assigned a score of 1. All other noun-phrases are assigned the prior.

4.1.2. Head Labeling Extraction

All the bootstrapping algorithms we discuss use the seeds to perform *head-labeling* to initialize the training set. The algorithm for head labeling is: any noun-phrase in the training set whose head matches a word on the seed list is assigned a score of 1. This may not lead to completely accurate initialization, if any of the seeds are ambiguous. We will discuss this in more detail in Section 5.1.

In order to evaluate the contribution of the head-labeling to overall performance of the bootstrapping, we performed experiments using the head-labeling alone as information in order to extracted from the unseen test set.

The algorithm for *head labeling extraction* is: any noun-phrase in the test set whose head matches a word on the seed list is assigned a score of 1. All other noun-phrases are assigned the prior.

4.2. Bootstrapping Methods

The bootstrapping methods we describe fall under the cotraining setting where the features naturally partition into multiple disjoint sets, any of which individually is sufficient to learn the task. The separation into feature sets we use for the experiments in this paper is that of noun-phrases, and noun-phrase-contexts.

4.2.1. Cotraining

Cotraining (Blum and Mitchell, 1998) is a bootstrapping algorithm that was originally developed for combining labeled and unlabeled data for text classification. At a high level, it uses a feature split in the data and starting from seed examples, labels the unlabeled data and adds the most confidently labeled examples incrementally. When used in our information extraction setting, the algorithm details are as follows:

1. Initialize NPs from both positive and negative seeds
2. Use labeled NPs to score contexts
3. Select k most confident positive and negative contexts, assign them the positive and negative labels
4. Use labeled contexts to label NPs
5. Select k most confident positive and negative NPs, assign them the positive and negative labels
6. goto 2.

Note that cotraining assumes that we can accurately model the data by assigning noun-phrases and contexts to a class. When we add an example, it is either a member of the class (assigned to the positive class, with a probability of 1.0) or not (assigned to the negative class, with a probability of 0.0 of belonging to the target class). As we will see in section 5.2., many noun-phrases, and many more contexts, are inherently ambiguous. Cotraining may harm its performance through its hard (binary 0/1) class assignment.

4.2.2. CoEM

coEM was originally proposed for semi-supervised text classification by Nigam & Ghani (Nigam and Ghani, 2000) and is similar to the cotraining algorithm described above, but incorporates some features of EM. coEM uses the feature split present in the data, like co-training, but is instead of adding examples incrementally, it is iterative, like EM. It starts off using the same initialization as cotraining and creates two classifiers (one using the NPs and the other using the context) to score the unlabeled examples. Instead of assigning the scored examples positive or negative labels, coEM uses the scores associated with *all* the examples and adds *all* of them to the labeled set probabilistically (in the same way EM does for semi-supervised classification). This process iterates until the classifiers converge.

Muslea et al. (Muslea et al., 2000) extended the co-EM algorithm to incorporate active learning and showed that it has a robust behavior on a large spectrum of problems because of its ability to ask for the labels of the most ambiguous examples, which compensates for the weaknesses of the underlying semi-supervised algorithm.

In order to apply coEM to learning information extraction, we seed it with a small list of words. All noun-phrases with those words as heads are assigned to the positive class, to initialize the algorithm.

Note that coEM does not perform a hard clustering of the data, but assigns probabilities between 0 and 1 of each noun-phrase and context belonging to the target class. This may reflect well the inherent ambiguity of many terms.

4.2.3. Meta-bootstrapping

Meta-bootstrapping (Riloff and Jones, 1999) is a simple two-level bootstrapping algorithm using two features sets to label one another in alternation. It is customized for information extraction, using the feature sets *noun-phrases* and *noun-phrase-contexts* (or *caseframes*). There is no notion of negative examples or features, but only positive features and unlabeled features. The two feature sets are used asymmetrically. The noun-phrases are used as initial data and the set of positive features grows as the algorithm runs, while the noun-phrase-contexts are relearned with each outer iteration.

Heuristics are used to score the features from one set at each iteration, based on co-occurrence frequency with positive and unlabeled features, using both frequency of co-occurrence, and diversity of co-occurring features. The highest scoring features are added to the positive feature list.

Meta-bootstrapping treats the noun-phrases and their contexts asymmetrically. Once a context is labeled as positive, *all* of its co-occurring noun-phrases are assumed to be positive. However, a noun-phrase labeled as positive is part of a committee of noun-phrases voting on the next context to be selected. After a phase of bootstrapping, all contexts learned are discarded, and only the best noun-phrases are retained in the permanent dictionary. The bootstrapping is then recommenced using the expanded list of noun-phrases. Once a noun-phrase is added to the permanent dictionary, it is assumed to be representative of the positive class, with confidence of 1.0.

Class	Seeds
locations	australia, canada, china, england, france, germany, japan, mexico, switzerland, united states
organizations	inc., praxair, company, companies, dataram, halter marine group, xerox, arco, rayonier timberlands, puretec
people	customers, subscriber, people, users, shareholders, individuals, clients, leader, director, customer

Table 1: Seeds used for initialization of bootstrapping.

4.3. Active Initialization

As we saw in the discussion of head-labeling (Section 4.1.2.), using seed words for initializing training may lead to initialization that includes errors. We give measures of the rate of errors in head-labeling in Table 3. We will augment the intialization of bootstrapping by correcting those errors before bootstrapping begins, and seeing the effects on test set extraction accuracy. We call this *active initialization*, by analogy to active learning.

5. Assumptions in Bootstrapping Algorithms

The bootstrapping algorithms described in Section 4.2. have a number of assumptions in common; that initialization from seeds leads to labels which are accurate for the target class, that seeds will be present in the data, that similar syntactic distribution correlates with semantic similarity, and that noun-phrases and their contexts are redundant and unambiguous with respect to the semantic classes we are attempting to learn. We assess the validity of each of these assumptions by examining the data.

5.1. Initialization from Seeds Assumption

All the algorithms we describe use seed words as their source of information about the target class. An assumption made by all the algorithms we present is that seed words suggested by a user will be present in the data. We assess this by comparing seed density for three different tasks over two types of data, one collected specifically for the task at hand, one drawn according to a uniform random distribution over documents on the world wide web. The seeds we use for initializing bootstrapping all algorithms are shown in Table 1. We show the density of seed words in different corpora in Table 2. Note that the `people` and `organizations` classes are much more prevalent in the company data we are working with than in documents randomly obtained using Yahoo's random URL page.

Another assumption that arises from using seeds is that labeling using them accurately labels items in the target semantic class. All three algorithms initialize the unlabeled data by using the seeds to perform *head labeling*. Any noun-phrase with a seed word as its head is labeled as positive. For example, when *canada* is in the seed word list, both "eastern canada" and "marketnet inc. canada" are labeled as being positive examples. Table 3 shows the accuracy for `locations` and `people`. For `people`, some

Corpus	Class	Seed-density (/10,000)
fixed	locations	18
random		21
fixed	organizations	112
random		17
fixed	people	70
random		33

Table 2: Density of seed words per 10,000 noun-phrases in fixes corpus of company web pages, and corpus of randomly collected web pages.

Class	Accuracy
`locations`	98%
`people`	95%

Table 3: Accuracy of labeling examples automatically using seed-heads.

words were mostly unambiguous, with the exception of a few examples, "customers", which was unambigous except in prhases such as "industrial customers". The seed-word "people" also led to some training examples of questionable utility, for example "invest in people". If we learn the context "invest in", it may not help in learning to extract words for people, in the general case. Other seed-words from the `people` class proved to be very ambiguous; "leader" was most often to used to describe a company, as in the sentence "Anacomp is a world leader in digital document-management services".

We will discuss the results of correcting these errors before beginning bootstrapping in Section 6.3.

5.2. Feature Sets Redundancy Assumption

The bootstrapping algorithms we discuss all assume that there is sufficient information in each feature set (noun-phrases and contexts) to use either to label an example. However, when we look at the ambiguity of noun-phrases in the test set (Table 4) we see that 81 noun-phrases were ambiguous between two classes, and 4 were ambiguous between three classes. This means that these 85 noun-phrases (2% of the 4413 unique noun-phrases occurring in the test set) are not in fact sufficient to identify the class. This discrepancy may hurt cotraining and meta-bootstrapping more, since they assume that we can classify noun-phrases into a class with 100% accuracy.

When we examine the same information for contexts (Table 4) we see even more ambiguity. 36% of contexts are ambiguous between two or more classes.

We have another measure of the inherent ambiguity of the noun-phrases making up our target class when we measure the inter-rater(labeler) agreement on the test set. We randomly sampled 230 examples from the test collection, broken into two subsets of size 114 and 116 examples. We had four labelers label subsets with different amounts of information. The three conditions were:

- noun-phrase, local syntactic context, and full sentence (*all*)
- noun-phrase, local syntactic context (*np-context*)

Ambiguity	Class(es)	Number of NPs
1	none	3574
	loc	114
	org	451
	person	189
2	loc, none	6
	org, none	31
	person, none	25
	loc, org	6
	org, person	13
3	loc, org, none	1
	org, person, none	3

Table 4: Distribution of test NPs in classes

Ambiguity	Class(es)	Number of Pats
1	none	1068
	loc	25
	org	98
	person	59
2	loc, none	51
	org, none	271
	person, none	206
	loc, org	5
	org, person	50
3	loc, org, none	18
	org, person, none	83
4	loc, org, person, none	6

Table 5: Distribution of test patterns in classes

- noun-phrase only (*np*).

The labelers were asked to label each example with any or all of the labels `organization`, `person` and `location`. Before-hand, they each labeled 100 examples separate from those described above (in the *all* condition) and discussed ways of resolving ambiguous cases (agreeing, for example, to count "we" as both `person` and `organization` when it could be referring to the organization or the individuals in it. The distribution of conditions to labelers is shown in Figure 6.

We found that when the labelers had access to the noun-phrase, context, and the full sentence they occurred in, they agreed on the labeling 90.5% of the time. However, when one did not have the sentence (only the noun-phrase and context), agreement dropped to 88.5%. Our algorithms have only the noun-phrase and contexts to use for learning. Based on the agreement of our human labelers, we conjecture that the algorithms could do better with more information.

Labeler	Set 1 Condition	Set 2 Condition
1	NP-context	all
2	all	NP-context
3	NP	all
4	all	NP

Table 6: Conditions for inter-rate evaluation - All stands for NP, context and the entire sentence in which the NP-context pair appeared

5.3. Syntactic - Semantic Correlation Assumption

All the algorithms we address in this paper use the assumption that phrases with similar syntactic distributions have similar semantic meanings. It has been shown (Dagan et al., 1999) that syntactic cooccurrence leads to clusterings which are useful for natural language tasks. However, since we seek to extract items from a single semantic target class at a time, syntactic correlation may not be sufficient to represent our desired semantic similarity.

The mismatch between syntactic correlation and semantic similarity can be measured directly by measuring context ambiguity, as we did in Section 5.2.. Consider the context "visit <X>", which is ambiguous between all four of our classes `location`, `person`, `organization` and `none`. It occurs as a `location` in "visit our area", ambiguously between `person` and `organization` in "visit us", and as `none` in "visit our website".

Similarly, examining the ambiguous noun-phrases we see that occurring with a particular noun-phrase does not necessarily determine the semantics of a context. Three of the three-way ambiguous noun-phrases in our test set are: "group", "them" and "they". Adding "they" to the model when learning one class may cause an algorithm to add contexts which belong to a different class.

Meta-bootstrapping deals with this problem by specifically forbidding a list of 35 stop words (mainly prepositions) from being added to the dictionaries. In addition, the heuristic that a caseframe be selected by many different noun-phrases in the seed list helps prevent the addition of a single ambiguous noun-phrase to have too strong an influence on the bootstrapping. The probabilistic labeling used by coEM helps prevent problems from this ambiguity. Though we also implemented a stop-list for cotraining, its all-or-nothing labeling means that ambiguous words not on the stop list (such as "group") may have a strong influence on the bootstrapping.

6. Empirical Comparison of Bootstrapping Algorithms

After running bootstrapping with each algorithm we have two models: (1) a set of noun-phrases, with associated probabilities or scores, and (2) a set of contexts with probabilities or scores. We then use these models to extract examples of the target class from a held-out hand annotated test corpus. Since we are able to associate scores with each test example, we can sort the test results by score, and calculate precision-recall curves.

6.1. Extraction on the Test Corpus

There are several ways of using the models produced by bootstrapping to extract from the test corpus:

1. Use only the noun-phrases. This corresponds to using bootstrapping to acquire a lexicon of terms, along with probabilities or weights reflecting confidence assigned by the bootstrapping algorithm. This may have advantage over lists of terms (such as proper names) which

have no such probabilities associated with them. The probabilities allow us to sort extracted phrases and thus control whether we obtain few, highly probable members of the target class, or obtain good coverage, at the expense of accuracy. We will measure these trade-offs using precision and recall, discussed in Section 6.2..

2. Use only the contexts. In this case we discard the noun-phrases we learned during bootstrapping, and use only the contexts as extraction patterns for extracting on the test set. We extract a noun-phrase when it occurs with one of the contexts in our model, using the score assigned by that context. This may have the advantage of allowing greater generalization. Unseen words and phrases can be extracted from the test corpus, and overspecialization based on the training corpus can be avoided.

3. Use both models. To score a noun-phrase context pair in the test set, assume independence, and multiply the model noun-phrase and context scores to get a probability for the example. Noun-phrases and contexts not seen in the training corpus are given a score based on the prior probability. This has the advantage of combining all the information we acquired during training. This method is most effective for methods which assign probability-like scores (coEM and cotraining). For meta-bootstrapping, there is no natural way of combining the scores.

We experimented with these extraction methods for all three algorithms, and found that method 2, extracting using only the contexts, was by far the best for meta-bootstrapping, so all our results for meta-bootstrapping use this extraction method. CoEM and cotraining performed best with method 3, combining information from both noun-phrase and context models, so all results reported for coEM and cotraining use this extraction method.

6.2. Evaluation

We use the models to score all noun-phrase instances in the test corpus, using context-scoring for meta-bootstrapping, and noun-phrase-context scoring for coEM and cotraining, as described in Section 6.1.. Since we could select a variety of thresholds if we used our models for classifiation, depending on the target application, we use a large number of thresholds, calculating precision and recall for each. Precision is given by

$$Precision = \frac{tp_t}{tp_t + fp_t}$$

where tp_t is the number of correct examples above the threshold, and fp_t is the number of incorrect examples above the threshold. Recall is given by

$$Recall = \frac{tp_t}{tp_t + fn_t}$$

where tp_t is the number of correct examples above the threshold and fn_t is the number of correct examples below the threshold.

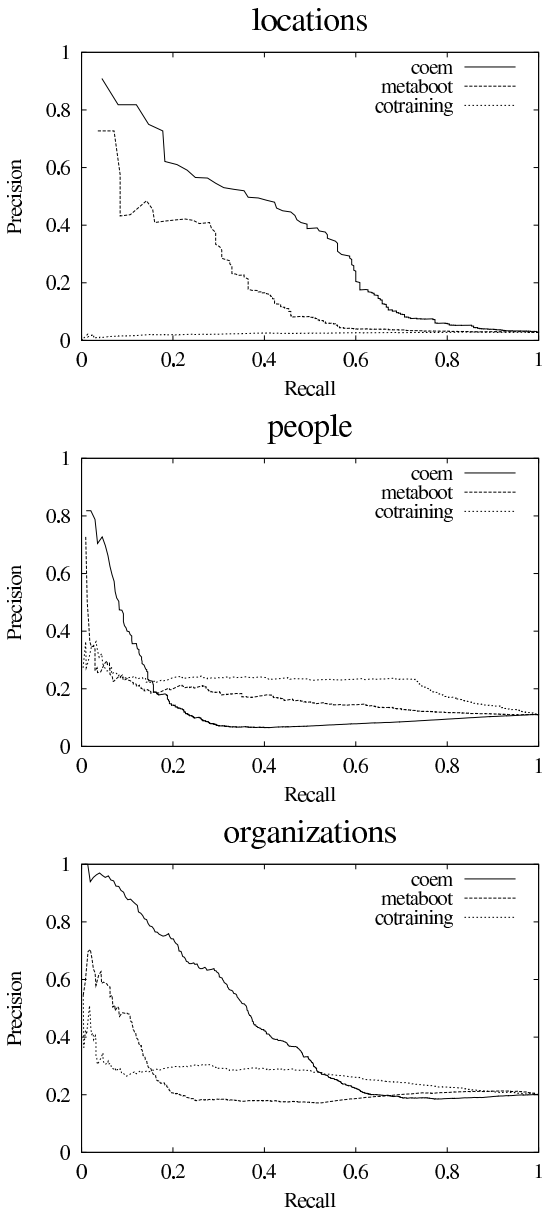


Figure 1: Comparison of bootstrapping using coEM, meta-bootstrapping and cotraining, for the classes `locations`, `people` and `organizations`.

6.3. Experimental Results

Figure 1 compares using models obtained by bootstrapping with coEM, meta-bootstrapping and cotraining, for extracting on a held out test set. CoEM performs better than meta-bootstrapping, while cotraining does very poorly.

Figure 2 shows that bootstrapping using unlabeled documents gives us significant gains over using just the seeds, or noun-phrases with the seeds as heads, for extracting from the test corpus. This difference is least marked for the class `people`, which had the most ambiguous seed words.

Figure 3 shows that only a small gain is obtained by hand-labeling all 669 examples matching the `location` seeds before commencing bootstrapping, and all 2521 examples matching the `people` class before commencing bootstrapping.

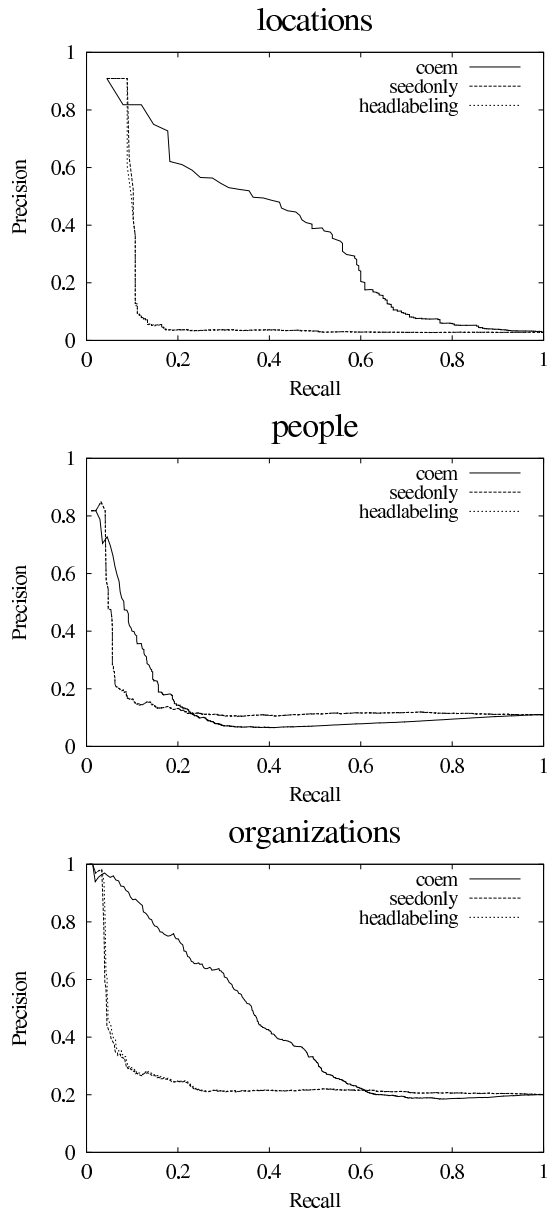


Figure 2: Comparison of the effects of using seeds alone, noun-phrases with seeds as heads (head-labeling) and models learned by bootstrapping with coEM to extract on the unseen test set. Seeds and head-labeling lead to good precision, but poor recall. Bootstrapping using coEM improves recall without loss of precision.

7. Discussion

The advantage coEM has over meta-bootstrapping and cotraining may reflect the good match between its probabilistic treatment of the data, and the inherent ambiguity of the classes. This permits an ambiguous example to be labeled with a probability that reflects its true ambiguity, rather than committing it to a class, then being overly influenced by its presence in that class. Since meta-bootstrapping repeatedly discards the contexts, ambiguity in the contexts does not hurt the algorithm as much as it hurts cotraining.

We can see from the comparison of gains from bootstrapping over using the seeds or head-labeling, that classes for which we have ambiguous seeds words, such as our

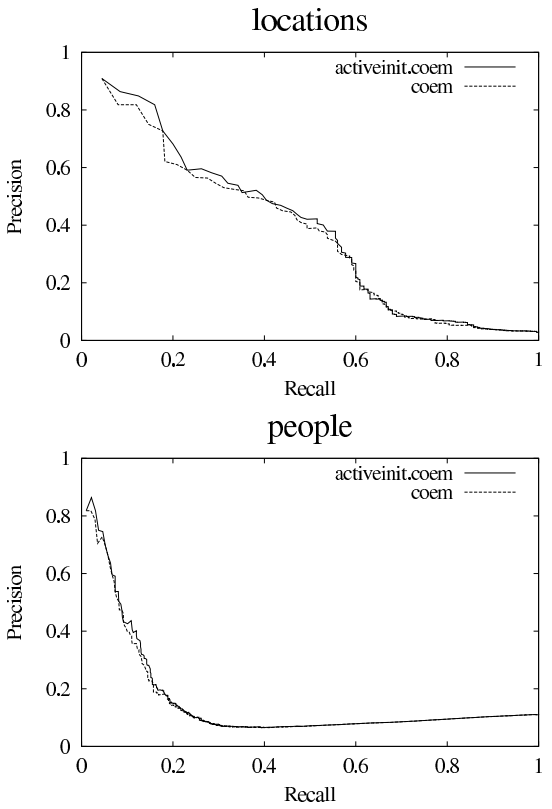


Figure 3: Comparison of the effects of hand-labeling all examples matching the seed-words before commencing bootstrapping (active initialization), against bootstrapping assuming all are correct (coem). A small gain is obtained by labeling all data input.

`people` class benefit less from bootstrapping than those with relatively unambiguous seed words. However, we still benefit from bootstrapping. This may be because the noise introduced by the ambiguous seed-words is somewhat mitigated by the presence of the less ambiguous seed words.

For `locations` and `people` we saw that correcting by hand the examples labeled using the seed words did not have a significant impact on the results. This means that for relatively unambiguous seed words, at least, hand-labeling them in context does not give us an advantage over using automatic head-labeling.

For the seed-words and datasets we used, seed density in the training corpus does not appear to be a major issue.

8. Conclusions and Future Work

We presented a range of bootstrapping algorithms for information extraction and provide experimenal results comparing cotraining, coEM and meta-bootstrapping over a common set of documents and semantic learning tasks. We also analyzed the underlying assumptions for each of the algorithms and found that performance is affected by the degree to which the assumptions are violated in the data set and the task at hand.

We also analyzed several ways of initializing the bootstrapping algorithms and found that the accuracy does not appear to hinge greatly on initialization that is 100% accurate. A greater density of seeds in the training set for a class (`organizations` and `people` had greater seed density

than `locations`) does not appear to lead to greater extraction accuracy on the held out test set. Algorithms which cater to the ambiguity inherent in the feature set are more reliable for bootstrapping, whether they do that by using the feature sets asymmetrically (like meta-bootstrapping), or by allowing probabilistic labeling of examples (like coEM).

Although we have limited the scope of this paper to algorithms that utilize a feature split present in the data (co-training setting), we believe that this comparison of algorithms should be extended to settings where such a split of the features dies not exist, for examples algorithms like expectation maximization (EM) over the entire combined feature set. It would also be helpful to extend the analysis to a greater variety of semantic classes and larger sets of documents.

Acknowledgements

We thank Tom Mitchell and Ellen Riloff for numerous, extremely helpful discussions and suggestions that contributed to the work described in this paper.

9. References

Avrim Blum and Tom Mitchell. 1998. Combining labeled and unlabeled data with co-training. In *COLT: Proceedings of the Workshop on Computational Learning Theory, Morgan Kaufmann Publishers*.

M. Collins and Y. Singer. 1999. Unsupervised Models for Named Entity Classification. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC-99)*.

M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. 1998. Learning to Extract Symbolic Knowledge from the World Wide Web. In *Proceedings of the Fifteenth National Conference on Artificial Intelligence*.

Ido Dagan, Lillian Lee, and Fernando Pereira. 1999. Similarity-based models of cooccurrence probabilities. *Machine Learning*, 34(1-3):43–69.

Tommi Jaakkola and David Haussler. 1999. Exploiting generative models in discriminative classifiers. In *Advances in NIPS 11*.

Thorsten Joachims. 1999. Transductive inference for text classification using support vector machines. In *Proceedings of ICML '99*.

Ion Muslea, Steven Minton, and Craig A. Knoblock. 2000. Selective sampling with redundant views. In *AAAI/IAAI*, pages 621–626.

Kamal Nigam and Rayid Ghani. 2000. Analyzing the effectiveness and applicability of co-training. In *CIKM*, pages 86–93.

Kamal Nigam, Andrew McCallum, Sebastian Thrun, and Tom Mitchell. 2000. Text classification from labeled and unlabeled documents using EM. *Machine Learning*, 39(2/3):103–134.

Ellen Riloff and Rosie Jones. 1999. Learning Dictionaries for Information Extraction by Multi-level Bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence*, pages 1044–1049. The AAAI Press/MIT Press.

E. Riloff. 1996. An Empirical Study of Automated Dictionary Construction for Information Extraction in Three Domains. 85:101–134.

R. Yangarber, R. Grishman, P. Tapanainen, and S. Huttunen. 2000a. Automatic acquisition of domain knowledge for information extraction. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*.

R. Yangarber, R. Grishman, P. Tapanainen, and S. Huttunen. 2000b. Unsupervised discovery of scenario-level patterns for information extraction. In *Proceedings of the Sixth Conference on Applied Natural Language Processing, (ANLP-NAACL 2000)*, pages 282–289.

Using Decision Trees to Predict Human Nouns in Spanish Parsed Text

Marisa Jiménez

Microsoft Research
One Microsoft Way
Redmond, WA 98052, USA
marialj@microsoft.com

Abstract

This paper discusses the use of decision tree models in the acquisition of human nouns for a Spanish monolingual dictionary. This method uses decision tree models to learn the contexts in which a pre-classified set of human nouns occur. We then use the predictions of the model to acquire new human nouns at run time during sentence parsing. The acquisition process was done in 5 stages. First, we annotated automatically all nouns in a selected corpus from Spanish Encarta as "human" and "non-human". Then, we parsed all sentences in the corpus, and extracted linguistic features from the parsed sentences in which each annotated noun occurs. Afterwards we built decision tree models using the data and the features extracted. The task was to classify and assign probabilities to the contexts in which human nouns occur. Finally, we dynamically acquired new human nouns for the Spanish dictionary during sentence parsing; we used the predictions made by the model in this acquisition task.

1. Introduction

Manual annotation schemes to acquire lexical knowledge are costly and time-consuming. To circumvent this problem, different methods to bootstrap already annotated data have been proposed in the literature. One of the bootstrapping methods proposed is using already existing taggers to annotate more data. Some of the work reported focuses on the use of already existing taggers to create mappings between the older tagger and the new tag set (Atwell et al., 1994; Teufel, 1994, among others). Other work proposes combining existing taggers to improve accuracy rates (Van Halteren et al., 1998; Brill and Wu, 1998; Van Halteren et al., 2000; Zavrel and Daelemans, 2000).

Another bootstrapping method that has been proposed is using the statistical distributions of already lexically-classified words to classify new words (Stevenson et al., 1999; Stevenson and Merlo, 1997; Schulte im Walde, 1998). Stevenson and Merlo (2000) discuss a method to automatically classify verbs into semantic classes by looking at the statistical distributions of a few annotated verbs within a big corpus.

Along the lines of the second method mentioned above, this paper discusses a bootstrapping technique to acquire human nouns for a Spanish monolingual dictionary. This method uses decision tree models to learn the contexts in parsed text in which a pre-classified set of human nouns occur. The predictions of the model are then used in the acquisition of new human nouns at run time during sentence parsing.

2. Human Nouns in our Spanish NLP System

The Spanish monolingual dictionary that is part of our Natural Language Processing (NLP) system contains 140,664 entries, of which 72,445 are nouns. Out of these 72,445 nouns, 9,068 are tagged as human nouns in the dictionary. These human nouns were annotated partially by hand and also automatically by using information in their dictionary definitions.

Our system has several strategies to deal with human nouns occurring in text and that are not part of the Spanish dictionary. The first strategy is using derivational morphology rules. If a noun is not in the Spanish dictionary, the system tries to derive it morphologically from a noun that is present in the dictionary. Furthermore, if the noun from which this unfound noun is derived is human, we copy the human information from the base noun. In figure 1 we provide an example of a noun record created derivationally. The noun *camarerito* 'little waiter' is derived from *camarero* 'waiter'. As the base noun is tagged Humn (which stands for "human") in the dictionary, the human tag is copied to the derived noun as well.

```
{Segtype   NOUN
Nodetype   NOUN
Nodename   NOUN1
Ft-Lt      1-1
String     "camarerito"
Lex        "camarerito"
Lemma      "camarerito"
Bits       Masc Pers3 Sing
Derived Humn Count
Anim N_ito
Prob       1.00000
Parent     NP1 "camarerito ."
Infl       Noun-casa
Bases
{Lemma     "camarero"
Bits       N_ito
Cat        Noun}
```

Figure 1. Example of a human noun record created by derivational morphology

We also have a strategy to identify human names that are not in the dictionary when they occur in a collocation. In figure 2, we provide an example of a human name identified by our system, *Boris Karloff*. Although neither *Boris* nor *Karloff* is in the Spanish dictionary, the system is able to recognize them as the first name and last name of a person. Nevertheless, if either *Boris* or *Karloff* appears alone, they are not identified as human names.

```
{Segtype   NOUN
 Nodetype  NOUN
 Nodename  NOUN1
 Ft-Lt     1-2
 String    "Boris Karloff"
 CopyOf    NOUN2
 Rules     (GatherNames)
 Constits  (NOUN3 NOUN2)
 Lemma     "Boris_Karloff"
 Bits      Masc Pers3 Sing PrprN
           Factoid InitCap Humn
           Nme Anim Frme Unfnd
 Prob      1.00000
 Parent    NP1 "Boris Karloff."
 Factrecs  FIRSTNAME1 "Boris"
           LASTNAME1 "Karloff"
 FactPred  person
 FactClass PERSON }
```

Figure 2. Record of a human name identified by our system

Despite these strategies, our system sometimes fails to identify some human nouns that are encountered in text. Knowing whether a noun is human is essential for our Spanish parser as this information is used to identify sentential subjects. Sentential position alone is not sufficient for successful subject identification because Spanish subjects may appear in multiple positions.

Decisions on subject identification are taken as our parser builds up the syntactic tree. Whether a noun is human or not is crucial for subject identification in many instances. One of these cases is when a sentence contains two noun phrases (NPs) that both appear to the right of the verb. If one of the NPs is recognized as human, and the other is not, our parser takes the human NP to be the subject of the sentence.

```
Ayer declaró la ley marcial el presidente de la república.
DECL1     AVP1      ADV1*     "Ayer"
          VERB1*    "declaró"  (Subject NP1  Object NP2)
          NP2       DETP1     ADJ1*     "la"
                    NOUN1*    "ley"  (Predadj AJP1)
                    AJP1      ADJ2*     "marcial"
          NP1       DETP2     ADJ3*     "el"
                    NOUN2*    "presidente"
                    PP1       PP2       PREP1*    "de"
                              DETP3     ADJ4*     "la"
                              NOUN3*    "república"
          CHAR1     "."
```

Translation: Yesterday, the president of the Republic declared martial law.

Figure 3. Example of a Spanish sentence with two NPs to the right of the verb.

In figure 3 we provide an example of a Spanish sentence where human information on an NP is used for subject identification. In this sentence there are two NPs appearing to right of the verb *declaró* 'you-formal declared', NP1, *el presidente de la república* 'the president of the Republic', and NP2, *la ley marcial* 'martial law'. In order to determine that NP1 is the subject of the sentence, the parser uses the fact that the head of the NP *presidente* is marked human in the Spanish dictionary.

3. Motivation and Experiment Design

Motivated by the importance of human nouns for our NLP system, we designed a bootstrapping method to add new human nouns to the Spanish dictionary. This method uses decision tree models to learn the contexts in parsed text in which a pre-classified set of human nouns occur. The predictions made by the model are then used in the dynamic acquisition of new human nouns during sentence parsing.

There were 5 stages in the experiment design:

- Automatic annotation of all nouns in a selected corpus into "human" and "non-human".
- Parsing of sentences in the selected corpus.
- Linguistic feature extraction from the parsed sentences in which each annotated noun occurs. The goal was determining which features were relevant or not with respect to human nouns.
- Building decision tree models using the features extracted. The task was to classify and assign probabilities to the contexts in which human nouns occur.
- Dynamically adding new human nouns to the Spanish dictionary based on the predictions made by the model.

In section 4 we will describe the first 4 stages of our experiment, which have to do with the different steps in building the decision tree models. In section 5 we will discuss the dynamic acquisition of new human nouns using the model predictions.

4. Using Decision Trees to Predict Human Nouns

4.1. Data and Feature Extraction

We used the Spanish version of Encarta as the data resource for our experiment because this encyclopedia is a good source of human nouns. We gathered 126,935 sentences, and extracted all their nouns. There were a total of 641,673 nouns, which we then annotated automatically. Those nouns that were recognized as human by our system were tagged as "human", and the rest were tagged as "not-human". Unfound words were excluded from the annotation task for obvious reasons.

We were quite confident that these automatic tags had a high degree of accuracy. Our confidence was based on the fact that the Spanish system has mechanisms to identify human nouns that are not in the dictionary. Furthermore, over the years we have done manual revisions of the 10,000 most common nouns in the Spanish dictionary; in these revisions we made sure that all the human nouns in the high-frequency set were tagged correctly.

Despite our degree of confidence in the accuracy of the automatic tags, we did some manual revision to have an estimation of our error rate. We reviewed 3,000 tagged nouns by hand; they were extracted at random from different parts of the corpus. We did not find any errors in the subset of tags reviewed; this gave us confidence that the error rate was small.[1]

[1] We would like to thank an anonymous reviewer for his/her comments on making sure that our tags were correct. His/her

The next step in the experiment was parsing all the sentences associated with the tagged nouns. We discarded sentences that did not obtain a complete parse. Afterwards, we automatically extracted 232 linguistic features from the parsed tree of the sentence associated with each tagged noun. Our approach was extracting the full set of features available in the parse, instead of performing manual feature selection. Nevertheless, one of the features extracted, "the ending of the noun", was selected manually. We included this morphological feature because of its highly suspected relevance.

The pool of extracted features fell into the following categories:

- All verbal features present in the main verb of the sentence.
- Selected features of the parent and the grandparent of the noun, such as gender, number, and (in) definiteness, among others.
- All the features present in the pre-modifiers and post-modifiers of the parent.
- The lemma of the preposition governing the parent, if present.
- The syntactic label of the parent and the grandparent.
- The ending of the noun.

In figure 4 we provide a sample of some features extracted for the noun *películas* 'movies'. The noun and its sentence are listed first. Under *Values* some values for the features extracted are listed. The first value is always the value of the tag, which is "NoHuman" in this case.

Sentence: [películas]: Durante los últimos veinticinco años de su vida, Gabin hizo unas veinticuatro películas, entre ellas destaca El clan de los sicilianos (1969) de Henri Verneuil, obra maestra del cine negro francés.

Translation: During the last twenty-five years of his life, Gabin made twenty-four movies, among them 'The Sicilian Clan' (1969), directed by Henri Verneuil, master piece of the French *film noir*.

Values:
A~is_human_noun = "NoHuman"
A~has_possible_human_ending = "NoPossHumEnding"
1~Sing~Parent = "0"
1~Plur~Parent = "1"
1~Art~Parent = "1"
1~Def~Parent = "0"
1~PPobj~Parent = "0"
1~Nodetype~Parent = "NP"

Figure 4. Values of some of the features extracted for the noun *películas* 'movies'

4.2. Building and Examining the Models

Once the feature extraction was completed, we used the data and the values extracted to build decision tree models. The goal was to classify the features by their relevance in predicting human nouns. Decision trees are a popular tool used in machine-learning for classification tasks. They provide a classification of selected features and rank their relative importance in predicting a target feature. The tools that we used to build our decision trees are the WinMine toolkit (Chickering *et al.*, 1997, n.d.), developed at Microsoft Research. Decision trees built by WinMine predict a probability distribution over all possible target values.

These tools take as input a text file with the characteristics shown in figure 4. The data is split into training and testing at a 70/30 rate. For both training and testing, we only extracted features from sentences that had a complete parse. The tools produce several decision tree models at different levels of accuracy. All the models are in xml format.

In order to inspect the models, we use a model viewer that allows the user to view the shape of the decision tree. This tool also shows the relative importance of all the relevant features selected by the model. All features that are found relevant in the human model are connected with arrows to the target feature that we are trying to predict, which appears in the center of the viewer. The features that are not found relevant by the model are shown disconnected at the bottom of the screen.

In figure 5 we provide a snapshot of the four best predictors in the model for human nouns. The viewer has a slider (to the left of the figure) that highlights the strongest predictors as the slider goes up. The following characteristics of the main verb are the top predictors in the model shown in figure 5:

- The verb takes a nominal predicate object (e.g.: *Lo considero mi amigo* 'I consider him my friend').
- The verb is transitive.
- The verb is transitive and followed by preposition + infinitive (e.g.: *Te animo a venir* 'I encourage you to come').
- The verb takes an adjectival predicate object (e.g.: *No te creo tonto* 'I don't believe you stupid').

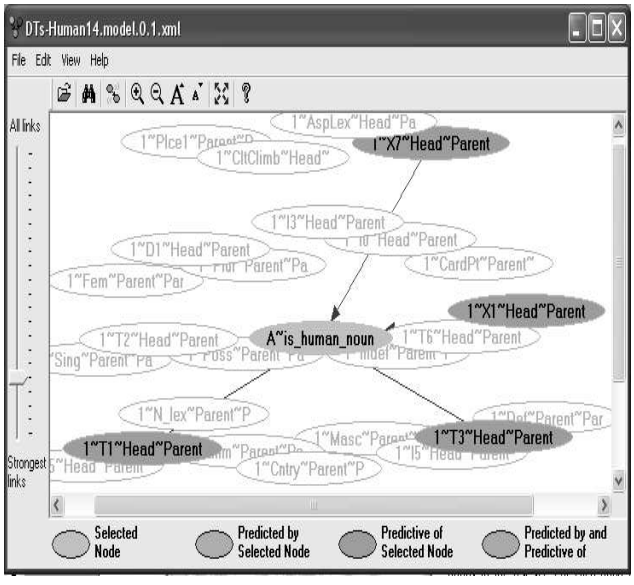


Figure 5. Snapshot of the four top predictors in the model for human nouns

concern was that we would tag as "non-human" true human nouns that were just missing the tag in our dictionary. The manual revision of a 3000 noun sample confirmed to us that the error rate was small.

4.3. Model Predictions

Out of the 232 features that were extracted for each noun, 111 were found to have predictive value in the model for human nouns. The selected features fell into the following categories:

- Ending of the noun. Certain endings such as *–or* '-er', and *–ista* '-ist' were found to be strong indicators that the noun is human.
- Governing preposition of the parent. Certain prepositions such as *a* 'to' and *por* 'by' were found likely to govern a human noun, while others such as *en* 'in' tend not to govern human nouns.
- The noun was upper case. Upper case was found to be a strong predictor in certain contexts, for example, nominal apposition.
- Syntactic label of the parent and grandparent.
- Whether the parent is in apposition to another noun.
- Syntactic features of the sentence main verb.
- Whether the parent of the noun has pre-modifiers.
- Some features of the pre-modifiers and post-modifiers of the parent (e.g.: whether the parent has a definite post-modifier, and the pre-modifier is a possessive pronoun).
- Whether the parent is definite.
- Whether the parent has a post-modifier that is a relative clause.

Some of the predictions made by the model were pretty straightforward, such as the relevance of the noun ending or capitalization. Other predictions were not as straightforward, such as the relevance of the parent being post-modified by a relative clause.

One of the main advantages of the decision tree model for human nouns was its complexity. The model had 1,194 branching nodes, which means that 1,194 linguistic decisions were made when predicting whether a noun is likely to be human or not. Manually coding each one of these decisions would be extremely time-consuming.

As for evaluation numbers, the best model had an overall accuracy of 84.29% over a 69.83% baseline. The baseline corresponds to the accuracy if the most frequent value (non-human) had been assigned to all nouns in the test set. In table 1 we provide the evaluation numbers for this model. These calculations are based on the 1922502 nouns in the test set. For each noun in the test set, the value predicted by the model was compared to the value observed.

	Human	Non-Human	Total
Total	58066	134436	192502
Predicted	54238	138264	
Correctly Predicted	46030	116228	162258
Precision	84.86%	84.06%	
Recall	79.27%	86.45%	
F-measure	81.97%	85.23%	
Baseline			69.83%
Overall Accuracy			84.29%

Table 1. Evaluation numbers for the human model

5. Using the Decision Tree Model to Acquire New Human Nouns

The last step in our experiment was using the predictions made by the model in the acquisition of human nouns that were not yet in the Spanish dictionary. The plan was to add new human noun records dynamically during sentence analysis using the decision tree predictions.

Our NLP system has in place rules to do lexical learning at run time during sentence parsing. Lexical-learning rules are used dynamically to create domain-specific corpus-based lexicons. These domain-specific dictionaries are used as supplements to the general dictionary (see Pentheroudakis (technical report), and Wu et al.,2002).

In preparation for lexical learning, we gathered 57,397 sentences from Spanish Encarta; we made sure that these sentences contained unfound words.[2] Afterwards, the decision tree model for human nouns was invoked from the lexical-learning rules while parsing the sentences.

After parsing the sentences, a new learned lexicon was created; this lexicon contained 16,902 human nouns. After quickly inspecting a handful of dictionary entries, we realized that a good amount of the learned nouns were human names. This realization seemed consistent with the fact that Encarta is an encyclopedia, and that our Spanish dictionary does not contain many proper names. Some examples of the names that were learned were *Filippo, Yourcenar* and *Hemingway*, among others. Among the "non-proper name" human nouns that were learned were *zapatistas* 'zapatists', *antirreeleccionista* 'anti-reelectionist', and *clarinero* 'clarinist', among others.

To evaluate the accuracy of the learned dictionary, we randomly gathered 600 nouns from the pool of sentences used for the creation of the dictionary. We then manually reviewed all the words in the 600 set. We checked whether each noun was truly human or not, and whether it was part of the learned dictionary. 41 nouns from the 600 set were discarded because they were either typos or did not have a noun part of speech; we ended up with a total of 559 nouns.[3] In table 2, we provide the results of the manual evaluation.

	Human	Non-Human	Total
Total	269	290	559
Predicted	230	329	
Correctly Predicted	205	265	470
Precision	89.13%	80.54%	
Recall	76.20%	91.37%	
Baseline			51.88%
F-measure	82.16%	85.61%	
Overall Accuracy			84.07%

Table 2. Summary of manual evaluation of 559 nouns from Encarta

[2] The first time that we invoked the model during parsing, we did not make sure that unfound words were in the data. As a result, few new human nouns were learned.

[3] Our NLP system assigns by default a noun part of speech to unfound words. It is possible that some of the unfound nouns that were in the test set were not nouns.

6. Conclusions

In this paper we have presented a method to augment the number of human nouns in the Spanish dictionary of our NLP system. This method uses decision tree models to learn the contexts in which a pre-classified set of human nouns occur. The experiment was done in 5 stages. First, we annotated automatically all nouns in a selected corpus from Spanish Encarta into "human" and "non-human". We then parsed all sentences in the corpus, and extracted several linguistic features from each parsed sentence. We then built decision tree models using the features extracted. The task was to classify and assign probabilities to the contexts in which human nouns occur. Finally we added dynamically new human nouns to the Spanish dictionary based on the predictions made by the model.

Evaluation of our experiment showed that we were able to learn a significant amount of human nouns at good accuracy levels. This method reduces human effort in dictionary maintenance. We see another two advantages to using decision tree models for human noun acquisition. The first one is that the model makes complex decisions that would be very costly and time-consuming to be hand-coded. And, second, the model made more complete predictions than our native speaker intuitions.

7. Acknowledgments

We would to thank the members of the NLP group at Microsoft Research for their comments and help at various stages during the development of this paper.

8. References

Atwell, E., J. Hughes, and C. Souter (1994). Amalgam: Automatic Mapping among Lexico-grammatical Annotation Models. Technical report, Internal Paper, CCALAS, Leeds University.

Brill, E. and J. Wu (1998). Classifier Combination for Improved Lexical Disambiguation. In *COLING-ACL'98* Montreal, Canada.

Chikering, D. Max nd. *WinMine Toolkit Home Page*. http.//research.microsoft.com/~dmax/WinMine/Tooldoc .htm.

Pentheroudakis, J. (2001). *Lex Rules!*. Technical report.

Schulte im Walde, S., 1998. Automatic Semantic Classification of Verbs according to their Alternation Behaviour. AIMS Report 4(3), IMS, Universität Stuttgart.

Stevenson, S. and P. Merlo (1997). Lexical Structure and Processing Complexity. *Language and Cognitive Processes*, 12(1-2):349-399.

Stevenson, S., P. Merlo, N. Karaeva, and K. Whitehouse (1999). Supervised Learning of Lexical Semantic Verb Classes using Frequency Distributions. In *Procs of SigLex '99*, College Park, Maryland.

Stevenson, S. and P. Merlo (2000). Automatic Lexical Acquisition Based on Statistical Distributions. *Proceedings of COLING 2000*, Saarbrüecken, Germany.

Teufel, S. (1995). A Support Tool for Tagset Mapping. In *Proc. of of the Workshop SIGDAT (EACL95)*

Van Halteren, H., J. Zavrel, and W. Daelemans (1998). Improving Data Driven Wordclass Tagging by System Combination. In *Proceedings of ACL-COLING'98*, Montreal, Canada.

Van Halteren, H., J. Zavrel, and W. Daelemans (2001). Improving Accuracy in NLP through Combination of Machine Learning Systems. *Computational Linguistics 27 (2), 1*99-230.

Wu, A., J. Pentheroudakis, and Z. Jiang (2002). Dynamic Lexical Acquisition in Chinese Sentence Analysis. Submitted to COLING 2002 for consideration.

Zavrel, H.J. and W. Daelemans (2000). Bootstrapping a Tagged Corpus through Combination of Existing Heterogeneous Taggers. *International Conference on Language Resources and Evaluation.* Athens, Greece.

X-TRACTOR: A Tool For Extracting Discourse Markers

Laura Alonso*, Irene Castellón*, Lluís Padró†

*Department of General Linguistics
Universitat de Barcelona
{lalonso, castel}@lingua.fil.ub.es

†TALP Research Center
Software Department
Universitat Politècnica de Catalunya
padro@lsi.upc.es

Abstract

Discourse Markers (DMs) are among the most popular clues for capturing discourse structure for NLP applications. However, they suffer from inconsistency and uneven coverage. In this paper we present X-TRACTOR, a language-independant system for automatically extracting DMs from plain text. Seeking low processing cost and wide applicability, we have tried to remain independent of any hand-crafted resources, including annotated corpora or NLP tools. Results of an application to Spanish point that this system succeeds in finding new DMs in corpus and ranking them according to their likelihood as DMs. Moreover, due to its modular architecture, X-TRACTOR evidences the specific contribution of each out of a number of parameters to characterise DMs. Therefore, this tool can be used not only for obtaining DM lexicons for heterogeneous purposes, but also for empirically delimiting the concept of DM.

1. Motivation

The problem of capturing discourse structure for complex NLP tasks has often been addressed by exploiting surface clues that can yield a partial structure of discourse (Marcu, 1997; Dale and Knott, 1995; Kim et al., 2000). Cue phrases such as *because*, *although* or *in that case*, usually called Discourse Markers (DMs), are among the most popular of these clues because they are both highly informative of discourse structure and have a very low processing cost.

However, they present two main shortcomings: inconsistency in their characterisation and uneven coverage. The lack of consensus about the concept of DM, both theoretically and for NLP applications, is the main cause for these two shortcomings. In this paper, we will show how a knowledge-poor approach to lexical acquisition is useful for addressing both these problems and providing partial solutions to them.

1.1. Delimitation of the concept of DM

A general consensus has not been achieved about the concept of DM. The set of DMs in a language is not delimited, nor by intension neither by extension. But however controversial DM characterisation may be, there is a core of well-defined, prototypical DMs upon which a high consensus can be found in the literature. By studying this lexicon and the behaviour of the lexical units it stores in naturally occurring text, DM characterising features can be discovered. These features can be applied to corpus to obtain lexical items that are similar to the original ones. Applying bootstraping techniques, these newly identified lexical items can be incorporated to the lexicon and this enhanced lexicon can be used for discovering new characterising features. This process can be repeated until the obtained lexical items are not considered valid any more.

It may be argued that enlarging this starting set implies making it more controversial, by adding items whose status as DMs is questionable. However, being empirically grounded, this enlargement is relatively unbiased, and it yields an enhancement of the concept of DM that may be useful for NLP applications.

Taking it to the extreme, unendlessly enhancing the concept of DM implies that anything loosely signalling discourse structure would be considered as a DM. Although this might sound absolutely undesirable, it could be argued that a number of lexical items can be assigned a varying degree of marking strength or *markerhood*[1]. It would be then up to the human expert to determine the load of *markerhood* required for a lexical item to be considered a DM in a determined theoretical framework or application. Lexical acquisition can evidence the load of discursive information in every DM by evaluating it according to the DM characterising features used for extraction.

1.2. Scalability and Portability of DM Resources

Work concerning DMs has been mainly theoretical, and applications to NLP have been mainly oriented to restricted NLGeneration applications. So, DM resources of wide coverage have still to be built. The usual approach to building DM resources is fully manual. For example, DM lexicons are built by gathering and describing DMs from corpus or literature on the subject, a very costly and time-consuming process. Moreover, due to variability among humans, DM lexicons tend to suffer from inconsistency in their extension and intension. To inherent human variability, one must add the general lack of consensus about the appropriate characterisation of DMs for NLP. All this prevents reusability of these costly resources.

[1]By analogy with *termhood*(Kageura and Umino, 1996), which is the term used in terminology extraction to indicate the likelihood that a term candidate is an actual term, we have called *markerhood* the likelihood that a DM candidate is an actual DM.

As a result of the fact that DM resources are built manually, they present uneven coverage of the actual DMs in corpus. More concretely, when working on previously unseen text, it is quite probable that it contains DMs that are not in a manually built DM lexicon. This is a general shortcoming of all knowledge that has to be obtained from corpus, but it becomes more critical with DMs, since they are very sparse in comparison to other kinds of corpus-derived knowledge, such as terminology. As follows, due to the limitations of humans, a lexicon built by mere manual corpus observation will cover a very small number of all possible DMs.

The rest of the paper is organised as follows. In Section 2., we present the architecture of the proposed extraction system, X-TRACTOR, with examples of an application of this system to acquiring a DM lexicon for discourse-based automated text summarisation in Spanish. In Section 2 we present the results obtained for this application, to finish with conclusions and future directions.

2. Proposed Architecture

One of the main aims of this system is to be useful for a variety of tasks or languages. Therefore, we have tried to remain independent of any hand-crafted resources, including annotated texts or NLP tools. Following the line of (Engehard and Pantera, 1994), syntactical information is worked by way of patterns of function words, which are finite and therefore listable. This makes the cost of the system quite low both in terms of processing and human resources.

Focusing on adaptability, the architecture of X-TRACTOR is highly modular. As can be seen in Figure 1, it is based in a language-independent kernel implemented in perl and a number of modules that provide linguistic knowledge.

The input to the system is a starting DM lexicon and a corpus with no linguistic annotation. DM candidates are extracted from corpus by applying linguistic knowledge to it. Two kinds of knowledge can be distinguished: general knowledge from the language and that obtained from a starting DM lexicon.

The DM extraction kernel works in two phases: first, a list of all might-be-DMs in the corpus is obtained, with some characterising features associated to it. A second step consists in ranking DM candidates by their likelihood to be actual markers, or *markerhood*. This ranked list is validated by a human expert, and actual DMs are introduced in the DM lexicon. This enhanced lexicon can be then re-used as input for the system.

In what follows we describe the different parts of X-TRACTOR in detail.

2.1. Linguistic Knowledge

Two kinds of linguistic knowledge are distinguished: general and lexicon-specific. General knowledge is stored in two modules. One of them accounts for the distribution of DMs in naturally occurring text in the form of rules. It is rather language-independant, since it exploits general discursive properties such as the occurrence in discursively salient contexts, like beginning of paragraph or sentence. The second module is a list of stopwords or function words of the language in use.

Lexicon-specific knowledge is obtained from the starting DM lexicon. It also consists of two modules: one containing classes of words that constitute DMs and another with the rules for legally combining these classes of words. We are currently working in an automatic process to induce these rules from the given classes of words and the DMs in the lexicon.

In the application of this system to Spanish, we started with a Spanish DM lexicon consisting of 577 DMs [2]. Since this lexicon is oriented to discourse-based text summarisation, each DM is associated to information useful for the task (see Table 1), such as *rhetoric type*. We adapted the system so that some of this information could also be automatically extracted for the human expert to validate. Results were excellent for the feature of *syntactic type*, and very good for *rhetorical content* and *segment boundary*.

We transformed this lexicon to the kind of knowledge required by X-TRACTOR, and obtained 6 classes of words (adverbs, prepositions, coordinating conjunctions, subordinating conjunctions, pronouns and content words), totalling 603 lexical items, and 102 rules for combining them. For implementation, the words are listed and they are treated by pattern-matching, and the rules are expressed in the form of *if - then - else* conditions on this pattern-matching (see Table 2).

2.2. DM candidate extraction

DM candidates are extracted by applying the above mentioned linguistic knowledge to plain text. Since DMs suffer from data sparseness, it is necessary to work with a huge corpus to obtain a relatively good characterisation of DMs. In the application to Spanish, strings were extracted by at least one of the following conditions:

- Salient location in textual structure: beginning of paragraph, beginning of the sentence, marked by punctuation.
- Words that are typical parts of DMs, such as those having a strong rhetorical content. thetorical content types are similr to those handled in RST (Mann and Thompson, 1988).
- Word patterns, combinations of function words, sometimes also combined with DM-words.

2.3. Assessment of DM-candidate markerood

Once all the possible might-be-DMs are obtained from corpus, they are ponderated as to their *markerhood*, and a ranked list is built.

Different kinds of information are taken into account to assess *markerhood*:

- **Frequency** of occurrence of the DM candidate in corpus, normalised by its length in words and exclusive of stopwords. Normalisation is achieved by the function $normalised\ frequency = length \cdot \log(frequency)$.

[2] We worked with 784 expanded forms corresponding to 577 basic cue phrases

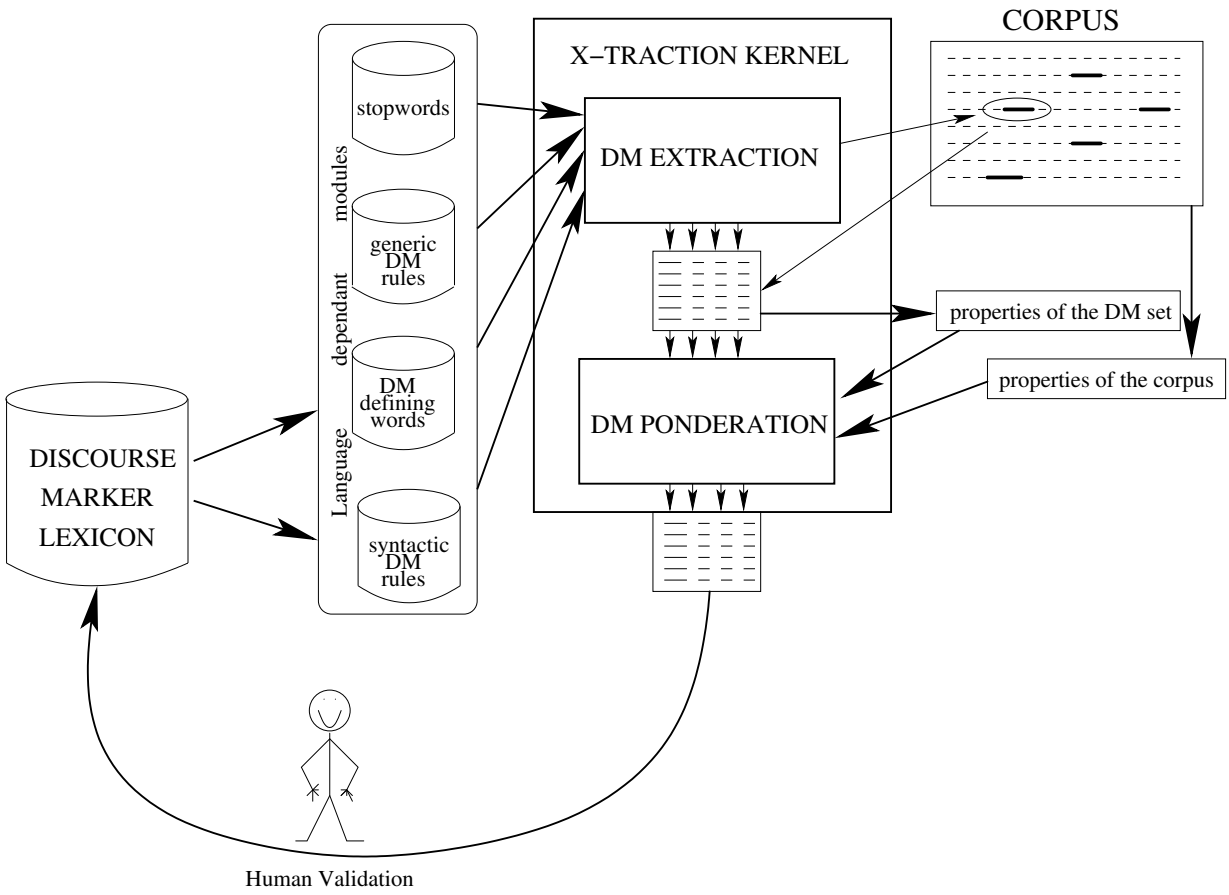


Figure 1: Architecture of X-Tractor

DM	boundary	syntactic type	rhetorical type	direction	con tent
además	not appl.	adverbial	satellizer	inclusion	reinforcement
a pesar de	strong	preposition	satellizer	right	concession
así que	weak	subordinating	chainer	right	consequence
dado que	weak	subordinating	satellizer	right	enablement

Table 1: Sample of the cue phrase lexicon

- Frequency of occurrence in **discursively salient context**. Discursively salient contexts are preferred occurrence locations for DMs. This parameter has been combined with DM classes motivated by clustering in (Alonso et al., 2002).
- **Mutual Information** of the words forming the DM candidate. Word strings with higher mutual information are supposed to be more plausible lexical units.
- **Internal Structure** of the DM, that is to say, whether it follows one of the rules of combination of DM-words. For this application, X-TRACTOR was aimed at obtaining DMs other than those already in the starting lexicon, therefore, longer well-structured DM candidates were priorised, that is to say, the longer the rule that a DM candidate satisfies, the higher the value of this parameter.
- **Rhetorical Content** of the DM candidate is increased by the number of words with strong rhetorical content it contains. These words are listed in one of the modules of external knowledge, and each has a rhetorical content associated to them. This rhetorical content can be pre-assigned to the DM candidate for the human expert to validate.
- **Lexical Weight** accounts for the the presence of non frequent words in the DM candidate. Unfrequent words make a DM with high *markerhood* more likely as a segment boundary marker.
- **Linking Function** of the DM candidate accounts for its power to link spans of text, mostly by reference.
- **Length** of the DM candidate is relevant for obtaining new DMs if we take into consideration the fact that DMs tend to aggregate.

These parameters are combined by weighted voting for *markerhood* assessment, so that the importance of each of them for the final *markerhood* assessment can be adapted

```
for each word in string
        if word is a preposition, then
          if word-1 is an adverb, then
            if word-2 is a coordinating conjunction, then
              if word+1 is a rhetorical-content word, then
                if word+2 is a preposition, then
                  assign the DM candidate structural weight 5
                elsif word+2 is a subordinating conjunction, then
                  assign the DM candidate structural weight 5
                else assign the DM candidate structural weight 4
              elsif word+1 is a pronoun, then
                assign the DM candidate structural weight 4
            else assign the DM candidate structural weight 3
```

Figure 2: Example of rules for combination of DM-constituing words

to different targets. By assigning a different weight to each one of these parameters, the system can be used for extracting DMs useful for heterogeneous tasks, for example, automated summarisation, anaphora resolution, information extraction, etc.

In the application to Spanish, we were looking for DMs that signal discourse structure useful for automated text summarisation, that is to say, mostly indicators of relevance and coherence relations.

3. Results and Discussion

We ran X-TRACTOR on a sample totalling 350,000 words of Spanish newspaper corpus, and obtained a ranked list of DMs together with information about their syntactical type, rhetorical content and an indication of their potential as segment boundary markers. Only 372 out of the 577 DMs in the DM lexicon could be found in this sample, which indicates that a bigger corpus would provide a better picture of DMs in the language, as will be developed below.

3.1. Evaluation of Results

Evaluation of lexical acquisition systems is a problem still to be solved. Typically, the metrics used are standard IR metrics, namely, *precision* and *recall* of the terms retrieved by an extraction tool evaluated against a document or collection of documents where terms have been identified by human experts (Vivaldi, 2001). Precision accounts for the number of term candidates extracted by the system which have been identified as terms in the corpus, while recall states how many terms in the corpus have been correctly extracted.

This kind of evaluation presents two main problems: first, the bottleneck of hand-tagged data, because a large-scale evaluation implies a costly effort and a long time for manually tagging the evaluation corpus. Secondly, since terms are not well-defined, there is a significant variability between judges, which makes it difficult to evaluate against a sound golden standard.

For the evaluation of DM extraction, these two problems become almost unsolvable. In the first place, DM density in corpus is far lower than term density, which implies that judges should read a huge amount of corpus to identify a number of DMs significant for evaluation. In practical terms, this is almost unaffordable. Moreover, X-TRACTOR's performance is optimised for dealing with huge amounts of corpus. On the other hand, the lack of a reference concept for DM makes inter-judge variability for DM identification even higher than for term identification.

Given these difficulties, we have carried out an alternative evaluation of the presented application of the system. To give a hint of the recall of the obtained DM candidate list, we have found how many of the DMs in the DM lexicon were extracted by X-TRACTOR, and how many of the DM candidates extracted were DMs in the lexicon[3]. To evaluate the goodness of *markerhood* assessment, we have found the ratio of DMs in the lexicon that could be found among the first 100 and 1000 highest ranked DM candidates given by X-TRACTOR. To evaluate the enhancement of the initial set of DMs that was achieved, the 100 highest ranked DMs were manually revised, and we obtained the ratio of actual DMs or strings containing DMs that were not in the DM lexicon. Noise has been calculated as the ratio of non-DMs that can be found among the 100 highest ranked DM candidates.

3.2. Parameter Tuning

To roughly determine which were the parameters more useful for finding the kind of DMs targeted in the presented application, we evaluated the goodness of each single parameter by obtaining the ratio of DMs in the lexicon that could be found within the 100 and 1000 DM candidates ranked highest by that parameter.

In Figure 3 it can be seen that the parameters with best behaviours in isolation are *content*, *structure*, *lexical weight* and *occurrence in pausal context*, although none of them performs above a dummy baseline fed with the same corpus sample. This baseline extracted 1- to 4-word strings after punctuation signs, and ranked them according to their frequency, so that the most frequent were ranked highest. Frequencies of strings were normalised by length, so that $normalised\ frequency = length \cdot \log(frequency)$. Moreover, the frequency of strings containing stopwords was reduced.

[3]We previously checked how many of the DMs in the lexicon could actually be found in corpus, and found that only 386 of them occurred in the 350,000 word sample; this is the upper bound of in-lexicon DM extraction.

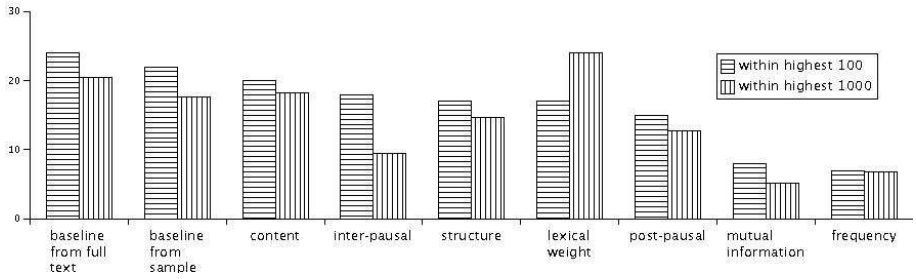


Figure 3: Ratio of DM andidates that contain a DM in the lexicon among the 100 and 1000 highest ranked by each individual parameter

	baseline	X-TRACTOR
Coverage of the DM lexicon	88%	87.5%
ratio of DMs in the lexicon		
within 100 highest ranked	31%	41%
within 1000 highest ranked	21%	21.6%
Noise		
within the 100 highest ranked	57%	32%
Enhancement Ratio		
within the 100 highest ranked	9%	15%

Table 2: Results obtained by X-TRACTOR and the baseline

However, the same dummy baseline performed better when fed with the whole of the newspaper corpus, consisting of 3,5 million words. This, and the bad performance of the parameters that are more dependant on corpus size, like *frequency* and *mutual information*, clearly indicates that the performance of X-TRACTOR, at least for this particular task, will tend to improve when dealing with huge amounts of corpus. This is probably due to the data sparseness that affects DMs.

This evaluation provided a rough intuition of the goodness of each of the parameters, but it failed to capture interactions beteween them. To assess that, we evaluated combinations of parameters by comparing them with the lexicon. We finally came to the conclusion that, for this task, the most useful parameter combination consisted in assigning a very high weight to structural and discourse-contextual information, and a relatively important weight to content and lengh, while no weight at all was assigned to frequency or mutual information. This combination of parameters also provides an empirical approach to the delimitation of the concept of DM, by eliciting the most influential among a set of DM-characterising features.

However, the evaluation of parameters failed to capture the number of DMs non present in the lexicon retrieved by each parameter or combination of parameters. To do that, the highest ranked DM candidates of each of the lists obtained for each parameter or parameter combination should have been revised manually. That's why only the best combinations of parameters were evaluated as to the enhancement of the lexicon they provided.

3.3. Results with combined parameters

In Table 2 the results of the evaluation of X-TRACTOR and the mentioned baseline are presented. From the sample of 350,000 words, the baseline obtained a list of 60,155 DM candidates, while X-TRACTOR proposed 269,824. Obviously, not all of these were actual DMs, but both systems present an 88% coverage of the DMs in the lexicon that are present in this corpus sample, which were 372.

Concerning goodness of DM assessment, it can be seen that 43% of the 100 DM candidates ranked highest by the baseline were or contained actual DMs, while X-TRACTOR achieved a 68%. Out of these, the baseline succeeded in identifying a 9% of DMs that were not in the lexicon, while X-TRACTOR identified a 15%. Moreover, X-TRACTOR identified an 8% of temporal expressions. The fact that they are identified by the same features characterising DMs indicates that they are very likely to be treated in the same way, in spite of heterogeneous discursive content.

In general terms, it can be said that, for this task, X-TRACTOR outperformed the baseline, suceeded in enlarging an initial DM lexicon and obtained quality results and low noise. It seems clear, however, that the dummy baseline is useful for locating DMs in text, although it provides a limited number of them.

4. Conclusions and Future Directions

By this application of X-TRACTOR to a DM extraction task for Spanish, we have shown that bootstrap-based lexical acquisition is a valid method for enhancing a lexicon of DMs, thus improving the limited coverage of the starting resource. The resulting lexicon exploits the properties of the input corpus, so it is highly portable to restricted domains. This high portability can be understood as an equivalent of domain independence.

The use of this empirical methodology circumvents the bias of human judges, and elicits the contribution of a number of parameters to the identification of DMs. Therefore, it can be considered as a data-driven delimitation of the concept of DM. However, the impact of the enhancement obtained by bootstraping the lexicon should be assessed in terms of prototypicality, that is to say, it should be studied how enlarging a starting set of clearly protoypical DMs

may lead to finding less and less prototypical DMs. For an approach to DM prototypicality, see (Alonso et al., 2002).

Future improvements of this tool include applying techinques for interpolation of variables, so that the tuning of the parameters for *markerhood* assessment can be carried out automatically. Also the process of rule induction from the lexicon to the rule module can be automatised, given classes of DM-constituting-words and classes of DMs. Moreover, it has to be evaluated in bigger corpora.

Another line of work consists in exploiting other kinds of knowledge for DM extraction and ponderation. For example, annotated corpora could be used as input, tagged with morphological, syntactical, semantic or even discursive information. The resulting DM candidate list could be pruned by removing proper nouns from it, for example, with the aid of a proper noun data base or *gazetteer* (Arévalo et al., 2002).

To test the portability of the system, it should be applied to other tasks and languages. An experiment to build a DM lexicon for Catalan is currently under progress. To do that, we will try to alternative strategies: one, translating the linguistic knowledge modules to Catalan and directly applying X-TRACTOR to a Catalan corpus, and another, obtaining an initial lexicon by applying the dummy baseline presented here and carrying out the whole bootstrap process.

5. Acknowledgements

This research has been conducted thanks to a grant associated to the X-TRACT project, PB98-1226 of the Spanish Research Department. It has also been partially funded by projects HERMES (TIC2000-0335-C03-02) and PETRA (TIC2000-1735-C02-02).

6. References

Laura Alonso, Irene Castellón, Lluís Padró, and Karina Gibert. 2002. Clustering discourse markers. submitted.

Montse Arévalo, Xavi Carreras, Lluís Màrquez, M.Antònia Martí, Lluís Padró, and M.José Simón. 2002. A proposal for wide-coverage spanish named entity recognition. Technical Report LSI-02-30-R, Dept. LSI, Universitat Politècnica de Catalunya, Barcelona, Spain.

Robert Dale and Alistair Knott. 1995. Using linguistic phenomena to motivate a set of coherence relations. *Discourse Processes*, 18(1):35–62.

C. Engehard and L. Pantera. 1994. Automatic natural acquisition of a terminology. *Journal of Quantitative Linguistics*, 2(1):27–32.

Kyo Kageura and Bin Umino. 1996. Methods of automatic term recognition: A review. *Terminolgy*, 3(2):259–289.

Jung Hee Kim, Michael Glass, and Martha W. Evens. 2000. Learning use of discourse markers in tutorial dialogue for an intelligent tutoring system. In *COGSCI 2000, Proceedings of the 22nd Annual Meeting of the Cognitive Science Society*, Philadelphia, PA.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organisation. *Text*, 3(8):234–281.

Daniel Marcu. 1997. From discourse structures to text summaries. In Mani and Maybury, editors, *Advances in Automatic Text Summarization*, pages 82 – 88.

Jorge Vivaldi. 2001. *Extracción de candidatos a término mediante combinación de estrategias heterog éneas*. Ph.D. thesis, Departament de Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya.

Creating and Using Semantics for Information Retrieval and Filtering

State of the Art and Future Research

Several experiments have been carried out in the last 15 years investigating the use of various resources and techniques (e.g., thesauri, synonyms, word sense disambiguation, etc.) to help refine or enhance queries. However, the conclusions drawn on the basis of these experiments vary widely. Results of some studies have led to the conclusion that semantic information serves no purpose and even degrades results, while others have concluded that the use of semantic information drawn from external resources significantly increases the performance of retrieval software. At this point, several question arise:

- Why do these conclusions vary so widely?
- Is the divergence a result of differences in methodology?
- Is the divergence a result of a difference in resources? What are the most suitable resources?
- Do results using manually constructed resources differ in significant ways from results using automatically extracted information?
- From corpus building to terminology structuring, to which methodological requirements resources acquisition has to comply with in order to be relevant to a given application?
- What is the contribution of specialized resources?
- Are present frameworks for evaluation (e.g., TREC) appropriate
- for evaluation of results?.

These questions are fundamental not only to research in document retrieval, but also for information searching, question answering, filtering, etc. Their importance is even more acute for multilingual applications, where, for instance, the question of whether to disambiguate before translating is fundamental.

Moreover, the increasing diversity of monolingual as well as multilingual documents on the Web invite to focus attention on lexical variability in connection with textual genre and with questioning the resources reusability stance.

The goal of this workshop is to bring together researchers in the domain of document retrieval, and in particular, researchers on both sides of the question of the utility of enhancing queries with semantic information gleaned from languages resources and processes.

The workshop will provide a forum for presentation of the different points of view, followed by a roundtable in which the participants will assess the state of the art, consider the results of past and on-going work and the possible reasons for the considerable differences in their conclusions. Ultimately, they will attempt to identify future directions for research.

Workshop Organisers

Christian Fluhr, CEA, France
Nancy Ide, Vassar College, USA
Claude de Loupy, Sinequa, France
Adeline Nazarenko, LIPN, Université de Paris-Nord, France
Monique Slodzian, CRIM, INALCO, France

Workshop Programme Committee

Table of Contents

Finding Similar Documents in Document Collections

Thorsten Brants and Reinhard Stolle

Palo Alto Research Center (PARC)
3333 Coyote Hill Rd, Palo Alto, CA 94304, USA
{brants,stolle}@parc.com

Abstract

Finding similar documents in natural language document collections is a difficult task that requires general and domain-specific world knowledge, deep analysis of the documents, and inference. However, a large portion of the pairs of similar documents can be identified by simpler, purely word-based methods. We show the use of Probabilistic Latent Semantic Analysis for finding similar documents. We evaluate our system on a collection of photocopier repair tips. Among the 100 top-ranked pairs, 88 are true positives. A manual analysis of the 12 false positives suggests the use of more semantic information in the retrieval model.

1. Introduction

Collections of natural language documents that are focused on a particular subject domain are commonly used by communities of practice in order to capture and share knowledge. Examples of such "focused document collections" are FAQs, bug-report repositories, and lessons-learned systems. As such systems become larger and larger, their authors, users and maintainers increasingly need tools to perform their tasks, such as browsing, searching, manipulating, analyzing and managing the collection. In particular, the document collections become unwieldy and ultimately unusable if obsolete and redundant content is not continually identified and removed.

We are working with such a knowledge-sharing system, focused on the repair of photocopiers. It now contains about 40,000 technician-authored free text documents, in the form of tips on issues not covered in the official manuals. Such systems usually support a number of tasks that help maintain the utility and quality of the document collection. Simple tools, such as keyword search, for example, can be extremely useful. Eventually, however, we would like to provide a suite of tools that support a variety of tasks, ranging from simple keyword search to more elaborate tasks such as the identification of "duplicates." Fig. 1 shows a pair of similar tips from our corpus. These two tips are about the same problem, and they give a similar analysis as to why the problem occurs. However, they suggest different solutions: Tip 118 is the "official" solution, whereas Tip 57 suggests a short-term "work-around" fix to the problem. This example illustrates that "similarity" is a complicated notion that cannot always be measured along a one-dimensional scale. Whether two or more documents should be considered "redundant" critically depends on the task at hand. In the example of Fig. 1, the work-around tip may seem redundant and obsolete to a technician who has the official new safety cable available. In the absence of this official part, however, the work-around tip may be a crucial piece of information.

Our goal is to develop techniques that analyze the conceptual contents of natural language documents at a granularity that is fine enough to capture distinctions like the one between Tips 57 and 118, described in the previous paragraph. In order to do that, we are designing formal representations of document contents that will allow us to assess not only whether two documents are about the same subject but also whether two documents actually say the same thing. We are currently focusing on the tasks of computer-assisted redundancy resolution. We hope that our techniques will eventually extend to support even more ambitious tasks such as the identification and resolution of inconsistent knowledge, knowledge fusion, question answering, and trend analysis.

We believe that, in general, the automated or computer-assisted management of collections of natural language documents requires a fine-grained analysis and representation of the documents' contents. This fine granularity in turn mandates deep linguistic processing of the text and inference capabilities using extensive linguistic and world knowledge. Following this approach, our larger research group has implemented a prototype, which we will briefly describe in the next section. This research prototype system is far from complete. Meanwhile, we are investigating to what extent currently operational techniques are useful to support at least some of the tasks that arise from the maintenance of focused document collections. We have investigated the utility of Probabilistic Latent Semantic Analysis (PLSA) (Hofmann, 1999b) for the task of finding similar documents. Section 3. describes our PLSA model and Section 4. reports on our experimental results in the context of our corpus of repair tips. In that section, we also attempt to characterize the types of similarities that are easily detected and contrast them to the types that are easily missed by the PLSA technique. Finally, we speculate how symbolic knowledge representation and inference techniques that rely on a deep linguistic analysis of the documents may be coupled with statistical techniques in order to improve the results.

2. Knowledge-Based Approach

Our goal is to build a system that supports a wide range of knowledge management tasks for focused document collections. We believe that powerful tools for tasks like redundancy resolution, topic browsing, question answering, knowledge fusion, and so on, need to analyze and represent the documents' conceptual contents at a fine level of granularity.

Concentrating on the task of redundancy resolution, our

	Tip 57		Tip 118
Problem:	Left cover damage	Problem:	The current safety cable used in the 5100 Document Handler fails prematurely, causing the Left Document Handler Cover to break.
Cause:	The left cover safety cable is breaking, allowing the left cover to pivot too far, breaking the cover.	Cause:	The plastic jacket made the cable too stiff. This causes stress to be concentrated on the cable ends, where it eventually snaps.
Solution:	Remove the plastic sleeve from around the cable. Cutting the plastic off of the cable makes the cable more flexible, which prevents cable breakage. Cable breakage is a major source of damage to the left cover.	Solution:	When the old safety cable fails, replace it with the new one, which has the plastic jacket shortened.

Figure 1: Example of Eureka tips

project group has so far built a prototype whose goal is to identify conceptually similar documents, regardless of how they are written. This task requires extensive knowledge about language and of the world. Since most of this knowledge engineering effort is performed by hand at the moment, our system's coverage is currently limited to fifteen pairs of similar tips. We are in the process of scaling the system up by one to two orders of magnitude. Eventually, we hope to also support more general tasks, namely identify the parts of two documents that overlap; and identify parts of the documents that stand in some relation to each other, such as expanding on a particular topic or being in mutual contradiction. Such a system will enable the maintenance of vast document collections by identifying potential redundancies or inconsistencies for human attention.

State-of-the-art question answering and information extraction techniques (e.g., (Bear et al., 1997)) are sometimes able to identify entities and the relations between them at a fine level of granularity. However, the functionality and coverage of these techniques is typically restricted to a limited set of types of entities and relations that have been formalized upfront using static templates. Like a small number of other research projects (e.g., the TACITUS project (Hobbs et al., 1993)), our approach is based on the belief that the key to solving this problem is a principled technique for producing formal representations of the conceptual contents of the natural language documents. In our approach, a deep analysis based on Lexical Functional Grammar theory (Kaplan and Bresnan, 1982) combined with Glue Semantics (Dalrymple, 1999) produces a compact representation of the syntactic and semantic structures for each sentence. From this language-driven representation of the text, we map to a knowledge-driven representation of the contents that abstracts away from the particular natural language expression. This mapping includes several—not necessarily sequential—steps. In one step, we rely on a domain-specific ontology to identify canonicalized entities and events that are talked about in the text. In our case, these entities and events include things like parts, e.g., photoreceptor belt, and relevant activities such as cleaning, for example. Another step performs thematic role assignments and assembles fragments of conceptual structures from the normalized entities and events (e.g., cleaning a photoreceptor belt). Furthermore, certain relations are normalized; for example, "stiff" and "flexible" (in Fig. 1) both refer to the rigidity of an object, one being the inverse of the other. Yet another step composes structure fragments into higher-level structures that reflect causal or temporal relations, such as action sequences or repair plans. All steps involve ambiguity resolution as a central problem, which requires inference based on extensive linguistic and world knowledge. For a more detailed description of this approach and its scalability, see (Crouch et al., 2002).

Finally, we assess the similarity of two documents using a variant of the Structure Mapping Engine (SME) (Forbus et al., 1989). SME anchors its matching process in identical elements that occur in the same structural positions in the base and target representations, and from this builds a correspondence. The larger the structure that can be recursively constructed in this manner, while preserving a systematicity constraint of one-to-one correspondence between base and target elements and the identicality of anchors, the greater the similarity score.

We expect that the fine-grained conceptual representations discussed in this section will eventually enable our system to detect whether two documents are not only about the same subject but also saying the same thing. Many interesting cases of similarity can, however, be detected with lighter-weight techniques. This is the topic of the next section.

3. The Word-Based Statistical Model

While in the general case deep processing, knowledge about the world, and inference are necessary to identify similar documents, there may be a large number of similar pair that can be discovered by a shallow approach. We now view the task of finding similar pairs of documents as an information retrieval problem where documents are matched based on the words that occur in the documents, i.e., we use a vector space model of the documents. Comparison is done using Probabilistic Latent Semantic Analysis (PLSA) (Hofmann, 1999b).

3.1. Document Preprocessing

Each document is first preprocessed by:

1. Separating the document fields. Each tip usually comes with additional administrative information like author, submission date, location, status, contact information, etc. We extract the information that is contained in the `CHAINS`, `PROBLEM`, `CAUSE`, and `SO-`

LUTION fields[1].

2. Tokenizing the document. Words and numbers are separated at white space, punctuation is stripped, abbreviations are recognized.

3. Lemmatizing each token, i.e., each word is uniquely mapped to a base form. We use the LinguistX lemmatizer[2] to perform this task.

Steps 1 to 3 identify the terms in the vocabulary. We select the subset of those terms that occur in at least two documents. Given this vocabulary, each document d is represented by its term-frequency vector $f(d,w)$, where w are the terms of the document.

3.2. Probabilistic Latent Semantic Indexing

Probabilistic Latent Semantic Analysis (PLSA) is a statistical latent class model or aspect model (Hofmann, 1999a; Hofmann, 1999b). It can be seen as a statistical view of Latent Semantic Analysis (LSA) (Deerwester et al., 1990). The model is fitted to a training corpus by the Expectation Maximization (EM) algorithm (Dempster et al., 1977). It assigns probability distributions over classes to words and documents and thereby allows them to belong to more than one class, and not to only one class as is true of most other classification methods. PLSA represents the joint probability of a document d and a word w based on a latent class variable z:[3]

$$P(d,w) = P(d)\sum_z P(w|z)P(z|d) \tag{1}$$

The model makes an independence assumption between word w and document d if the latent class z is given, i.e., $P(w|z,d) = P(w|z)$. PLSA has the following view of how a document is generated: first a document $d \in \mathcal{D}$ (i.e., its dummy label) is chosen with probability $P(d)$. For each word in document d, a latent topic $z \in \mathcal{Z}$ is chosen with probability $P(z|d)$, which in turn is used to choose the word $w \in \mathcal{W}$ with probability $P(w|z)$.

A model is fitted to a document collection $\mathcal{D}$ by maximizing the log-likelihood function $\mathcal{L}$:

$$\mathcal{L} = \sum_{d\in\mathcal{D}}\sum_{w\in d} f(d,w) log P(d,w) \tag{2}$$

The E-step in the EM-algorithm is

$$P(z|d,w) = \frac{P(z)P(d|z)P(w|z)}{\sum_{z'} P(z')P(d|z')P(w|z')} \tag{3}$$

and the M-step consists of

$$P(w|z) = \frac{\sum_d f(d,w)P(z|d,w)}{\sum_{d,w'} f(d,w')P(z|d,w')} \tag{4}$$

$$P(d|z) = \frac{\sum_w f(d,w)P(z|d,w)}{\sum_{d',w} f(d',w)P(z|d',w)} \tag{5}$$

$$P(z) = \frac{\sum_{d,w} f(d,w)P(z|d,w)}{\sum_{d,w} f(d,w)} \tag{6}$$

The parameters are either randomly initialized or according to some prior knowledge.

After having calculated the reduced dimensional representations of documents in the collection, we map the vectors back to the original term space to yield vectors $P(w|d)$ by

$$P(w|d) = \sum_z P(w|z)P(z|d) \tag{7}$$

$P(w|d)$ can be seen as a smoothed version of the empirical distribution $r(w|d) = f(d,w)/f(d)$ of words in the document. The advantage of the smoothed version is that it captures semantic similarities through the lower-dimensional representation.

Note that this process is intended for the pairwise comparison of all documents in the training collection. It can be extended to new documents q (query or test documents) by using the folding-in process. Folding-in uses Expectation-Maximization as in the training process; the E-step is identical, the M-step keeps all the $P(w|z)$ constant and recalculates $P_{fi}(z|q)$. Usually, a very small number of iterations is sufficient for folding-in. We get a smoothed representation of a folded-in document by

$$P_{fi}(w|q) = \sum_z P(w|z)P_{fi}(z|q) \tag{8}$$

This corresponds to the PLSI-U model described in (Hofmann, 1999b).

3.3. Document Comparison

A standard way of comparing vector space representations of documents d_1 and d_2 is to calculate the cosine similarity score of tf-idf weighted document vectors (Salton, 1988):

$$\mathrm{sim}_{\cos}(d_1,d_2) = \frac{\sum_w \hat{f}(d_1,w)\hat{f}(d_2,w)}{\sqrt{\sum_w \hat{f}(d_1,w)^2}\sqrt{\sum_w \hat{f}(d_2,w)^2}} \tag{9}$$

$\hat{f}(d,w)$ is the weighted frequency of word w in document d:

$$\hat{f}(d,w) = f(d,w)\log\frac{N}{df(w)} \tag{10}$$

where N is the total number of documents, and $df(w)$ is the number of documents containing word w.

We additionally perform the comparison on the PLSA representation of $P(w|d)$. Pairwise comparisons are done by

$$\mathrm{sim}^{\mathrm{PLSA}}_{\cos}(d_1,d_2) = \frac{\sum_w P(w|d_1)P(w|d_2)}{\sqrt{\sum_w P(w|d_1)^2}\sqrt{\sum_w P(w|d_2)^2}} \tag{11}$$

[1]The CHAINS field contains a numerical identifier of the product line.

[2]For information about the LinguistX tools, see www.inxight.com/products/linguistx/

[3]Unless otherwise noted, we use the following notational conventions: training documents $d, d' \in \mathcal{D}$, test documents $q, q' \in \mathcal{Q}$, words $w, w' \in \mathcal{W}$, and classes $z, z' \in \mathcal{Z}$.

Table 1: Precision of the statistical model for the n top-ranked pairs. A pair of tips is considered a "true positive" if their conceptual contents are categorized to be the same, similar, or in the subset relationship.

n	precision
10	100%
20	100%
30	100%
40	96%
50	92%
60	92%
70	90%
80	87%
90	88%
100	88%

Both similarities are combined with a weight λ to yield the final similarity score (see (Hofmann, 1999b)).

$$\mathrm{sim}(d_1, d_2) = \lambda \mathrm{sim}_{\cos}(d_1, d_2) + (1 - \lambda)\mathrm{sim}_{\cos}^{\mathrm{PLSA}}(d_1, d_2) \quad (12)$$

The output of the algorithm is a list of pairs ranked according to their similarity.

4. Experiments

We applied the algorithm described in Section 3. to a subset of the Eureka database consisting of 1,321 tips. PLSA representations of $P(w|d)$ were created for each tip, and pairs of tips were ranked according to their similarity. Following (Hofmann, 1999b), we created models with $Z = 32, 48, 64, 80, 128$ latent classes, calculated the average $P(w|d)$. The similarity score was combined with the standard tf-idf cosine similarity with a weight of $\lambda = \frac{1}{6}$.

4.1. Precision and Recall

We manually inspected the 100 top-ranked pairs of tips and classified their similarity by hand according to the types of similarity described in Section 4.2.. The results are shown in Table 1. Of the 10 top-ranked pairs, all 10 were actual duplicates,[4] of the 40 top-ranked pairs, 96% were true positives, and so on. The manual inspection of the 100 top-ranked pairs (of the potential 871,860 pairs) revealed 88 true positives.

Independent manual sampling of the subset of 1,321 tips, which is a very tedious and time-consuming task, revealed 17 similar pairs (14 pairs and 1 triple). 3 of these pairs were among the top 100 emitted by the word-based statistical model. This is a recall of 18% on the manually identified similar pairs. However, it is unclear how this number relates to the overall recall because the distribution of the other similar pairs is currently unclear.

[4] A pair of tips is considered "duplicates" if their conceptual contents are categorized to be the same. A pair of tips is considered a "true positive" if their conceptual contents are categorized to be the same, similar, or in the subset relationship. See Section 4.2..

Table 2: Number of pairs with structural and conceptual match in the 100 top-ranked pairs of documents. We are interested in finding the conceptually same/similar/subset pairs. False positives are shown in *italics*.

		conceptual				
		same	sim	subset	diff	sum
surface	same	24	0	10	*2*	36
	sim	17	24	13	*8*	62
	diff	0	0	0	*2*	2
	sum	41	24	23	*12*	100

4.2. Types of Similarity

The word-based statistical model of Section 3. seems to be good at identifying pairs whose texts are similar *at a surface level*. In order to see how well the model does at identifying pairs whose contents are *conceptually similar*, we manually performed a qualitative evaluation and classified each of the 100 top-ranked pairs according to the following criteria:

Surface similarity of texts: *same, similar, different*. Surface similarity describes the similarity of the set of words and syntactic constructions used in the documents. *Same* means that the documents are (almost) identical. *Similar* means that some words may be different or replaced by synonyms (e.g., "fault" vs. "failure" vs. "problem", "motor" vs. "drive", "line" vs. "wire", etc.), constructions are different, order of sentences may be different. *Different* means that the texts are different.

Conceptual similarity of contents: *same, similar, subset, different*. Conceptual similarity refers to the semantic/conceptual contents of the document, independent of how it is expressed as surface text. *Same* means that the documents have (almost) the same contents (e.g., "cutting the plastic off of the cable makes the cable more flexible" vs. "the plastic jacket made the cable too stiff"). *Similar* means that there is a significant overlap of conceptual contents between the two documents; for example, the tips describe the same problem but suggest different solutions (see Fig. 1), or, the tips describe an analogous problem exhibited at different mechanical parts (see Fig. 2).

Subset describes cases where the conceptual contents of one document form a proper subset of the conceptual contents of the other document—for example, if one document elaborates on the other. *Different* describes conceptually different documents.

Table 2 shows how many of the pairs fall into the different categories. Since the PSLA model is word-based, almost none of the pairs have different surface similarity. In the 100 top-ranked pairs, the majority of false positives occur when the surface texts are similar but the conceptual contents are different (8 out of 12).

The algorithm identifies surface similarity very well, only 2 out of 100 pairs are different at the surface text level.

Tip 690		Tip 714	
Problem:	08-110, Tray 3 misfeed	Problem:	08-100, Tray 1 misfeed
Cause:	J201 Pin 1 loose. Drive coupling set screw loose, Blower hose came off, Fang plate out of adjustment, Stack height out of adjustment, Defective DRCC1.	Cause:	Set screw on feed clutch loose. Stack height sensor out of bracket. Feeder drive coupling loose. Blower hose off.
Solution:	Reseat J201 Pin 1. Tighten drive coupling, Reconnect blower hose , Adjust fang plate, Adjust stack height. Replace DRCC1.	Solution:	Adjust clutch. Repair stack height sensor. Tighten feeder driv e coupling. Repair blower hose.

Figure 2: True positive: this pair at rank 68 has similar surface text and is similar at the conceptual level.

Tip 1280		Tip 1281	
Problem:	Xerox Binder 120. The "READY FOR AUTO FEED" message does not change when set clamp assy is pulled in	Problem:	Xerox Binder 120. The Binder 120 does not display "Ready for auto feed" message.
Cause:	Set Clamp extended sensor (Q23) is "H" all the time	Cause:	Set Clamp extended sensor (Q23) is "Lo" all the time
Solution:	check the set clamp sensor wires for an open circuit, if ok, Replace the set clamp extended sensor (Q23)	Solution:	Check the set clamp extended sensor wires for Short circuit to frame, Set clamp out flag is in the sensor correctly, if ok, replace the sensor.

Figure 3: False positive: this pair at rank 37 has almost the same surface text but is different at the conceptual level.

These two pairs involve very long documents (average of 1030 tokens per document compared to 132 tokens per document overall average). The documents have an overlap in vocabulary, but the sentences and sequences of sentences are very different.

Correlation with conceptual similarity can also be found, but it is smaller. 10 out of 100 pairs were categorized as the same or similar at the surface but are conceptually different; from the viewpoint of a user in the context of a conceptual task, these pairs should not be identified as similar tips. We believe that a deeper analysis of the document contents as outlined in Section 2. will help distinguish between conceptually different documents and, therefore, reduce the number of such false positives.

One of the two pairs that are almost the same at the surface level but have different conceptual contents is shown in Fig. 3.

They use the same or very similar words, but make opposite statements at the conceptual level. Tip 1280 describes a sensor signal that is erroneously "high" because of an open circuit. Tip 1281 describes a sensor signal that is erroneously "low" because of a short circuit. This difference cannot be found by the word-based statistical model. The topics of these two documents are very similar; however, a correct analysis of the contents requires the recognition of the difference between "does not display" and "does not change", the difference between "Lo" and "H", and the difference between "open circuit" and "short circut" despite the fact that these phrases often occur in similar contexts.

Fig. 4 shows a pair with similar surface texts but different conceptual contents. Tip 227 explains how to repair or prevent a particular failure that is caused by a ring's wearing out. Tip 173 says that an improved repair kit can be ordered; it also provides a work-around for the case in which that improved kit is not available.

The two examples in Figures 3 and 4 show that in many cases it is necessary to process the text more deeply than at the word level in order to be able to recognize fine-grained distinctions in the documents' contents. On the other hand, a large number of true positives are actually discovered by the word-based model (88 out of the 100 top-ranked pairs). The word-based statistical model even finds cases in which the conceptual contents are similar, but where this fact is not immediately obvious from the surface-level texts. Fig. 2 shows an example of this case. The two tips describe almost the same fault situation, except that one of them occurs in connection with Tray 1 while the other one occurs in connection with Tray 3. Even for a human—at least for an untrained human—, this pair is difficult to detect.

The examples suggests that symbolic and statistical techniques may be good at different tasks that complement each other nicely. Statistical techniques seem to be good at identifiying that the two tips are about the same topic. Knowledge-based techniques—specifically, a domain ontology—may help distinguish "Fuser Couplings" from the "Fuser Couplings and Shaft Repair Kit" (cf. Fig. 4), which in turn may trigger further distinctions between the two tips based on domain-specific knowledge. Similarly, the example in Fig. 3 suggests that a statistical analysis coupled with a limited normalization of relations that occur frequently in the domain may be a promising direction to pursue.

Fig. 5 shows the rank of a pair vs. its similarity. Our data set contains 1,321 documents, i.e., there are 871,860 pairs. Word-based similarity does not decrease linearly. There is a large drop at the beginning, then the curve is relatively flat, and it suddenly drops again at the very end. All of the manually found similar pairs (the 17 pairs described in Section 4.1.) are marked with a ∘ in the graph; they are among the first 7% (the lowest rank is 57,014). We do currently not

	Tip 173		Tip 227
Problem:	Improved Fuser Couplings 600K31031 Tag P-184. Broken calls when servicing failed Fuser Drive Couplings.	Problem:	Fuser Couplings and Shaft Repair Kit, 605K3950, Tag P-129. The retaining ring that holds the Fuser Assembly Drive Coupling in place wears out and falls off the shaft.
Cause:	The parts needed to repair a Fuser Drive failure are presently contained in two separate Kits. If the service representative does not have both Kits in inventory the service call is interrupted.	Cause:	The Fuser Assembly Drive Coupling rubs against the retaining ring as it turns.
Solution:	1. To repair Fuser Drive failures, order the new Fuser Couplings an d Shaft Repair Kit 600K31031, TAG P-184. This kit contains all the parts in Fuser Couplings and Shaft Repair Kit 605K3950 except that the improved Drive Coupling, issued separately in Kit 600K31030, has been substituted. 2. If you have 600K31030 as well as 605K3950 in inventory, these Kits can be salvaged to provide the same parts as the new Kit. Open 605K3950 and discard only the Fuser Drive Coupling, then use the Coupling contained in Kit number 600K31030 in its place.	Solution:	On the next service call check to see if P-129 is installed. If Tag P-129 is not installed, order and install the Fuser Couplings and Shaft Repair Kit, 605K3950.

Figure 4: False positive: this pair at rank 86 has similar surface text and is about similar parts, but is different at the conceptual level.

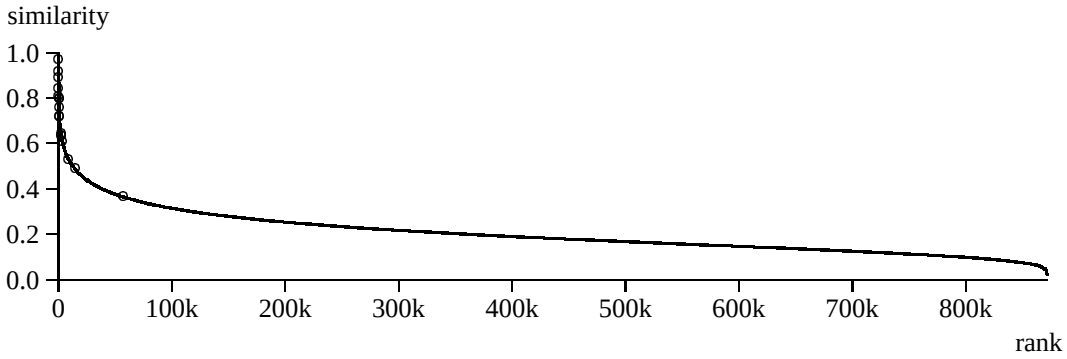


Figure 5: Rank vs. PLSA similarity. Manually found pairs are marked with ∘.

know whether there are any similar pairs below this rank, but it is probably safe to assume that almost all of the similar pairs are within the initial portion of the graph. Even if the presented statistical method does not rank all similar pairs at the very top, it seems to efficiently place them in a small initial segment at the top.

One focus of our current research effort is to understand the capabilities and limitations of the current PLSA model in order to design an improved system by, for example, (1) supplying the PLSA model with better-suited information for any given particular task, or (2) using the current version of the PLSA model as a prefilter for the knowledge-based approach.

5. Conclusions

We address the problem of matching the conceptual contents of documents. The domain of the documents in our experiments is the repair of photocopiers. In general, the problem requires world knowledge and deep processing of the documents. But in a large number of cases, similar documents can be found by shallow processing and a word-based statistical model. A quantitative evaluation shows that 88 of the 100 statistically top-ranked documents are true positives. An analysis of the erroneous cases indicates where the statistical model could benefit from deeper processing. Two important types of information that are currently absent from our statistical model are negation and relations between entities. We expect that providing the model with more semantic information along these lines will improve our system's performance and allow it to make finer distinctions among the documents' contents.

6. References

J. Bear, D. Israel, J. Petit, and D. Martin. 1997. Using information extraction to improve document retrieval. In E. M. Voorhees and D. K. Harman, editors, *The Sixth Text REtrieval Conference (TREC-6)*, pages 367–377. NIST.

R. Crouch, C. Condoravdi, R. Stolle, T. King, V. de Paiva, J. O. Everett, and D. G. Bobrow. 2002. Scalability of redundancy detection in focused document collections. In *Proceedings First International Workshop on Scalable Natural Language Understanding (SCANALU-2002)*, Heidelberg, Germany.

M. Dalrymple, editor. 1999. *Semantics and Syntax in Lexical Functional Grammar: The Resource Logic Approach*. MIT Press, Cambridge, MA.

S. Deerwester, S. Dumais, G. W. Furnas, T. Landauer, and R. Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407.

A. P. Dempster, N. M. Laird, and D. B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society*, 39(1):1–21.

K. D. Forbus, B. Falkenhainer, and D. Gentner. 1989. The structure mapping engine: Algorithm and examples. *Artificial Intelligence*, 41(1):1–63.

J. R. Hobbs, M. Stickel, S. Appelt, and P. Martin. 1993. Interpretation as abduction. *Artificial Intelligence*, 63(1–2):69–142.

T. Hofmann. 1999a. Probabilistic latent semantic analysis. In *Proceedings of Uncertainty in Artificial Intelligence*, Stockholm, Sweden.

T. Hofmann. 1999b. Probabilistic latent semantic indexing. In *Proceedings of SIGIR-99*, pages 35–44, Berkeley, CA.

R. M. Kaplan and J. Bresnan. 1982. Lexical-functional grammar: A formal system for grammatical representation. In J. Bresnan, editor, *The Mental Representation of Grammatical Relations*, pages 173–281. MIT Press, Cambridge, MA.

G. Salton. 1988. *Automatic Text Processing: The Transformation, Analysis, and Retrieval of Information by Computer*. Addison Wesley.

Robust Photo Retrieval Using World Semantics

Hugo Liu*, Henry Lieberman*

* MIT Media Laboratory
Software Agents Group
20 Ames St., E15-320G
Cambridge, MA 02139, USA
{hugo, lieber}@media.mit.edu

Abstract

Photos annotated with textual keywords can be thought of as resembling documents, and querying for photos by keywords is akin to the information retrieval done by search engines. A common approach to making IR more robust involves query expansion using a thesaurus or other lexical resource. The chief limitation is that keyword expansions tend to operate on a word level, and expanded keywords are generally lexically motivated rather than conceptually motivated. In our photo domain, we propose a mechanism for robust retrieval by expanding the *concepts* depicted in the photos, thus going beyond lexical-based expansion. Because photos often depict places, situations and events in everyday life, concepts depicted in photos such as place, event, and activity can be expanded based on our "common sense" notions of how concepts relate to each other in the real world. For example, given the concept "surfer" and our common sense knowledge that surfers can be found at the beach, we might provide the additional concepts: "beach", "waves", "ocean", and "surfboard". This paper presents a mechanism for robust photo retrieval by expanding annotations using a world semantic resource. The resource is automatically constructed from a large-scale freely available corpus of commonsense knowledge. We discuss the challenges of building a semantic resource from a noisy corpus and applying the resource appropriately to the task.

1. Introduction

The task described in this paper is the robust retrieval of annotated photos by a keyword query. By "annotated photos," we mean a photo accompanied by some metadata about the photo, such as keywords and phrases describing people, things, places, and activities depicted in the photo. By "robust retrieval," we mean that photos should be retrievable not just by the explicit keywords in the annotation, but also by other implicit keywords conceptually related to the event depicted in the photo.

In the retrieval sense, annotated photos behave similarly to documents because both contain text, which can be exploited by conventional IR techniques. In fact, the common query enrichment techniques such as thesaurus-based keyword expansion developed for document retrieval may be applied to the photo retrieval domain without modification.

However, keyword expansion using thesauri is limited in its usefulness because keywords expanded by their synonyms can still only retrieve documents directly related to the original keyword. Furthermore, naïve synonym expansion may actually contribute more noise to the query and negate what little benefit keyword expansion may add to the query, namely, if keywords cannot have their word sense disambiguated, then synonyms for all the word senses of a particular word may be used in the expansion, and this has the potential to retrieve many irrelevant documents.

1.1. Relevant Work

Attempting to overcome the limited usefulness of keyword expansion by synonyms, various researchers have tried to use slightly more sophisticated resources for query expansion. These include dictionary-like resources such as lexical semantic relations (Voorhees, 1994), and keyword co-occurrence statistics (Peat and Willet, 1991; Lin, 1998), as well as resources generated dynamically through relevance feedback, like global document analysis (Xu and Croft, 1996), and collaborative concept-based expansion (Klink, 2001).

Although some of these approaches are promising, they share some of the same problems as naïve synonym expansion. Dictionary-like resources such as WordNet (Fellbaum, 1998) and co-occurrence frequencies, although more sophisticated that just synonyms, still operate mostly on the word-level and suggest expansions that are lexically motivated rather than conceptually motivated. In the case of WordNet, lexical items are related through a very limited set of nymic relations. Relevance feedback, though somewhat more successful than dictionary approaches, requires additional iterations of user action and we cannot consider it fully automated retrieval, which makes it an inappropriate candidate for our task.

1.2. Photos vs. Documents

With regard to our domain of photo retrieval, we make a key observation about the difference between photos and documents, and we exploit this difference to make photo retrieval more robust. We make the observation that photos taken by an ordinary person has more structure and is more predictable than the average document on the web, even though that structure may not be immediately evident. The contents of a typical document such as a web page are hard to predict, because there are too many types and genres of web pages and the content does not predictably follow a stereotyped structure. However, with typical photos, such as one found in your photo album, there is more predictable structure. That is, the intended subject of photos often includes people and things in common social situations. Many of these situations depicted, such as weddings, vacations, sporting events, sightseeing, etc. are common to human experience, and therefore have a high level of predictability.

Take for example, a picture annotated with the keyword *"bride"*. Even without looking at the photo, a person may be able to successfully guess who else is in the photo, and what situation is being depicted. Common

sense would lead a person to reason that brides are usually found at weddings, that people found around her may be the groom, the father of the bride, bridesmaids, that weddings may take place in a chapel or church, that there may be a wedding cake, walking down the aisle, and a wedding reception. Of course, common sense cannot be used to predict the structure of specialty photos such as artistic or highly specialized photos; this paper only considers photos in the realm of consumer photography.

1.2.1. A Caveat

Before we proceed, it is important to point out that any semantic resource that attempts to encapsulate common knowledge about the everyday world is going to be somewhat culturally specific. The previous example of brides, churches and weddings illustrates an important point: knowledge that is *obvious* and *common* to one group of people (in this case, middle-class USA) may not be so obvious or common to other groups. With that in mind, we go on to define the properties of this semantic resource.

1.3. World Semantics

Knowledge about the spatial, temporal, and social relations of the everyday world is part of commonsense knowledge. We also call this *world semantics*, referring to the meaning of everyday concepts and how these concepts relate to each other in the world.

The mechanism we propose for robust photo retrieval uses a world semantic resource in order to expand concepts in existing photo annotations with concepts that are, *inter alia*, spatially, temporally, and socially related. More specifically, we automatically constructed our resource from a corpus of English sentences about commonsense by first extracting predicate argument structures, and then compiling those structures into a Concept Node Graph, where the nodes are commonsense concepts, and the weighted edges represent commonsense relations. The graph is structured much like MindNet (Richardson et al., 1998). Performing concept expansion using the graph is modeled as spreading activation (Salton and Buckley, 1988). The relevance of a concept is measured as the semantic proximity between nodes on the graph, and is affected by the strength of the links between nodes.

This paper is structured as follows: First, we discuss the source and nature of the corpus of commonsense knowledge used by our mechanism. Second, a discussion follows regarding how our world semantic resource was automatically constructed from the corpus. Third, we show the spreading activation strategy for robust photo retrieval, and give heuristics for coping with the noise and ambiguity of the knowledge. The paper concludes with a discussion of the larger system to which this mechanism belongs, potential application of this type of resource in other domains, and plans for future work.

2. OMCS: A Corpus of Common Sense

The source of the world semantic knowledge used by our mechanism is the Open Mind Common Sense Knowledge Base (OMCS) (Singh, 2002) - an endeavor at the MIT Media Laboratory that aims to allow a web-community of teachers to collaboratively build a database of "common sense" knowledge.

It is hard to define what actually constitutes common sense, but in general, one can think of it as knowledge about the everyday world that most people within some population consider to be "obvious." As stated earlier, common sense is somewhat culturally specific. Although many thousands of people from around the world collaboratively contribute to Open Mind Common Sense, the majority of the knowledge in the corpus reflects the cultural bias of middle-class USA. In the future, it may make sense to tag knowledge by their cultural specification.

OMCS contains over 400,000 semi-structured English sentences about commonsense, organized into an ontology of commonsense relations such as the following:

- A is a B
- You are likely to find A in/at B
- A is used for B

By semi-structured English, we mean that many of the sentences loosely follow one of 20 or so sentence patterns in the ontology. However, the words and phrases represented by A and B (see above) are not restricted. Some examples of sentences in the knowledge base are:

- Something you find in (a restaurant) is (a waiter)
- The last thing you do when (getting ready for bed) is (turning off the lights)
- While (acting in a play) you might (forget your lines)

The parentheses above denote the part of the sentence pattern that is unrestricted. While English sentence patterns has the advantage of making knowledge easy to gather from ordinary people, there are also problems associated with this. The major limitations of OMCS are four-fold. First, there is ambiguity resulting from the lack of disambiguated word senses, and from the inherent nature of natural languages. Second, many of the sentences are unusable because they may be too complex to fully parse with current parser technology. Third, because there is currently no truth maintenance mechanism or filtering strategy for the knowledge gathered (and such a mechanism is completely nontrivial to build), some of the knowledge may be anomalous, i.e. not common sense, or may plainly contradict other knowledge in the corpus. Fourth, in the acquisition process, there is no mechanism to ensure a broad coverage over many different topics and concepts, so some concepts may be more developed than others.

The Open Mind Commonsense Knowledge Base is often compared with its more famous counterpart, the CYC Knowledge Base (Lenat, 1998). CYC contains over 1,000,000 hand-entered rules that constitute "common sense". Unlike OMCS, CYC represents knowledge using formal logic, and ambiguity is minimized. In fact, it does not share any of the limitations mentioned for OMCS. Of course, the tradeoff is that whereas a community of non-experts contributes to OMCS, CYC needs to be somewhat carefully engineered. Unfortunately, the CYC corpus is not publicly available at this time, whereas OMCS *is* freely available and downloadable via the website (www.openmind.org/commonsense).

Even though OMCS is a more noisy and ambiguous corpus, we find that it is still suitable to our task. By

normalizing the concepts, we can filter out some possibly unusable knowledge (Section 3.2). The impact of ambiguity and noise can be minimized using heuristics (Section 4.1). Even with these precautionary efforts, some anomalous or bad knowledge will still exist, and can lead to seemingly semantically irrelevant concept expansions. In this case, we rely on the fail-soft nature of the application that uses this semantic resource to handle noise gracefully.

3. Constructing a World Semantic Resource

In this section, we describe how a usable subset of the knowledge in OMCS is extracted and structured specifically for the photo retrieval task. First, we apply sentence pattern rules to the raw OMCS corpus and extract crude predicate argument structures, where predicates represent commonsense relations and arguments represent commonsense concepts. Second, concepts are normalized using natural language techniques, and unusable sentences are discarded. Third, the predicate argument structures are read into a Concept Node Graph, where nodes represent concepts, and edges represent predicate relationships. Edges are weighted to indicate the strength of the semantic connectedness between two concept nodes.

3.1. Extracting Predicate Argument Structures

The first step in extracting predicate argument structures is to apply a fixed number of mapping rules to the sentences in OMCS. Each mapping rule captures a different commonsense relation. Commonsense relations, insofar as what interests us for constructing our world semantic resource for photos, fall under the following general categories of knowledge:

1. Classification: A dog is a pet
2. Spatial: San Francisco is part of California
3. Scene: Things often found together are: restaurant, food, waiters, tables, seats
4. Purpose: A vacation is for relaxation; Pets are for companionship
5. Causality: After the wedding ceremony comes the wedding reception.
6. Emotion: A pet makes you feel happy; Rollercoasters make you feel excited and scared.

In our extraction system, mapping rules can be found under all of these categories. To explain mapping rules, we give an example of knowledge from the aforementioned Scene category:

```
somewhere THING1 can be is PLACE1
somewherecanbe
THING1, PLACE1
0.5, 0.1
```

Mapping rules can be thought of as the grammar in a shallow sentence pattern matching parser. The first line in each mapping rule is a sentence pattern. THING1 and PLACE1 are variables that approximately bind to a word or phrase, which is later mapped to a set of canonical commonsense concepts. Line 2 specifies the name of this predicate relation. Line 3 specifies the arguments to the predicate, and corresponds to the variable names in line 1. The pair of numbers on the last line represents the confidence weights given to forward relation (left to right), and backward relation (right to left), respectively, for this predicate relation. This also corresponds to the weights associated with the directed edges between the nodes, THING1 and PLACE1 in the graph representation.

It is important to distinguish the value of the forward relation on a particular rule, as compared to a backward relation. For example, let us consider the commonsense fact, *"somewhere a bride can be is at a wedding."* Given the annotation *"bride,"* it may be very useful to return *"wedding."* However, given the annotation *"wedding,"* it seems to be less useful to return *"bride," "groom," "wedding cake," "priest,"* and all the other things found in a wedding. For our problem domain, we will generally penalize the direction in a relation that returns hyponymic concepts as opposed to hypernymic ones. The weights for the forward and backward directions were manually assigned based on a cursory examination of instances of that relation in the OMCS corpus.

Approximately 20 mapping rules are applied to all the sentences (400,000+) in the OMCS corpus. From this, a crude set of predicate argument relations are extracted. At this time, the text blob bound to each of the arguments needs to be normalized into concepts.

3.2. Normalizing Concepts

Because any arbitrary text blob can bind to a variable in a mapping rule, these blobs need to be normalized into concepts before they can be useful. There are three categories of concepts that can accommodate the vast majority of the parseable commonsense knowledge in OMCS: Noun Phrases (things, places, people), Attributes (adjectives), and Activity Phrases (e.g.: *"walk the dog," "buy groceries."*), which are verb actions that take either no argument, a direct object, or indirect object.

To normalize a text blob into a Noun Phrase, Attribute or Activity Phrase, we tag the text blob with part of speech information, and use these tags filter the blob through a miniature grammar. If the blob does not fit the grammar, it is massaged until it does or it is rejected altogether. Sentences, which contain text blobs that cannot be normalized, are discarded at this point. The final step involves normalizing the verb tenses and the number of the nouns. Only after this is done can our predicate argument structure be added to our repository.

The aforementioned noun phrase, and activity phrase grammar is shown below in a simplified view. Attributes are simply singular adjectives.

```
NOUN PHRASE:
(PREP) (DET|POSS-PRON) NOUN
(PREP) (DET|POSS-PRON) NOUN NOUN
(PREP) NOUN POSS-MARKER (ADJ) NOUN
(PREP) (DET|POSS-PRON) NOUN NOUN NOUN
(PREP) (DET|POSS-PRON) (ADJ) NOUN PREP NOUN

ACTIVITY PHRASE:
(PREP) (ADV) VERB (ADV)
(PREP) (ADV) VERB (ADV) (DET|POSS-PRON) (ADJ) NOUN
(PREP) (ADV) VERB (ADV) (DET|POSS-PRON) (ADJ) NOUN NOUN
(PREP) (ADV) VERB (ADV) PREP (DET|POSS-PRON) (ADJ) NOUN
```

The grammar is used as a filter. If the input to a grammar rule matches any optional tokens, which are in parentheses, then this is still considered a match, but the output will filter out any optional fields. For example, the phrase, *"in your playground"* will match the first rule and the phrase will stripped to just *"playground."*

3.3. Concept Node Graph

To model concept expansion as a spreading activation task, we convert the predicate argument structures gathered previously into a Concept Node Graph by mapping arguments to concept nodes, and predicate relations to edges connecting nodes. Forward and backward edge weights come from the mapping rule associated with each predicate relation. A segment of the graph is shown in Figure 1.

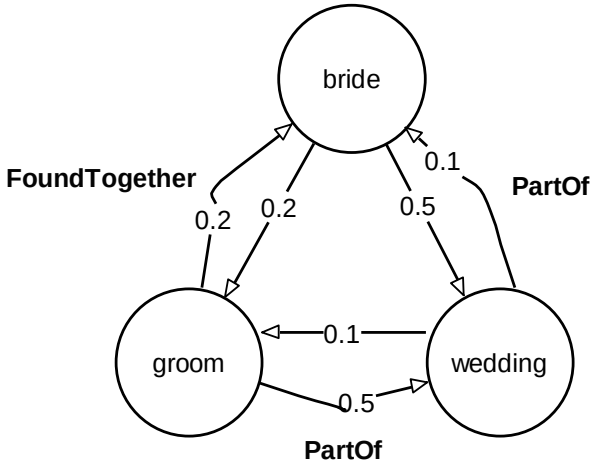


Figure 1. A portion of the Concept Node Graph. Nodes are concepts, and edges correspond to predicate relations.

The following statistics were compiled on the automatically constructed resource:

- 400,000+ sentences in OMCS corpus
- 50,000 predicate argument structures extracted
- 20 predicates in mapping rules
- 30,000 concept nodes
- 160,000 edges
- average branching factor of 5

4. Concept Expansion Using Spreading Activation

In this section, we explain how concept expansion is modeled as spreading activation. We propose two heuristics for re-weighting the graph to improve relevance. Examples of the spreading activation are then given.

In spreading activation, the origin node is the concept we wish to expand (i.e. the annotation) and it is the first node to be activated. Next, nodes one hop away from the origin node are activated, then two levels away, and so on. A node will only be activated if its activation score (AS) meets the activation threshold, which is a tolerance level between 0 (irrelevant) and 1.0 (most relevant). The origin node has a score of 1.0. Given two nodes A and B, where A has 1 edge pointing to B, the activation score of B is given in equation (1).

$$AS(B) = AS(A) * weight(edge(A,B)) \quad \textbf{(1)}$$

When no more nodes are activated, we have found all the concepts that expand the input concept up to our set threshold.

4.1. Heuristics to Improve Relevance

One problem that can arise with spreading activation is that nodes that are activated two or more hops away from the origin node may quickly lose relevance, causing the search to lose focus. One reason for this is noise. Because concept nodes do not make distinctions between different word senses (an aforementioned problem with OMCS), it is possible that a node represents many different word senses. Therefore, activating more than one hop away risks exposure to noise. Although associating weights with the edges provides some measure of relevance, these weights form a homogenous class for all edges of a common predicate (recall that the weights came from mapping rules).

We identify two opportunities to re-weight the graph to improve relevance: reinforcement and popularity. Both of these heuristics are known techniques associated with spreading activation networks (Salton and Buckley, 1988). We motivate their use here with observations about our particular corpus, OMCS.

4.1.1. Reinforcement

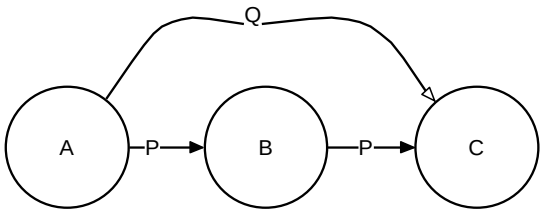


Figure 2. An example of reinforcement

As illustrated in Figure 2, we make the observation that if node C is connected to node A through both paths P and Q, then C would be more relevant to A than had either path P or Q been removed. We call this *reinforcement* and define it as two or more corroborating pieces of evidence, represented by paths, that two nodes are semantically related. The stronger the reinforcement, the higher the potential relevance.

Looking at this in another way, if three or more nodes are mutually connected, they form a cluster. Examples of clusters in our corpus are higher-level concepts like weddings, sporting events, parties, etc., that each have many inter-related concepts associated with them. Within each such cluster, any two nodes have enhanced relevance because the other nodes provide additional paths for reinforcement. Applying this, we re-weight the graph by detecting clusters and increasing the weight on edges within the cluster.

4.1.2. Popularity

The second observation we make is that if an origin node A has a path through node B, and node B has 100

children, then each of node B's children are less likely to be relevant to node A than if node B had had 10 children.

We refer to nodes with a large branching factor as being popular. It happens that popular nodes in our graph tend to either correspond to very common concepts in commonsense, or tend to have many different word senses, or word contexts. This causes its children to have in general, a lower expectation of relevance.

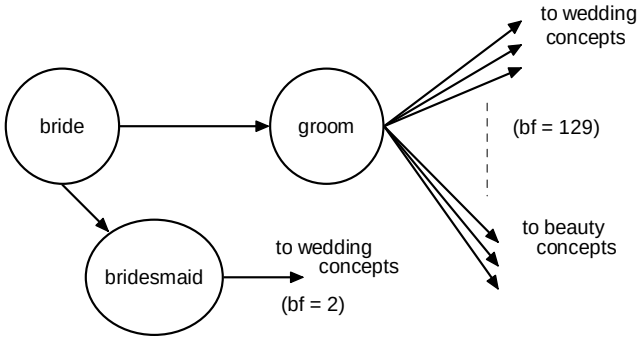


Figure 3. Illustrating the negative effects of popularity

As illustrated in Figure 3, the concept *bride* may lead to *bridesmaid* and *groom*. Whereas *bridesmaid* is a more specific concept, not appearing in many contexts, *groom* is a less specific concept. In fact, different senses and contexts of the word can mean "the groom at a wedding," or "grooming a horse" or "he is well-groomed." This causes *groom* to have a much larger branching factor.

It seems that even though our knowledge is common sense, there is more value associated with more specific concepts than general ones. To apply this principle, we visit each node and discount the weights on each of its edges based on the metric in equation (2). (α and β are constants):

$$newWeight = oldWeight * discount$$
$$discount = \frac{1}{\log(\alpha * branchingFactor + \beta)} \quad (2)$$

4.2. Examples

Below are actual runs of the concept expansion program using an activation threshold of 0.1. They were selected to illustrate what can be commonly expected from the expansions, including limitations posed by the knowledge.

```
>>> expand("bride")
('love', '0.632'), ('wedding', '0.5011')
('groom', '0.19'), ('marry', '0.1732')
('church', '0.1602'), ('marriage', '0.1602')
('flower girl', '0.131') ('happy', '0.131')
('flower', '0.131') ('lake', '0.131')
('cake decoration', '0.131') ('grass', '0.131')
('priest', '0.131') ('tender moment', '0.131')
('veil', '0.131') ('wife', '0.131')
('wedding dress', '0.131') ('sky', '0.131')
('hair', '0.1286') ('wedding bouquet', '0.1286')
('snow covered mountain', '0.1286')

>>> expand('london')
('england', '0.9618') ('ontario', '0.6108')
('europe', '0.4799') ('california', '0.3622')
('united kingdom', '0.2644') ('forest', '0.2644')
('earth', '0.1244')

>>> expand("symphony")
('concert', '0.5') ('music', '0.4')
('theatre', '0.2469')
('conductor', '0.2244')
('concert hall', '0.2244')
('xylophone', '0.1') ('harp', '0.1')
('viola', '0.1') ('cello', '0.1')
('wind instrument', '0.1') ('bassoon', '0.1')
('violin', '0.1')

>>> expand("listen to music")
('relax', '0.4816') ('be entertained', '0.4816')
('have fun', '0.4') ('relaxation', '0.4')
('happy', '0.4') ('hang', '0.4')
('hear music', '0.4') ('dorm room', '0.4')
('understand', '0.4') ('mother', '0.2')
('happy', '0.136')
('get away', '0.136') ('listen', '0.136')
('change psyche', '0.136') ('show', '0.1354')
('dance club', '0.1295') ('frisbee', '0.1295')
('scenery', '0.124') ('garden', '0.124')
('spa', '0.124') ('bean bag chair', '0.124')
```

The expansion of *"bride"* shows the diversity of relations found in the semantic resource. *"Love"* is some emotion that is implicitly linked to brides, weddings, and marriage. Expansions like *"priest", "flower girl,"* and *"groom"* are connected through social relations. *"Wife"* seems to be temporally connected. To *"marry"* indicates the function of a wedding.

However, there are also expansions whose connections are not as obvious, such as *"hair,"* and *"lake."* There are also other expansions that may be anomalies in the OMCS corpus, such as *"tender moment"* and *"snow covered mountain."* These examples point to the need for some type of statistical filtering of the knowledge in the corpus, which is not currently done.

In the last expansion example, the concept of "listen to music" is arguably more abstract than the wedding concept, and so the expansions may seem somewhat arbitrary. This illustrates one of the limitations of any common sense acquisition effort: deciding upon which topics or concepts to cover, how well they are covered, and to what granularity they are covered.

5. Conclusion

In this paper, we presented a mechanism for robust photo retrieval: using a world semantic resource to expand a photo's annotations. The resource was automatically constructed from the publicly available Open Mind Common Sense corpus. Sentence patterns were applied to the corpus, and simple predicate argument structures were extracted. After normalizing arguments into syntactically neat concepts, a weighted concept node graph was constructed. Concept expansion is modeled as spreading activation over the graph. To improve relevance in spreading activation, the graph was re-weighted using heuristics for reinforcement and popularity.

This work has not yet been formally evaluated. Any evaluation will likely take place in the context of the larger system that this mechanism is used in, called (A)nnotation and (R)etrieval (I)ntegration (A)gent (Lieberman et al., 2001) ARIA is an assistive software agent which automatically learns annotations for photos by observing how users place photos in emails and web pages. It also monitors the user as s/he types an email and finds opportunities to suggest relevant photos. The idea of using world semantics to make the retrieval process more robust comes from the observation that concepts depicted in photos are often spatially, temporally, and socially related in a commonsensical way. While the knowledge extracted from OMCS does not give very complete coverage of many different concepts, we believe that what concept expansions *are* done have added to the robustness of the retrieval process. Sometimes the concept expansions are irrelevant, but because ARIA engages in opportunistic retrieval that does not obstruct the user's task of writing the email, the user does not suffer as a result. We sometimes refer to ARIA as being "fail-soft" because good photo suggestions can help the task, but the user can ignore bad photo suggestions.

Robust photo retrieval is not the only IR task in which semantic resources extracted from OMCS have been successfully applied. (Liu et al., 2002) used OMCS to perform inference to generate effective search queries by analyzing the user's search goals. (Liu and Singh, 2002) uses the subset of causal knowledge in OMCS to generate crude story scripts.

In general, the granularity of the knowledge in OMCS can benefit any program that deals with higher-level social concepts of the everyday world. However, because of limitations associated with this corpus such as noise, ambiguity, and coverage, OMCS is likely to be only useful at a very shallow level, such as providing an associative mechanism between everyday concepts or performing first-order inference.

Future work is planned to improve the performance of the mechanism presented in this paper. One major limitation that we have encountered is noise, stemming from ambiguous word senses and contexts. To overcome this, we hope to apply known word sense disambiguation techniques to the concepts and the query, using word sense co-occurrence statistics, WordNet, or LDOCE. A similar approach could be taken to disambiguate meaning contexts, but it is less clear how to proceed.

Another point of future work is the migration from the sentence pattern parser to a broad coverage parser so that we can extract more kinds of commonsense relations from the corpus, and make more sentences "usable."

6. Acknowledgements

We thank our colleagues Push Singh, and Kim Waters at the MIT Media Lab, Tim Chklovski at the MIT AI Lab, and Erik Mueller at IBM, who are also working on the problem of commonsense, for their contributions to our collective understanding of the issues. We would especially like to thank Push for directing OMCS and for his advocacy of commonsense.

7. References

Fellbaum, C. (ed.), 1998. *WordNet: An Electronic Lexical Database*. MIT Press, Cambridge, MA.

Klink, S., 2001. Query reformulation with collaborative concept-based expansion. *Proceedings of the First International Workshop on Web Document Analysis*, Seattle, WA.

Lenat, D., 1998. *The dimensions of context-space*, Cycorp technical report, www.cyc.com.

Lieberman, H., Rosenzweig E., Singh, P., 2001. Aria: An Agent For Annotating And Retrieving Images, *IEEE Computer*, July 2001, pp. 57-61.

Lin, D., 1998. Using collocation statistics in information extraction. *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, San Francisco, CA, http://www.muc.saic.com.

Liu, H., Lieberman, H., Selker, T., 2002. GOOSE: A Goal-Oriented Search Engine With Commonsense. *Proceedings of the 2002 International Conference on Adaptive Hypermedia and Adaptive Web Based Systems*, Malaga, Spain.

Liu, H., Singh, P., 2002. MAKEBELIEVE: Using Commonsense to Generate Stories. *Proceedings of the 20th National Conference on Artificial Intelligence (AAAI-02)* -- Student Abstract. Seattle, WA.

Open Mind Common Sense Website and Corpus. Available at: www.openmind.org/commonsense.

Peat, H.J. and Willett, P., 1991. The limitations of term co-occurrence data for query expansion in document retrieval systems. *Journal of the ASIS*, 42(5), 378-383.

Richardson, S.D., Dolan, W.B., Vanderwende, L., 1998. MindNet: Acquiring and Structuring Semantic Information from Text. *Proceedings of the joint ACL and COLING conference*, 1098-1102, Montreal.

Salton, G., and Buckley, C., 1988. On the Use of Spreading Activation Methods in Automatic Information Retrieval. *Proceedings of the 11th Ann. Int. ACM SIGIR Conf. on R&D in Information Retrieval (ACM)*, 147-160.

Singh, P., 2002. The public acquisition of commonsense knowledge. *Proceedings of AAAI Spring Symposium: Acquiring (and Using) Linguistic (and World) Knowledge for Information Access.* Palo Alto, CA, AAAI.

Voorhees, E., 1994. Query expansion using lexical-semantic relations. *Proceedings of ACM SIGIR Intl. Conf. on Research and Development in Information Retrieval*, pages 61—69.

Xu, J., and Croft, W.B., 1996. Query Expansion Using Local and Global Document Analysis. Proceedings of the Nineteenth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 4—11.

Managing Synonymy and Polysemy in a Document Retrieval System Using WordNet

Claude de Loupy[*]
Marc El-Bèze[**]

[*] Sinequa
51-54, rue Ledru-Rollin
94200 Ivry-sur-Seine
France
loupy@sinequa.com
http://www.sinequa.com

[**] Laboratoire Informatique d'Avignon
B.P. 1228 Agroparc
339 Chemin des Meinajaries
84911 Avignon Cedex 9
France
marc.elbeze@lia.univ-avignon.fr
http://www.lia.univ-avignon.fr/

Abstract

This paper reports several experiments of document retrieval with TREC-6 using semantic knowledge. In a first set of experiments, synonyms and hyponyms given by WordNet are used in order to enrich queries. A small improvement is shown. The second set uses a word sense disambiguation system in order to cope with polysemy. There is almost no modification of performances but this is an important result considering Sanderson's results. Our system performs at 72% of accuracy when Sanderson concludes a system performing at less than 90% degrades results. When using both query enrichment and WSD, the improvements are a little better, especially for the first document retrieved. Lastly, a small set of experiments using specialized thesauri is presented, showing important improvements.

Keywords

Document Retrieval, Word Sense Disambiguation, Synonymy, Polysemy, WordNet, HMM

1 Introduction

From the beginning of automatic Document Retrieval (DR), researchers have tried to use thesaurus. But results were often disappointing: Salton (1968) used the Harris Synonym Thesaurus and noted a fall of performances. both Harman (1988) and Voorhees (1993, 1994) using WordNet, came to the same conclusion, even if Harman noted that when the user is involved in the process, results are improved.

In this paper, we report several experiments using TREC-6 (Harman, 1997) for evaluation, WordNet (Miller *et al.*, 1990) as a semantic lexicon and a Word Sense Disambiguation (WSD) system trained on SemCor (Miller *et al.*, 1993). The results of these experiments contradict some widespread ideas and some conclusions of other experiments.

The DR system used is described in section 2. In section 3 several experiments using query enrichment with synonyms or hyponyms from WordNet are analyzed. In section 4, the impact of WSD in DR is shown. Section 5 reports experiments using both information and section 6 reports the use of specialized thesauri.

2 The Document Retrieval system used

The DR system used for these experiments is *IndeXal* (Loupy *et al.*, 1998a). The similarity measure is the one proposed by Harman (1986) with a slight modification:

$$(1) \quad score(d) = \frac{\sum_{x \in d \cap q} TF_d(x) \cdot IDF(x)}{\sum_{x \in q} TF_d(x) \cdot IDF(x)}$$

where $score(d)$ is the score of document d according to the query, and:

$$(2) \quad IDF(x) = -\log\left(\frac{n(x)}{N}\right)$$

$$(3) \quad TF_d(x) = K + (1-K) \cdot \frac{\log(O_d(x))}{\log(L_d)}$$

with: $n(x)$ the number of documents containing x, N the total number of documents, $O_d(x)$ the number of occurrences of x in d, L_d the length of document d, and K a coefficient (here is the modification). This coefficient is used to determine the relative importance of *IDF* and *TF*. Best scores are obtained with $K = 0.3$. In this paper, the results are evaluated on TREC-6 (Voorhees & Harman, 1997). Only *Titles* were used (that is 1 to 4 words queries). The results of table 1 will serve as reference for comparison with the other results. *stem* represents the results obtained with a classic stemming procedure[1] and *lem* the ones obtained with a POS tagging system called

[1] We used Porter's stemmer (Porter, 1980)

ECSta (Spriet & El-Bèze, 1997) and a lemmatization. The performances for French are good (96.5% of efficiency). We trained the tagger on the SemCor which is a very small corpus. The final performances are only 88.8% of correct assignation. This seems very weak, but considering only the tagging of content words (nouns, verbs, adjectives and adverbs), the error rate is only 3.9%. This seems sufficient for the following experiments.

	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6

Table 1: Basic results

Here, we chose to evaluate the different strategies using only the following statistics given in TREC:

- the number of relevant document retrieved (*Rel-Ret*)
- the average precision (*Av-Prec*)
- the precision at 5, 10, 20 and 100 documents retrieved
- the R-precision (*R-Prec*) that is the precision when there are as many documents returned by the system than relevant ones.

Though stemming seems to be the most efficient strategy, we can see that the precision of the first retrieved documents increases with lemmatization. We think that the precision of the first retrieved documents is the most important for evaluation because they are the documents a user will read and they can be used in an automatic relevance feedback procedure. So, lemmatization does not seem to be a bad strategy. But it would be interesting to improve performances concerning the other statistics, particularly for the precision when 20 documents are retrieved.
In the following experiments, enrichment and disambiguation procedures are used after lemmatization.

3 Using WordNet to Enrich Queries

Smeaton *et al.* (1995) showed relevant documents do not necessarily contain words of the query. One way to improve DR systems performances is to enrich the queries with synonyms or hyponyms.

3.1 Why are synonyms important?

Figure 1 shows the sets of documents containing "*woman*" or "*parliament*" or both terms or none of them and their intersection with the set of relevant documents for query *321* ("*woman in parliament*"). We can see that 10% of relevant documents do not contain the terms "*woman*" and "*parliament*".
It is legitimate to expect that the query enrichment should help the DR systems to retrieve these 10%. Using query enrichment with synonyms and hyponyms, Smeaton *et al.* (1995) retrieved 5% of relevant documents of TREC-3 (Harman, 1994) that do not contain any word of the queries. The following sections show experiments on TREC-6 using query enrichment with synonyms and hyponyms from WordNet 1.5 (Miller *et al.*, 1993).

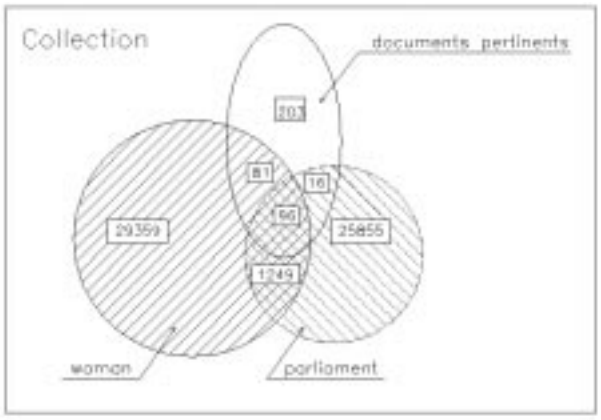


Figure 1: Distribution of documents for request 321

3.2 Presentation of the method

3.2.1 Similarity with enrichment

Enrichment is made at the word level. If a word x of the query has 2 synonyms (y and z), x is replaced by $X = (x, \alpha \cdot y, \alpha \cdot z)$ where $\alpha \in [0,1]$ indicates the importance given to the synonyms compared with the original word. So, we create a pseudo-word X with

(4) $$n(X) = \sum_d C(x,d)$$

with $C(x,d)=1$ if the document d contains x and $C(x,d)=\alpha$ if d does not contain x but contains y or z.
and

(5) $$O_d(X) = O_d(x) + \alpha \cdot O_d(y) + \alpha \cdot O_d(z)$$

It is very important to note that synonyms is taken into account for the calculation of *IDF* and *TF*. Usually, in query enrichment systems, words are added to the query as if they were independent. So each added word has its own *IDF* and *TF*.

3.3 Using Synonyms

In order to enrich queries, we used WordNet 1.5 synsets. 91 591 synsets are given in WordNet 1.5.

3.3.1 A single sense

In this first experiment, only monosemic words are expanded and the expansion concerns only monosemic synonyms. Therefore, polysemy has no influence on the results. The following table gives the results obtained according to the weight α ($\alpha = 0$ corresponds to the lem-basic results and $\alpha = 1$ means that synonyms are as important as original words).
Firstly, we can observe that modifications of the results are very small. But this is a very important observation. It is usually said that the use of synonyms decrease precision and here we can see that it is not the case.
Actually, only 22 queries are concerned by this enrichment. Compared with lemmatization, the performances are increased for 10 of them and decreased for the others (if we consider the average precision). If we

take $\alpha = 0.5$, the average precision is slightly increased (0.3) compared with *Av-prec* but this is not significant. The important fact is that all the decreases in average precision are lower that 1% (absolute values) when the query *317* ("*Unsolicited Faxes*") shows a 7.1 gain if it is enriched by "*unsought*" and "*facsimile*". The other increases are smaller than 4%.

α	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
0.1	1985	21.2	39.6	36.2	30.5	16.8	25.7
0.3	1988	21.3	39.6	36.2	30.5	16.8	25.8
0.5	1988	21.3	40.4	36.6	30.6	16.8	25.8
0.7	1986	21.4	40.4	36.8	30.6	16.8	25.8
0.9	1984	21.3	40.0	36.6	30.9	16.8	25.5
1	1981	21.0	39.2	36.2	30.9	16.8	25.5

Table 2: Enrichment of monosemic words with monosemic synonyms

Concerning query *302*, it is important to note that the gain (+2.8%) is not strictly due to synonymy enrichment. The query is: "*poliomyelitis post polio*". The terms *polio* and *poliomyelitis* are synonyms in WordNet. So, after enrichment, the query is: "(*poliomyelitis* OR *polio*) *post* (*polio* OR *poliomyelitis*)". There is no addition of words, but the calculation of scores is modified. This suggests the system should benefit of a modification of the similarity measure presented in section 2 (formulae 1, 2 and 3).

3.3.2 Several senses

In this section, we want to take into account the number of senses of original words and synonyms in order to see if it is interesting to enrich polysemic words. Table 3 gives the results of an enrichment according to the maximum of senses (*n*) an enriched word and its synonyms have.

n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
1	1988	21.3	40.4	36.6	30.6	16.8	25.8
2	1996	21.4	40.8	36.6	30.7	17.0	26.0
3	1991	21.3	41.2	36.2	30.5	16.9	25.9
4	1967	21.3	40.8	36.0	30.1	16.8	25.7
5	1961	21.4	40.8	36.4	30.3	16.7	26.0
6	1964	21.4	40.8	36.4	30.5	16.7	26.0
7	1957	21.2	40.4	35.6	30.4	16.6	25.8
8	1960	21.2	40.4	35.2	30.5	16.5	25.7
9	1959	21.1	40.0	34.6	30.2	16.3	25.6
∞	1959	21.1	40.0	34.4	30.2	16.3	25.6

Table 3: Enrichment of polysemic words with polysemic synonyms

Here again, there are almost no differences between the basic lemmatization results and the one obtained after enrichment. Nevertheless, it is important to note that there is no decrease of performances when the words which have 3 or less senses are enriched with words which have 3 or less senses. This result will be used in the section 4.

3.4 Using Hyponyms

Another way to enrich queries is to use hyponyms instead of synonyms. The following table gives the results of such an enrichment according to the maximum number of senses (*n*) a word must have to be enriched by its hyponyms (if they also have less than *n* senses).

n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
1	1999	21.4	39.6	36.2	30.9	16.8	25.7
2	2015	21.3	40.4	36.4	30.9	17.1	25.8
3	2026	21.4	41.6	36.2	31.5	17.0	25.9
4	2004	21.3	41.2	36.2	31.2	16.8	25.7
5	2003	21.3	41.2	36.4	31.7	16.8	25.9
syn	1991	21.3	41.2	36.2	30.5	16.9	25.9

Table 4: Enrichment of polysemic words with polysemic hyponyms

The differences are more important here. 32 queries are modified by this enrichment. For 20 queries, performances are increased (up to 9.2 % in absolute value) and for 12 of them performances decrease (down to –6.5 %). In fact, performances are better when using hyponyms instead of synonyms.
It is important to note that the performances for the first 20 documents are approximately the same using hyponyms or stemming.

4 WSD and DR

Polysemy is a very important problem in DR. In this section, we start by a reminder of some important previous experiments. Then, we shall present our own experiments.

4.1 Important previous works[2]

The most cited work concerning the use of WSD for DR is (Sanderson, 1994). Sanderson's conclusion is that a WSD system performing with less than 90% of accuracy decreases results of DR. This is really a problem because the two Senseval evaluations (Kilgarriff & Palmer, 2000) show that the performances of such systems is less than 80%.
This work has been criticized by Schütze and Pedersen (1995) because the use of pseudo-words (Yarowsky, 1993) by Sanderson does not fit the real behavior of polysemic words. They even showed an improvement of performances using WSD on TREC evaluation. But their system is based on automatic construction of a thesaurus.
Gonzalo *et al.* (1998b) used the SemCor (Miller *et al.*, 1993) in order to build an evaluation framework where the importance of WSD and synonymy can be easily evaluated. They report a great improvement of performances. This is encouraging but not really a proof. The evaluation corpus is very special: queries were built manually as abstracts of the SemCor documents and they

[2] A more precise description of previous works can be found in Sanderson (2000).

consider there is only one relevant document for a "query".
They also evaluated the influence of disambiguation errors, confirming the results of Sanderson: 10% of wrong disambiguation leads to a decrease in DR results. But, using both WSD and synonymy enrichment, the tolerance of errors is very much higher: with a WSD system performing at 70%, performances are increased and even with 40% of good identification, performances are stable. These results are a bit strange but quiet encouraging for further experiments.
In a further paper, Gonzalo *et al.* (1999) reproduced the Sanderson's experiments using pseudo-words and found a threshold of 75% instead of the 90% expected. This result is more in agreement with the ones of this paper.
The next sections present the use of a complete WSD system in a TREC experiment. We show that, even if performances are not increased, a quite basic system performing between 71.5% and 74.6% of accuracy does not degrade results.

4.2 Presentation of the Method

In section 3.2.2, we saw that enriching original words with synonyms even when they have three senses could be interesting. In this section, we use a WSD system in order to choose the most probable one, two or three senses for words according to their contexts.
The WSD system (Loupy *et al.*, 1998b) is based on HMM. A Baum-Welch algorithm (Baum *et al.*, 1970) is used in order to keep several senses. This is important for document retrieval in view of the following facts:

- the WSD system can do mistakes (see performances below)
- even if the sense of a word is obvious, the other senses are often kept in mind
- since WordNet senses are very fine grained (41 senses for the verb "*run*"), keeping several senses can be useful in order to represent a coarser sense which do not exist.
- it is sometimes impossible to disambiguate a polysemic word (Kilgarriff, 1994) even for a human being.

The HMM model were trained on the SemCor and its performances were evaluated using 95% for training and 5% for tests. The following table gives the scores when 1, 2 or 3 senses are kept, considering all words (*all*) or only ambiguous ones (*amb*). Moreover, two results are given for each case. The first one corresponds to an evaluation when part-of-speech is known (real evaluation of WSD) and the second one when this POS is not known (real world). The model is a bigram one. With unisem, the performances are slightly inferior (of about 0.4).

		1 sense		2 senses		3 senses	
		all	*amb*	*all*	*amb*	*all*	*amb*
POS	**known**	74.6%	62.5%	87.7%	78.0%	92.9%	83.2%
	unk.	71.5%	59.7%	84.5%	74.9%	89.8%	79.6%

Table 5: Performances of the WSD system

So, the performances are really lower than the one given by Sanderson when he said that a WSD system must perform at 90% or more.

4.3 A simple use of disambiguation

If we keep 3 senses during disambiguation, there are many ways to use it. Figure 2 shows the combinations between a disambiguated query and a disambiguated document.

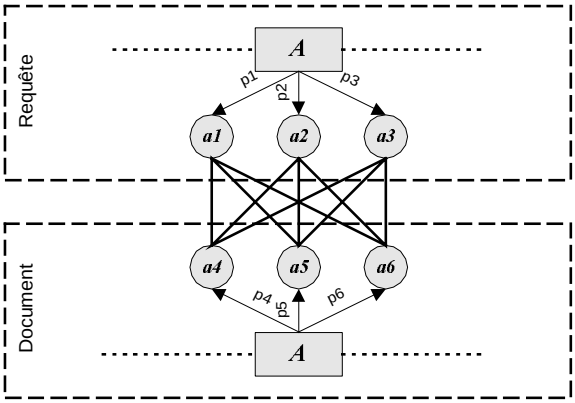


Figure 2 : Combinations between query and document when using disambiguation

Several combinations were tested. Table 6 gives the results. The first line (*all*) gives the results when no disambiguation is made. The other lines (*m-n*) represent a disambiguation where *m* senses are kept in the query and *n* senses are kept in the documents.

m-n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
1-1	1976	20.9	41.2	36.0	30.3	16.8	25.1
1-2	1978	21.1	41.2	35.6	30.6	16.7	25.4
1-3	1979	21.1	40.4	35.8	30.6	16.8	25.5
2-1	1979	21.1	40.8	36.2	30.5	16.8	25.4
2-2	1974	21.2	41.2	36.2	31.2	16.8	25.6
2-3	1969	21.2	40.8	36.2	31.2	16.7	25.6
3-1	1983	21.1	40.4	36.0	30.7	16.8	25.6
3-2	1969	21.2	40.8	36.6	31.4	16.8	25.6
3-3	1971	21.2	40.8	36.4	31.4	16.7	25.6

Table 6: Results of a simple use of WSD in a DR system

Performances are almost the same with WSD (whatever the strategy is) and without. But, here again, there is a very tiny improvement for the first documents retrieved.
If we consider only Average Precision for the *1-1* strategy, results are improved for 24 queries and decreased for only 10 queries. But, while no query is improved by more than 1%, the query *339* ("*Alzheimer's drug treatment*") decreases by 9.5%. The fall of precision of the other queries is less than 1.3%.
So, even if we consider that the problem of the query *339* is an "accident", improvements are very poor. But, we can also conclude that a WSD system performing at 72% does not decrease results of a DR system contrary to what Sanderson claims.

Another interesting point is that there is almost no modification of recall.

4.4 Using sense probability from WSD system

Previous experiments were made without taking into account the probabilistic information (probability of each of the three senses) given by the WSD system. It should be interesting to use them. The similarity measure is the same as the one given in section 2 but the way the number of occurrences is counted is modified:

(6) $$n(x)=S_q(x)\cdot\sum_d Max\left(S_d(x,i)\right)$$

where $S_d(x,i)$ is the probability given by the WSD system to the word-sense x at the position i and $S_Q(x)$ the probability of the word-sense x in the query.

(7) $$O_d(x)=S_Q(x)\cdot\sum_i S_d(x,i)$$

Table 7 gives the results of such a heuristic.

	m-n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
	stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
	all	1984	21.1	39.6	36.0	30.7	16.8	25.6
sense	**1-1**	1976	20.9	41.2	36.0	30.3	16.8	25.1
	2-2	1974	21.2	41.2	36.2	31.2	16.8	25.6
	3-3	1971	21.2	40.8	36.4	31.4	16.7	25.6
req	**1-1**	1913	20.0	39.2	33.6	29.4	16.2	23.6
	2-2	1921	20.3	39.6	34.8	29.9	16.2	24.0
	3-3	1929	20.3	39.6	35.0	29.8	16.2	24.0
doc	**1-1**	1859	18.6	37.2	31.6	27.5	15.1	22.3
	2-2	1881	18.7	36.0	31.8	28.5	15.2	23.2
	3-3	1884	18.8	37.2	32.4	28.5	15.1	23.3
req+doc	**1-1**	1774	17.8	36.4	31.4	26.4	14.4	21.1
	2-2	1777	17.9	36.4	31.8	26.5	14.5	21.2
	3-3	1780	17.9	36.4	31.6	26.6	14.5	21.2

Table 7: Results of using WSD in a DR system taking probabilities into account

The lines *sens* give the results reported in section 4.3 (probabilities are not involved in scores). The lines *req* report the use of WSD probabilities for queries only, *doc* for documents only and *req+doc* for both queries and documents.

We can see that the results have decreased. This is very surprising. Another heuristic may help us to overcome this problem.

5 Using Both WSD and Query Enrichment

In the previous sections, we use query enrichment and WSD in separate experiments. In this section, we shall combine both strategies. The following tables show the performances obtained when one, two or three senses are kept after WSD.

m-n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
syn 1-1	1999	21.4	39.6	36.2	30.9	16.8	25.7
wsd 1-1	1976	20.9	41.2	36.0	30.3	16.8	25.1
syn+wsd	1971	21.1	42.4	36.2	30.2	16.8	25.6

Table 8: combining enrichment and WSD with one sense

m-n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
syn 2-2	2015	21.3	40.4	36.4	30.9	17.1	25.8
wsd 2-2	1974	21.2	41.2	36.2	31.2	16.8	25.6
syn+wsd	1968	21.4	42.4	36.8	31.4	16.8	26.0

Table 9: combining enrichment and WSD with two senses

m-n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
syn 3-3	2026	21.4	41.6	36.2	31.5	17.0	25.9
wsd 3-3	1971	21.2	40.8	36.4	31.4	16.7	25.6
syn+wsd	1968	21.3	42.0	36.8	31.4	16.8	25.7

Table 10: combining enrichment and WSD with three senses

The results show little improvements when keeping 2 or 3 senses and enriching with WordNet synonyms. Of course, the question is: is the gain interesting compared to the cost?

6 Combining synonyms and stemming

As the use of synonyms does not show any improvement, another possibility is to use both information. The following table gives the results of this strategy.

n	Rel-Ret	Av-Prec	5	10	20	100	R-Prec
stem	2068	22.3	38.2	34.8	31.9	17.2	26.1
lem	1984	21.1	39.6	36.0	30.7	16.8	25.6
syn 3-3	2026	21.4	41.6	36.2	31.5	17.0	25.9
lem-stem	2124	23.1	39.2	36.0	31.7	17.9	26.9
lem-stem-syn	2140	22.7	40.8	36.0	32.8	17.3	26.1

Table 11: Results of using both stemming and synonymy enrichment

In this table, we can see that the use of both lemmatization and stemming is more interesting than using one of these strategy alone. The strategy using all these information (*lem-stem-syn*) gives better results than *stem* and *lem* for all statistics. It seems to be an interesting strategy although the precision for 5 documents is lower than the use of synonyms.

Other experiments should be done to evaluate the performances of hyponyms used with stems and synonyms.

7 Using expert knowledge

It is clear for all that the use of expert knowledge should improve performances of DR systems (Anand *et al.*, 1995). For this experiment, a specialized lexicon has been built for each of the ten first queries of TREC-6. The time necessary for this construction is more or less 5 minutes per thesaurus. The built lexicons are, therefore, very small. It is clear that, we never looked at relevant documents to search for relevant terms. Words linked to the words of a query were added to this query. The following list gives the words used for each of the ten queries:

301: international organized (crime drug prostitution cocaine ectasy extasy heroin trafficking traffic terrorism terrosrist criminal mafia maffia triad tong cartel)

302: (poliomyelitis polio brunhilde lansing léon paralysis) post (polio poliomyelitis brunhilde lansing léon paralysis)

303: hubble (telescope space_telescope infrared_telescope optical_mirror space black_hole invisible_space big_bang) (achievement accomplishment)

304: endangered (specie coinage mintage) (mammal panda whale)

305: most (dangerous unsafe grave graver gravest grievous) (vehicle car bus highway road)

306: (african africa angola angolan luanda namibia namibian windhoek bostwana gaborome swaziland mbabame lesotho maseroni south_africa cape_town zimbabwe zimbabwean harare zambia zambian luzaka tanzania tanzanian dar_es_salamm burundi burundian bujumbura uganda ugandan kamdala rwanda rwandan kinshasa congo congolese brazzaville gabon gabonese libreville cameroon cameroonian yaoundé nigeria nigerian abuja chad chadian djamena ndjamena sudani sudanese khartoum ethiopia ethiopian addis_abeba eritrea eritrean asimara somalia somalian mogadishu egypt egyptian cairo libya libyan tripoli tunisia tunisian tunis algeria algerian algiers morocco moroccan rabat mauritania mauritanian nouskshott senegal senegalese dakar mali bamako sierra_leone freetown madagascar madagascana madagascan antananarivo) civilian (death kill war killed killing)

307: (new newer newest) hydroelectric (project undertaking task task projection)

308: (implant implantation) (dentistry dentist tooth)

309: rap music ((crime drug prostitution cocaine ectasy extasy heroin trafficking traffic terrorism terrosrist criminal mafia maffia triad tong cartel)

310: (radio phone) wave brain cancer

We can see, for example, *dentistry* is associated with *dentist* and *tooth* and *vehicle* with *car*, *bus*, *highway* and *road*.

Table 11 gives the results obtained.

4 values are studied: number of relevant documents retrieved (*Rel-Ret*), precision for 20 document retrieved (*20*), average precision and R-precision. They are compared in 3 experiments: the basic one (*bas* - see section 2), query enrichment by WordNet synonyms (*syn* - see section 3.3.2) and query enrichment using expert knowledge (*use* - synonyms, hyponyms, see also links). The last figure represent the gain using specialized thesaurus (*use-bas*).

We can see that, in almost all cases, a specialized thesaurus increases performances. For query *306*, the gain is only due to a very simple geographic thesaurus.

		301	**302**	**303**	**304**	**305**	**306**	**307**	**308**	**309**	**310**
Rel-Ret	lem	88	64	10	97	5	124	155	3	1	6
	syn	88	64	10	97	5	123	151	3	1	6
	use	108	65	10	103	3	165	150	4	1	6
	use-lem	+20	+1	=	+6	-2	+41	-5	+1	=	=
20	lem	45.0	75.0	10.0	35.0	0.0	35.0	50.0	15.0	0.0	10.0
	syn	45.0	70.0	15.0	35.0	0.0	65.0	50.0	15.0	0.0	10.0
	use	55.5	75.0	10.0	40.0	5.0	75.0	45.0	20.0	0.0	10.0
	use-lem	+10	=	=	+5	+5	+10	-5	+5	=	=
Av. Prec.	lem	5.8	62.9	19.6	10.8	0.2	13.4	26.2	58.3	0.2	7.9
	syn	5.7	65.2	19.5	11.0	0.3	13.3	25.4	58.3	0.2	7.9
	user	9.5	65.6	22.2	16.0	0.4	24.7	24.9	75.4	0.4	7.9
	use-lem	+3.7	+2.7	+2.6	+5.2	+0.2	+11.3	-1.3	+17.1	+0.2	=
R-Prec.	lem	15.2	60.0	10.0	26.5	0.0	21.7	38.1	50.0	0.0	15.4
	syn	15.4	63.1	10.0	26.5	0.0	22.9	37.1	50.0	0.0	15.4
	use	19.9	63.1	10.0	31.6	5.7	41.0	37.1	75.0	0.0	15.4
	use-lem	+4.7	+3.1	=	+5.1	+5.7	+19.3	-1.0	+25.0	=	=

Table 11: Using expert knowledge for TREC queries

8 Conclusion

The experiments reported in this paper were only made on TREC-6. In order to confirm the results, they should be applied on other evaluation frameworks. Moreover, it would be interesting to use different heuristics, specially in section 4.4. But these results already lead to several conclusions:

- Using synonymy enrichment not necessarily decreases precision.
- Using WSD not necessarily decreases recall.
- A WSD system performing at 72% of accuracy does not necessarily degrades results, contrary to Sanderson's conclusions.
- The contribution of synonymy enrichment and WSD can be very poor compared to the amount of work necessary to build the necessary resources and tools.
- The combination of resources gives the best results.
- The use of specialized resources can be very useful in order to improve performances.

Of course, it seems that the "cost" is too important regarding the small improvement. In fact, the problem may come from the knowledge source, that is WordNet. It has been often criticized for DR applications for the following reasons:

- Semantic links are only possible in the same part of speech (for instance, there is no link between "*to cook*" and "*cooking*") (Gonzalo *et al.*, 1998a).
- There is no link between words of the same domain. Fellbaum *et al.* (1996) point out that the words *tennis*, *racket*, *ball* and *tennis player* have no relation.

- Senses are too fine grained (Palmer, 1998).

Another problem is that some senses are ignored. Shütze and Pedersen (1995) noticed the sense *horse race* is ignored for the word *derby* which is only tagged as a *hat*. According to them, this is an argument to use specialized automatically built resources instead of a general manually built one. An alternative solution should be find at the intersection of the two worlds: using lexical resources to have a basic knowledge and learn some relations from corpus while indexing.
One very important fact is that it is almost every time beneficial to involve users in the whole process. The next step of information retrieval will be to interact with the user. And one of the most interesting way to do that is to use lexical resources (automatically built or not) and systems performing WSD in order to help the user and to save him time. Particularly, it should be interesting to manually disambiguate queries.

9 References

Anand S. S., Bell D. A., Hughes J. G. (1995). The role of domain knowledge in data mining. in *Proceedings of International Conference on Information and Knowledge Management* (CIKM'95), pp. 37-43.

Baum L.E., Petrie T., Soules G., Weiss N. (1970). *A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains*. AMS Vol. 41, No. 1, pp. 167-171.

Fellbaum C., Grabowski J., Landes S. , Baumann A. (1996). Matching words to senses in WordNet: naive vs expert differentiation of senses. *WordNet: An electronic lexical database and some of its applications*, (editor C. Fellbaum), MIT Press, Cambridge, USA.

Gonzalo J., Verdejo F., Peters C. , Calzolari N. (1998a). Applying EuroWordNet to Cross-Language Text Retrieval. *Computers and the humanities*, Special Issue on EuroWordNet.

Gonzalo J., Verdejo F., Chugur I., Cigarran J. (1998b). Indexing with WordNet synsets can improve text retrieval. in *Proceedings of the Workshop on Usage of WordNet for Natural Language Processing*.

Gonzalo, J., A. Peñas and F. Verdejo, 1999, Lexical Ambiguity and Information Retrieval Revisited. in *Proceedings of the 1999 Joint SIGDAT Conference on EMNLP and VLC*, pp 195-202.

Harman D. (1986). An experimental study of factors important in document ranking. in *Proceedings of ACM Conference on Research and Development in Information Retrieval*. Pisa, Italy.

Harman D. (1998). Towards interactive query expansion. in *Proceedings of the 11th Annual International ACMSIGIR Conference on Research & Development in Information Retrieval*, pp. 321-331. Grenoble, France.

Harman D. (1994). Overview of the Third Text REtrieval Conference (TREC-3). *NIST Special Publication 500-226*, p 1.

Kilgarriff A., Palmer M. (Editors) (2000). Special Issue on SENSEVAL. *Computers and the Humanities*.

Loupy C. de, Bellot P., El-Bèze M. , Marteau P.F. (1998a). *Query expansion and classification of retrieved documents*. Seventh Text Retrieval Conference (TREC-7), pp. 443-450. Gaithersburg, Maryland, USA.

Loupy C. de, El-Bèze M., Marteau P.-F. (1998b). *Word Sense Disambiguation using HMM Tagger*. First International Conference on Language Resources & Evaluation, pp. 1255-1258. Granada, Spain.

Miller G.A., Beckwith R., Fellbaum C., Gross D., Miller K. (1990) *Introduction to WordNet: An online lexical database*. International Journal of Lexicography 3 (4), pp. 235-244.

Miller G. A., Leacock C., Randee T., Bunker R. (1993) *A semantic concordance*. 3rd DARPA Workshop on Human Language Technology, pp. 303-308. Plainsboro, New Jersey, USA.

Palmer M. (1998). Are WordNet sense distinctions appropriate for computational lexicons? in *Proceedings of SENSEVAL Workshop*. Herstmonceux Castle, England.

Porter M.F. (1980). An algorithm for suffix stripping. in *Program* 14 (3), pp. 130-137.

Salton G. (1968). *Automatic information organization and retrieval*. McGraw-Hill Book Company.

Sanderson M. (1994). Word sense disambiguation and information retrieval in *Proceedings of the 17th annual international ACM-SIGIR conference on Research and development in information retrieval*, pp. 142-151.

Sanderson (2000). Retrieving with good sense. in *Information Retrieval* Vol. 2 No. 1, pp. 49-69.

Schütze H., Pedersen J. (1995). Information retrieval based on word senses. in *Proceedings of the 4th Annual Symposium on Document Analysis and Information Retrieval*, pp. 161-175.

Smeaton A. F., Kelledy F., O'Donnel R. (1995). *TREC-4 experiments at Dublin City University: Thresholding posting lists, query expansion with WordNet and POS tagging of Spanish*. TREC-4, pp. 373-390.

Spriet T., El-Bèze M. (1997). *Introduction of rules into a stochastic approach for language modeling*. Computational Models for Speech Pattern Processing, NATO ASI Series F, editor K.M. Ponting.

Voorhees E. M. (1993). Using WordNet to disambiguate word senses for text retrieval. in *Proceedings of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 171-180. Pittsburg, USA.

Voorhees E. M. (1994). Query expansion using lexical-semantic relations. in *Proceedings of the 17th annual international ACM-SIGIR Conference on Research and Development in Information Retrieval* (SIGIR'94), pp. 61-69.

Voorhees E., Harman D. (1997). *Overview of the sixth Text Retrieval Conference*. Sixth Text REtrieval Conference, NIST Special Publication 500-240, pp. 1-24. Gaithersburg, MD, USA.

Yarowsky D. (1993). One sense per collocation. in *Proceedings of the ARPA Human Language Technology Workshop*, pp. 266-271, San Francisco, USA.

The Semantic Wildcard

Rada F. MIHALCEA

University of Texas at Dallas
Richardson, Texas, 75083-0688
rada@utdallas.edu

Abstract

The IRSLO (Information Retrieval using Semantic and Lexical Operators) project aims at integrating semantic and lexical information into the retrieval process, in order to overcome some of the impediments currently encountered with today's information retrieval systems. This paper introduces the semantic wildcard, one of the most powerful operators implemented in IRSLO, which allows for searches along general-specific lines. The semantic wildcard, denoted with #, acts in a manner similar with the lexical wildcard, but at semantic levels, enabling the retrieval of subsumed concepts. For instance, a search for *animal#* will match any concept that is of type *animal*, including *dog*, *goat* and so forth, thereby going beyond the explicit knowledge stated in texts. This operator, together with a lexical locality operator that enables the retrieval of paragraphs rather than entire documents, have been both implemented in the IRSLO system and tested on requests of information run against an index of 130,000 documents. Significant improvement was observed over classic keyword-based retrieval systems in terms of precision, recall and success rate.

1. Introduction

As the amount of information continues to increase, there must be new ways to retrieve and deliver information. Information is of no use if it cannot be located and the key to information location is a retrieval system. Traditionally, information retrieval systems use keywords for indexing and retrieving documents. These systems end up retrieving a lot of irrelevant information along with some useful information that the query/question was intended to elicit. Moreover, implicit knowledge makes often the bridge between a question and a document, and classic retrieval systems do not have the capability of going beyond explicit knowledge embedded in texts, thereby missing the answers to such queries.

To overcome some of the impediments currently encountered with today's information retrieval systems, we have started the IRSLO (Information Retrieval using Semantic and Lexical Operators) project that aims at integrating semantic and lexical information into the retrieval process, to the end of obtaining improved precision and recall. This paper introduces the *semantic wildcard*, one of the most powerful operators implemented in IRSLO.

Users' information needs are most of the times expressed along general-specific lines, and this paper provides analytical support towards this fact. *What sport*, *What animal*, *What body part*, are all examples of question types that require implicit knowledge about what constitutes a *sport*, *animal*, or *body-part*. The *semantic wildcard*, denoted with #, is designed to retrieve subsumed concepts. For instance, a search for *animal#* will match any concept that is of type *animal*, thereby going beyond the explicit knowledge stated in texts.

The *semantic wildcard*, together with a lexical locality operator previously introduced that enables the retrieval of paragraphs rather than entire documents (Mihalcea, 1999), were implemented in the IRSLO system and tested on requests of information run against an index of 130,000 documents. Significant improvement was observed over classic retrieval systems, in terms of precision, recall and success rate.

The paper is organized as follows. First, we present an analysis of questions asked by real time users, bringing evidence towards the fact that information need is most of the times expressed along general-specific lines. Next, we show how a novel encoding scheme - referred to as *DD-encoding* - can be applied to WordNet, in order to exploit the general-specific relations encoded in this semantic net. We then present the architecture of IRSLO, with emphasis on the *semantic wildcard* operator and the *paragraph operator*, together with experiments, results and walk through examples.

2. Defining Information Need

In order to define users' information need and assess the role that may be played by semantics in an information retrieval environment, we have performed a qualitative and quantitative analysis of information requests expressed by users in the form of natural language questions. Two sets of data are used during the experiments: (1) the Excite question log, for a total of 68,631 questions asked by the users of a search engine and (2) the TREC-8, TREC-9 and TREC-10 questions, for a total of 1,393 questions.

The noisy Excite log was cleaned up with two filters. First, we extracted only those lines containing one of the keywords *Where*, *When*, *What*, *Which*, *Why*, *Who*, *How*, *Why* or *Name*. Next, we eliminated the lines containing the phrase *"find information"* to avoid the bias towards Web searching questions. [1]

From the total of 25,272 Excite *What* questions[2] we have randomly selected a subset of 5,000 questions that were manually analyzed and classified. The decision of what question type to assign to a particular question was

[1] To our knowledge, only one other large scale question analysis is mentioned in the literature (Hovy et al., 2001).

[2] We emphasize the experiments involving *What* questions, since they provide the largest coverage and are considered to be the most ambiguous types of questions. Similar analyses were performed for the other types of questions, but are not reported here due to lack of space.

merely based on the possibility of implementing a procedure that would make use of this question type in the process of finding relevant information. For instance, a question like *What does Acupril treat?* expects a DISEASE as answer, which is doable in the sense that an ontology like WordNet does have a disease node with pointers to a large number of disease names. On the other hand, *What about this Synthyroid class action?* does not require a specific answer, but rather information related to a topic, and therefore no question type is assigned to this question (the type NONE is used instead). For the entire set of 5,000 questions, 361 categories are extracted.

2.1. Quantitative Analysis

To the end of observing the behavior and learning rate associated with question types, subsets of different sizes were created and the number of question types was determined for each subset. The measurements were performed using a 10-fold cross validation scheme on randomly selected samples of data.

Figure 1 plots the distribution of question types with respect to the subset size. It turns out that the number of question types grows sublinearly with the number of questions. Moreover, we noticed a behavior of the curve similar with *Heaps' Law* (Heaps, 1978), which relates the number of words in a text with the text size. *Heaps' Law* states that the size of the vocabulary for a text of size n is $V = Kn^{\beta} = O(n^{\beta})$.

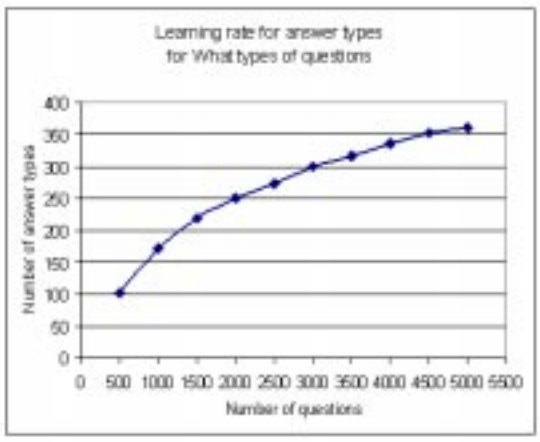


Figure 1: Number of question types vs. number of questions for *What* questions in the Excite log.

Denoting the number of question types with T_q and the number of questions with N_q, it follows:

$$\mathrm{T}_q = K{N_q}^{\beta} \quad (1)$$

The equation is solved by taking the log in both sides. For the Excite *What* set, it results a value of $K = 5.18$, respectively $\beta = 0.50$. The values of the two parameters are changed in the TREC *What* set: $K = 3.89$ and $\beta = 0.54$, which illustrates the difference in question types distribution for the uniform TREC set versus the noisy Excite set.

This is an interesting result, as it defines the behavior of question types with respect to the number of questions. Moreover, it gives us the capability of making estimates on what is the expected number of question types for N_q given questions. For instance, 10,000 questions will result in about 518 question types, 100,000 in about 1,638 question types, and so forth.

2.2. Qualitative Analysis

The qualitative analysis brings evidence for the organization of question types in semantic hierarchies, and supports the idea of incorporating semantics into information retrieval.

An analysis of the questions benchmarks suggested that the majority of question types are found in a general-specific (ISA) relation. This hypothesis is sustained by empirical evidence. We classified the questions into four categories as listed in Table 1[3]. It turns out that on average about 60% of the questions are clear general-specific questions. It is debatable whether or not the DEFINITION types of questions can be classified as general-specific questions or not. It is often the case that a definition requires a more general concept to explain an unknown entity (Prager et al., 2001), and therefore it could be considered as a general-specific information request. Under this hypothesis, it results an average of 80% of information requests being expressed along general-specific lines.

Information type	Frequency
Excite questions	
GENERAL-SPECIFIC	54.6%
DEFINITION	19.6%
NONE	14.8%
OTHER	10.8%
TREC questions	
GENERAL-SPECIFIC	65.0%
DEFINITION	20.9%
NONE	6.6%
OTHER	7.4%

Table 1: Information requests along general-specific lines

Figure 2 shows examples of annotated questions extracted from the Excite log, mapped on an *animal* hierarchy of question types.

The conclusion of these experiments is that the majority of information requests are expressed along general-specific lines, and therefore a semantic based retrieval system that exploits these relations would possibly increase the quality of the information retrieved. This idea was also expressed by (Berners-Lee et al., 2001) in the context of Semantic Web.

3. Conversion of WordNet to DD-encoding

On the one side, we have the users' information need expressed most of the times as a general-specific request.

[3]The OTHER category includes questions that require an answer that cannot be obtained by following a general-specific line. Examples of such question types are CAUSE, EFFECT, QUOTE|ALBUM, QUOTE|MOVIE, WORD-TRANSLATION, etc.

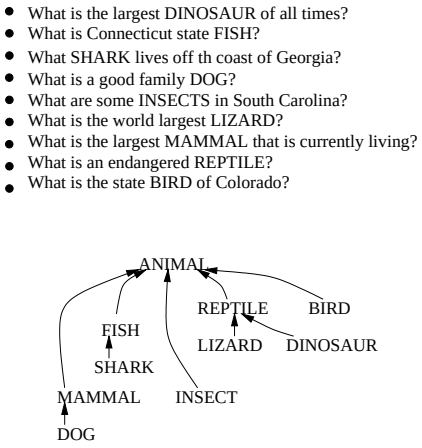


Figure 2: Question types mapped onto the *animal* hierarchy.

On the other side, we have WordNet (Miller, 1995) as the largest general purpose semantic network available today, which encodes about 86,605 general-specific (ISA) relations. We want to exploit as much as possible the semantic network structure of WordNet. To this end, we propose in this section a new encoding to be used for WordNet entries that would enable more efficient semantic searches. The so called *DD-encoding* was inspired by the Dewey Decimal code scheme used by librarians.

There are many times when keywords in a query are used with "generic" meanings and they are intended as representatives for entire categories of objects. *Foxes eat hens* is a statement that can be evaluated as a good match for *Animals eat meat*. Unfortunately, with current indexing and retrieval techniques this is not possible, unless both *animal* and *meat* are expanded with their subsumed concepts, which may sometimes become a tedious process. For this particular example, WordNet defines 7,980 concepts underneath *animal*, and there are 199 entries that inherit from *meat*, and therefore we end up with more than 1,500,000 (7,980 x 199) queries to cover the entire range of possibilities. Alternatively, if boolean queries are allowed and the OR operator is available, a query with 8,179 (7,980 + 199) terms can be used. None of these solutions seems acceptable and this is why none of them have been used so far.

We would like to find a way such that *fox* matches *animal* and we propose the employment of matching codes as an elegant solution to accomplish this task.

Finding the means that would allow for this type of matches is a problem of central interest for retrieval applications, as most information requests are expressed along general-specific lines. We want to retrieve documents containing *cat* in return to a search for *animal*, and retrieve *dachshund* and do not retrieve *cat* as the result of a search for *dog*.

To enable this type of general-specific searches and at the same time take advantage of the semantic structure already encoded in WordNet, we propose the employment of a codification scheme similar with the one used in librarian systems, and associate a code to each entry in WordNet.

The role of this code is to make evident to an external tool, such as an indexing or retrieval process, the relation that exists between inter-connected concepts. No information can be drawn from the simple reading of the *animal* and *dog* strings. Things are completely different when we look at *13.1* and *13.1.7*: the *implicit* relation between the two tokens has now been turned into an *explicit* one.

A code is assigned to each WordNet entry such that it replicates its parent code, and adds a unique identifier. For instance, if *animal* has code *13.1*, then *chordate*, which is a directly subsumed concept, has code *13.1.29*, *vertebrate* has code *13.1.29.3*, and so forth. Figure 3 illustrates a snapshot from the noun WordNet hierarchy and shows the *DD-codes* attached to each node. This encoding creates the grounds for matching at semantic levels in a manner similar with the lexical matches already employed by several information retrieval systems.

To our knowledge, this is a completely new approach taken towards the goal of making possible searches at semantic levels. The idea underneath this encoding is very simple but it allows for a powerful operator: the *semantic wildcard*.

3.1. Technical Issues

There are several implementation issues encountered during WordNet transformation, and we shall address them in this section.

Specifically, the new encoding is created using the following algorithm:

---

1. Start with the top of WordNet hierarchies. For each top, load its hyponyms, and for each hyponym go to step 2.
2. Execute the following steps:
2.1. Assign to the current synset the DD-code of its parent plus an unique identifier that is generated as a number in a successive series.
2.2. If the current synset has been already assigned a DD-code, then generate a special link between its parent and the current synset itself.
2.3. Load all hyponyms of current synset and go to step 2.

The algorithm performs a recursive traversal of the entire WordNet hierarchy and generates codes. A code is associated with a synset, and we created a list of pairs containing a synset offset (the current WordNet encoding) and a *DD-code*.

It is worth mentioning the case of multiple inheritance, handled by the Dewey classification system as an addition made for a particular category. For instance, 675+678 means *leather and rubber*. This solution is not satisfactory for our purpose, since it may result in very long codes. Instead, a list of *special links* (generated in step 2b) is created, containing all the links between a *second parent* and a child. For example, if *house* inherits from both *domicile* and *building*, we have the code 1.2.1.32.12.23 for *house*, 1.2.1.32.28.6 for *domicile* and 1.2.1.32.12 for *building*, and in addition a special link is generated to indicate that *domicile* is the parent of *house* even if no direct matching can be performed.

For the entire noun hierarchy in WordNet, 74,488 *DD-codes* were generated. In addition, 4,280 multiple inheritance links were created. The average length of a code is 16 characters. Given the fact that disk space is a cheap re-

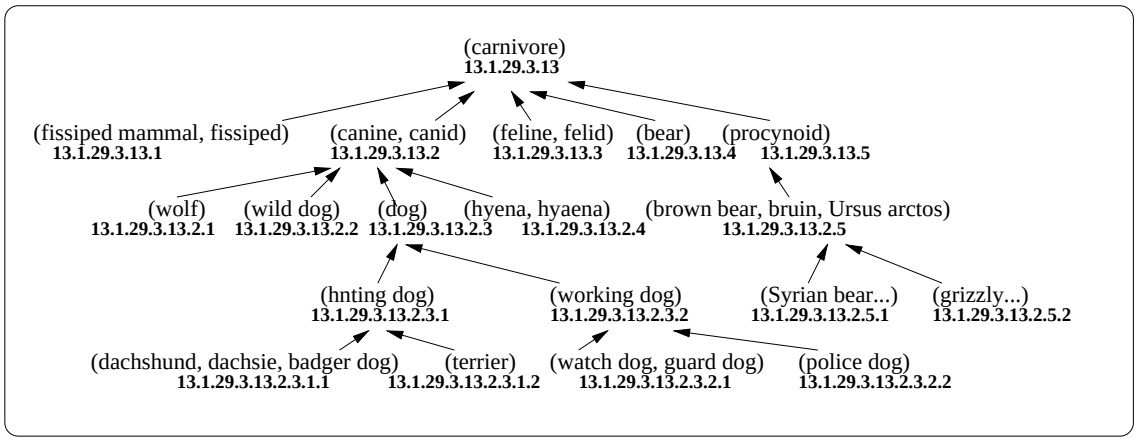


Figure 3: *DD-codes* assigned to a sample of the WordNet hierarchy

source, the length of the codes does not represent a real disadvantage of the proposed approach. Moreover, one should take into consideration that no optimizations were sought in the process of code generation. A simple strategy, like the usage of all 256 ASCII characters instead of using only the 1-9 digits, can shorten significantly the length of the codes (e.g. 1.2.1.32.12.23 changes into 1.2.1.z.b.f). Approaches like Huffman code or other compression methods can be as well exploited for this purpose, but we will not consider these issues here.

4. The IRSLO System

Our improved semantic based information retrieval system comprises the same main components as found in any other retrieval system.

4.1. Question/Query Processing

This stage usually includes a keyword selection process. It may sometimes imply keyword stemming or other processing, and in most cases keywords to be employed in the retrieval stage are selected based on weights, frequencies and stop-words lists.

In IRSLO, we start this stage with a simple tokenization and part of speech tagging using Brill tagger (Brill, 1995). Next, collocations are identified based on WordNet definitions. We also identify the baseform of each word.

Depending on the notation employed by the user, we distinguish three keyword types. (1) Words with a semantic wildcard, denoted with #. (2) Words to be searched by their *DD-code*, denoted with @ (synonymy marker). (3) Words with no special notation, to be sought in the index in their given form. By default, we assume a # assigned to the answer type word, and no other notation for the rest of the words. All words that are denoted with # or @ are passed on to a word sense disambiguation component that solves their semantic ambiguity. Alternatively, this step can be skipped and a default sense of one with respect to WordNet is assigned, with reasonable precision (over 75% as measured on SemCor). The results reported in this paper are based on a simplified implementation that considers the second alternative. Next, *DD-codes* are assigned to words in text and subsequently used in the retrieval process. *DD-codes* are currently assigned only to nouns, considered to be the most informative words. See section 3. for more details regarding *DD-encoding*.

We also face the task of identifying relevant keywords to be included in a query. Extensive analysis of keywords identification was previously reported in (Pasca, 2001). We use a simplified keywords identification procedure, based on the following rules:

1. Use all proper nouns and quoted words.
2. Use all nouns.
3. Use all adjectives in superlative form.
4. Use all numbers (cardinals).
5. If more than 200 documents are returned, use the adjectives modifying the first noun phrase.
6. If no documents are returned, drop the nouns acting as modifiers. Particular attention is paid to abstract nouns, such as type, kind, name, where the importance of the roles played by a head and a modifier in a noun phrase are interchanged.

Any of these keywords may be expressed using its corresponding *DD-code*. The answer type word is also important. It practically denotes the type of information sought, whether is a *country*, an *animal*, a *fish*, etc. We use a simple approach that selects the answer type as the head of the first noun phrase. There are few exceptions from this rule, consisting of the cases where the head is an abstract noun like *name*, *type*, *variety* and so forth, and in such cases we select its modifier. If the answer detected is of a generic type, such as *person*, *location*, *organization*, then we replace it with the corresponding named entity tag. Otherwise, the answer type word is assigned a # semantic wildcard. Notice that the answer type selection process is invoked only if there is no word a priori denoted with #.

After all these processing steps, we end up with a query in IRSLO format. The words that were assigned a semantic wildcard # are now represented as *DD-code**. The words with a synonymy marker are simply replaced with their *DD-code* (thereby allowing for the retrieval of synonym words in addition to the word itself). The other words are replaced with their baseform. See Section 5.4. for representation examples.

4.2. Document Processing

Typically, documents are simply tokenized and terms are extracted, in preparation for the indexing phase. Optionally, stop-words are eliminated and words are stemmed prior to indexing.

In IRSLO, documents are processed following similar steps to question processing. First, the text is tokenized and part of speech tagged. We have an additional component that involves named entity recognition (Lin, 1994). Next, we identify compound words, apply a disambiguation algorithm or, alternatively, assign to each word its default sense from WordNet. Finally we assign to each noun its corresponding *DD-code*.

At this stage, we also identify paragraphs and store them as one paragraph per line. This helps improving efficiency during paragraph retrieval.

4.3. Indexing and Retrieval

The indexing process is not different in any ways with respect to a classic information retrieval system. A TF/IDF weight is assigned to each term. We index complex terms, including the *DD-codes* attached to each noun and the named entity tags, when available. No additional stemming or stop-words elimination is performed. The retrieval system allows for flexible searches, including regular expressions. Based on *DD-codes*, we have the capability of using the *semantic wildcard* operator, in addition to the lexical wildcard. We also have the capability of retrieving named entities of a certain type (e.g. perform a search for *person*). Moreover, we allow for boolean operators and for the new *paragraph operator* for a more focused search. Documents are ranked using the TF/IDF weight associated with each keyword.

5. Experiments with IRSLO

This section focuses on the application of the *semantic wildcard* and *paragraph operator* within the IRSLO system. First, the semantic wildcard enables searches for information along general-specific lines. Second, the paragraph indexing component limits the scope of keywords search to a single paragraph, rather than an entire document.

5.1. Experimental Setup

Several standard text collections are made available through the Information Retrieval community. For our experiments, we have selected the *L.A. Times* collection, which includes a fairly large number of documents. There are more than 130,000 documents adding up to 500MB of text. *L.A. Times* is part of the TREC (Text REtrieval Conference) collections.

The main advantage of standard text collections is the fact that question sets and relevance judgments are usually provided in association with the document collection.

About 1,393 questions have been released during the TREC-8, TREC-9 and TREC-10 Q&A TREC competitions. Relevance judgments are provided for the first two competitions, i.e. for 893 questions. From the 893 questions, we selected only the *What* type of questions, as being the most ambiguous types of questions and the best candidates for the semantic wildcard operator. Subsequently, we identified those questions known to have an answer in the *L.A. Times* collection[4], and out of these 75 questions were randomly selected for further tests.

For this question set, we have the knowledge about the information expected in response to each question (answer patterns provided by the TREC community). We also have a list of *docid*-s pointing to documents containing the answer for each question (list of documents judged to contain a correct answer by TREC assessors). This information helps us measure *precision* and *recall*.

5.2. Evaluating Retrieval Effectiveness

A common methodology in evaluating information retrieval systems consists in measuring *precision* and *recall*. *Precision* is defined as the number of relevant documents retrieved over the total number of documents retrieved. *Recall* is defined as the number of relevant documents retrieved over the total number of relevant documents found in the collection. Additionally, the *F-measure* proposed in (Van Rijsbergen, 1979) provides the means for combining recall and precision into one single formula, using relative weights.

$$F_{measure} = \frac{(\beta^2 + 1.0) * P * R}{(\beta^2 * P) + R}$$

where P is precision, R is recall and β is the relative importance given to recall over precision. During the system evaluations reported here, we considered both precision and recall of equal importance, and therefore β is set to 1.

Moreover, we employ the *success rate* measure (Woods, 1997) as an indicative of how many questions were answered by the system. The *success rate* for a question/query is 1 if relevant documents/answers are found, and 0 otherwise.

Finally, we evaluate IRSLO results using the TREC Q&A score, with a different mark assigned to an answer depending on its position within the final rank. A correct answer on the first position results in a maximum score of 1.00. The second position gets 0.50, the third position is scored with 0.33, the fourth with 0.25 and the fifth and last one acceptable receives 0.20 points.

5.3. Experiments

Three types of experiments were performed, to evaluate the performance of the new *semantic wildcard* and *paragraph operator*.

Experiment 1. Extract the keywords[5] from each question and run the queries formed in this way against a classic index created with the *L.A. Times* collection. The purpose of this experiment is to simulate classic keyword-based retrieval systems. The ranking is provided through a TF/IDF weighting scheme.

Experiment 2. Extract the keywords from each question and run the queries against the paragraph index. In paragraph

[4]The set of 893 questions was devised to ensure an answer in the entire TREC collection, including 2.5GB of text in addition to the *LA Times* collection that we employ in our experiments

[5]See Section 4.1. for the keywords selection procedure

indexing, we use a boolean model that includes the *paragraph operator*, plus a measure that determines the closeness among keywords to rank the paragraphs.

Experiment 3. Again, extract keywords from questions and run them against the paragraph index. Additionally, we allow the *semantic wildcard* (including named entity tags) to be specified in the keywords.

The results of experiments 1 and 2 are compared, to show the power of paragraph indexing. Experiments 2 and 3 provide comparative results to support the use of semantics, specifically the *semantic wildcard*.

The first experiment represents a classic keyword-based information retrieval run, and therefore we evaluate it in terms of *precision*, *recall* and *F-measure*. The second and third experiments are also evaluated in terms of *precision*, *recall* and *F-measure*. Additionally, we use the *success rate* and *TREC score*.

5.4. Walk-through Examples

This section gives several running examples of the IRSLO system, using the *semantic wildcard* and *paragraph operator*.

Example 1. What is the brightest star visible from Earth?
Relevant paragraph. In the year 296036 , Voyager 2 will make its closest approach to Sirius , the brightest star visible from Earth .
Comments. The query formed in this case is star# AND bright AND Earth. Only two answers are found by the system, and the one listed above, which is the correct one, is ranked on the first position. Sirius is defined in WordNet as a star, and consequently was annotated as such in the text.

Example 2. What kind of sports team is the Buffalo Sabres?
Relevant paragraph. Another religious broadcasting company , Tri - State Christian TV Inc. of Marion , Ill. , which was set up with the help of loan guarantees from Trinity , announced recently that it has purchased WNYB Channel 49 in Buffalo , N.Y. , from the Buffalo Sabres hockey team for $2.5 million .
Comments. The query employed is team# AND Buffalo AND Sabres. The original query team# AND sport AND Buffalo AND Sabres did not return any answers, and consequently the back off scheme was invoked and dropped noun modifiers. A total of six paragraphs are found in return to this question, all of them correct.

Example 3. What U.S. Government agency registers trademarks?
Relevant paragraph. After your application arrives at the Patent Office , it is turned over to an attorney who determines whether there is anything " confusingly similar "between your trademark and others [...]
Comments. Patent Office is a type of Government agency, and therefore the query U.S. AND government_agency# AND trademark leads to the correct answer.

Example 4. What cancer is commonly associated with AIDS?
Relevant paragraph. A team of transplant specialists at City of Hope National Medical Center in Duarte is among several groups nationwide that plan to test the experimental procedure on a small number of patients with AIDS - related lymphomas , or tumors of the lymph nodes .
Comments. The query employed is cancer# AND AIDS. The answer was found at rank 4, and it seems that none of the teams in the TREC competition identified this answer, because there is no direct reference in the text to cancer, but only a hidden relation from lymphomas to cancer. Our semantic model has the capacity to detect such non-explicit relations.

5.5. Results

Tests were performed using the benchmark of 75 questions. For each question, we run three experiments, as mentioned earlier. (1) Keyword-based information retrieval using a TF/IDF scheme. (2) Paragraph indexing and retrieval (i.e. enable the paragraph operator). (3) An experiment that involves both paragraph operator and semantic wildcard.

Precision, *recall* and *F-measure* are determined for all these experiments. We have also determined *success rate* and *TREC score*.

Ten sample requests of information are presented below, with their evaluations shown in Table 2. The following notations are used: P = *precision*, R = *recall*, F = *F-measure*, SR = *Success Rate*, TS = *TREC score*.

1. What American composer wrote the music for "West Side Story"?
2. What U.S. Government agency registers trademarks?
3. What U.S. state's motto is "Live free or Die"?
4. What actor first portrayed James Bond?
5. What animal do buffalo wings come from?
6. What cancer is commonly associated with AIDS?
7. What city does McCarren Airport serve?
8. What instrument does Ray Charles play?
9. What is the population of Japan?
10. What is the tallest building in Japan?

Cumulative results for all 75 questions are compared in Table 2. It turns out that the *F-measure* doubles when paragraph indexing is used with respect to document indexing, with increased *precision* and lower *recall*, as expected. The *success rate* is determined for the second and third experiments to evaluate the effect of the *semantic wildcard* over simple paragraph indexing, and an increase of 17% is observed. As of the *TREC score*, the additional use of semantics brings a gain of 34% with respect to simple paragraph indexing.

These results are very encouraging, and in agreement with the suggestions made in (Light et al., 2002) that query expansion and semantic relations are essential for increased performance, for information retrieval in general and Q&A systems in particular.

6. Related Work

Significant work has been performed in the field of semantics applied to information retrieval. The most important directions include: (1) query expansion (Voorhees, 1998), (2) phrase indexing (Strzalkowski et al., 1996), (3) conceptual indexing (Woods, 1997), (4) semantic indexing (Sussna, 1993), (Krovetz, 1997). In addition, the Semantic Web is a new field that considers the use of semantics for Web applications (Berners-Lee et al., 2001).

7. Conclusion

This paper has introduced the *semantic wildcard*, a novel operator that enables the use of semantics in information retrieval applications. The *semantic wildcard*, together with the new *paragraph operator*, were implemented in the IRSLO system. Experiments were performed on a collection of 130,000 documents with 75 *What*-questions extracted from the questions released during TREC competitions. Three experiments were performed. (1) One that

Question number	Experiment												
	1. Classic IR			2. Par.op.					3. Sem.wildcard + par.op.				
	P	R	F	P	R	F	SR	TS	P	R	F	SR	TS
1	0.14	0.21	0.17	0.50	0.07	0.12	1	1.00	0.75	0.86	0.80	1	1.00
2	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	1.00	1.00	1.00	1	1.00
3	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	1.00	0.67	0.80	1	1.00
4	0.25	0.44	0.32	0.43	0.17	0.24	1	1.00	0.16	1.00	0.27	1	0.00
5	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.43	1.00	0.60	1	0.33
6	0.08	0.84	0.14	0.03	1.00	0.03	1	0.00	0.37	0.74	0.49	1	0.25
7	1.00	1.00	1.00	1.00	1.00	1.00	1	1.00	1.00	1.00	1.00	1	1.00
8	0.00	0.00	0.00	0.00	0.00	0.00	0	0.00	0.38	1.00	0.55	1	0.50
9	0.03	1.00	0.05	0.04	0.50	0.07	1	0.00	0.08	0.33	0.13	1	1.00
10	0.03	0.50	0.06	0.40	0.50	0.44	1	1.00	1.00	1.00	1.00	1	1.00

Table 2: Precision, recall, F-measure, success rate and TREC score for 10 sample requests of information

Measure	Experiment		
	1. Classic IR	2. Par.op.	3. Sem.wildcard. + par.op.
Precision	0.05	0.12	0.12
Recall	0.66	0.57	0.61
F-measure	0.092	0.19	0.20
Success rate	-	66.0%	77.3%
TREC score	-	43.4%	58.3%

Table 3: Comparative results for (1) keyword-based information retrieval (2) paragraph operator and (3) paragraph operator + semantic wildcard

simulates classic keyword-based information retrieval with a TF/IDF weighting scheme. (2) A second experiment that implements the *paragraph operator*. (3) Finally, a third experiment where both *semantic wildcard* and *paragraph operator* are employed. Various measures were used to evaluate the performance attained during these experiments, and all measures have proved the efficiency of our *semantic wildcard* operator, respectively the *paragraph operator*, over keyword-based retrieval techniques. As a follow-up analysis, it would be interesting to determine the *min* and *max* bounds proposed in (Light et al., 2002) for the precision achievable on a question set when the semantic wildcard is enabled.

8. References

T. Berners-Lee, J. Hendler, and O. Lassila. 2001. The Semantic Web. *Scientific American*, 1(501), May.

E. Brill. 1995. Transformation-based error driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–566, December.

H.S. Heaps. 1978. *Information Retrieval, Computational and Theoretical Aspects*. Academic Press.

E. Hovy, L. Gerber, U. Hermjakob, C.-Y. Lin, and D. Ravichandran. 2001. Toward semantics-based answer pinpointing. In *Proceedings of the Human Language Technology Conference, HLT 2001*, San Diego, CA.

R. Krovetz. 1997. Homonymy and polysemy in information retrieval. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL-97)*, pages 72–79.

M. Light, G.S. Mann, E. Riloff, and E. Breck. 2002. Analyses for elucidating current question answering technology. *Journal of Natural Language Engineering (forthcoming)*.

D. Lin. 1994. Principar - an efficient, broad-coverage, principle-based parser. In *In Proceedings of the Fifteenth International Conference on Computational Linguistics COLING-ACL '94*, pages 42–48, Kyoto, Japan.

R. Mihalcea. 1999. Word sense disambiguation and its application to the Internet search. Master's thesis, Southern Methodist University.

G. Miller. 1995. Wordnet: A lexical database. *Communication of the ACM*, 38(11):39–41.

M. Pasca. 2001. *High performance question answering from large text collections*. Ph.D. thesis, Southern Methodist University.

J. Prager, D. Radev, and K. Czuba. 2001. Answering what-is questions by virtual annotation. In *Proceedings of the Human Language Technology Conference, HLT 2001*, San Diego, CA.

T. Strzalkowski, L. Guthrie, J. Karigren, J. Leistensnider, F. Lin, J. Perez-Caballo, T. Straszheim, J. Wang, and J. Wilding. 1996. Natural language information retrieval, TREC-5 report. In *Proceedings of the 5th Text Retrieval Conference (TREC-5)*, pages 291–314, Gaithersburg, Maryland, November.

M. Sussna. 1993. Word sense disambiguation for free-text indexing using a massive semantic network. In *Proceedings of the second international conference on Information and knowledge management CIKM '93*, pages 67–74, Washington, November.

C.J. Van Rijsbergen. 1979. *Information Retrieval*. London: Butterworths. available on-line at http://www.dcs.gla.ac.uk/Keith/Preface.html.

E.M. Voorhees. 1998. Using WordNet for text retrieval. In *WordNet, An Electronic Lexical Database*, pages 285–303. The MIT Press.

W.A. Woods. 1997. Conceptual indexing: A better way to organize knowledge. Technical Report SMLI TR-97-61, Sun Microsystems Laboratories, April. available online at: *http://www.sun.com/ research/techrep/ 1997/abstract-61.html*.

Statistical Query Disambiguation, Translation and Expansion in Cross-Language Information Retrieval

Fatiha Sadat*, Akira Maeda†, Masatoshi Yoshikawa‡*, Shunsuke Uemura*

* Graduate School of Information Science, Nara Institute of Science and Technology (NAIST)
8916-5 Takayama, Ikoma. Nara 630-0101. Japan

† CREST, Japan Science and Technology Corporation (JST)
‡ National Institute of Informatics (NII)
{fatia-s, aki-mae, yosikawa, uemura}@is.aist-nara.ac.jp

Abstract

Query expansion is considered as one of the most important methods in improving the effectiveness of information retrieval. By combining query expansion with dictionary-based translation and statistics-based disambiguation, in order to overcome query terms ambiguity, information retrieval should become much more efficient. In the present paper, we focus on query terms disambiguation via, a combined statistical method both before and after translation, in order to avoid source language ambiguity as well as incorrect selection of target translations. Query expansion techniques through relevance feedback were performed prior to either the first or the second disambiguation processes. We tested the effectiveness of the proposed combined method, by an application to a French-English Information Retrieval. Experiments involving TREC data collection revealed the proposed disambiguation and expansion methods to be highly effective.

1. Introduction

In recent years, the number of studies concerning Cross-Language Information Retrieval (CLIR) has grown rapidly, due to the increased availability of linguistic resources for research. Cross-Language Information Retrieval consists of providing a query in one language and searching document collections in one or more languages. Therefore, a translation form is required. In the present paper, we focus on query translation, disambiguation and expansion in order to improve the effectiveness of information retrieval through various combinations of these methods. First, we are interested to find retrieval methods that are capable of performing across languages and which do not rely on scarce resources such as parallel corpora. Bilingual Machine Readable-Dictionaries (MRDs), more prevalent than parallel texts, appear to be a good alternative. However, simple translations tend to be ambiguous and yield poor results. A combination that includes a statistical approach for a disambiguation can significantly reduce errors associated with *polysemy*[1] in dictionary translation. In addition, automatic query expansion, which has been known to be among the most important methods in overcoming the word mismatch problem in information retrieval, is also considered. As an assumption to reduce the effect of ambiguity and errors that a dictionary-based method would cause, a combined statistical disambiguation method is performed both prior to and after translation. Although, the proposed information retrieval system is general across languages in information retrieval, we conducted experiments and evaluations concerning French-English information retrieval.

The remainder of the present paper is organized as follows. Section 2 provides a brief overview of related works. Both dictionary-based and the proposed disambiguation methods are described in Section 3. A combination involving query expansion is described in Section 4. Evaluation and discussion of the experiments of the present study are presented in Section 5. Section 6 involves Word Sense Disambiguation and Section 7 describes the conclusion of the present paper.

2. Related Research in CLIR

The potential of knowledge-based technology has led to increasing interest in CLIR. The query translation of an automatic MRD, on its own, has been found to lead to a drop in effectiveness of 40-60 % compared to monolingual retrieval (Hull and Grefenstette, 1996; Ballesteros and Croft, 1997). Previous studies have used MRDs successfully, for query translation and information retrieval (Yamabana et al., 1996; Ballesteros and Croft, 1997; Hull and Grefenstette, 1996). However, two factors limit the performance of this approach. The first is that many words do not have a unique translation and sometimes the alternate translations have very different meanings (*homonymy and polysemy).* The fact that a single word may have more than one sense is called *ambiguity.* Translation ambiguity significantly exacerbates the problem in CLIR (Oard, 1997). Most of the previously proposed disambiguation strategies rely on statistical approaches, but without considering ranking or selection of source query terms, which affect directly the selection of target translations. The second challenge is that dictionary may lack some terms that are essential for a correct interpretation of the query. In the present study, we propose the concept of the combined statistical disambiguation technique, applied prior to and after dictionary translation to solve lexical semantic ambiguity. In addition, a monolingual thesaurus is introduced to overcome bilingual dictionary limitation. Automatic query expansion through relevance feedback, which has been used extensively to improve the effectiveness of an information retrieval (Ballesteros and Croft, 1997; Loupy et al., 1998), is considered. Selection of expansion terms was performed through various means. In the present study, we use a ranking factor to select the best expansion terms-those related to all source query terms, rather than to just one query term.

[1] Polysemy is a word, which has more than one meaning.

3. Translation/Disambiguation in CLIR

There are two types of lexical semantic ambiguity with which a machine translation system must contend: there is ambiguity in the source language where the meaning of a word is not immediately apparent but also ambiguity in the target language when a word is not ambiguous in the source language but it has two or more possible translations (Hutchins and Sommers, 1992). In the present research, query translation/disambiguation phases are performed after a simple *stemming* process of query terms, replacing each term with its inflectional root and each verb with its infinitive form, as well removing most plural word forms, stop words and stop phrases. Three primary tasks are completed using the translation/disambiguation module. First, an *organization of source query terms*, which is considered key to the success of the disambiguation process, will select best pairs of source query terms. Next a *term-by-term translation* using the dictionary-based method (Sadat et al., 2001), where each term or phrase in the query is replaced by a list of its possible translations, is completed. Missing words in the dictionary, which are essential for the correct interpretation of the query. This may occur either because the query deals with a technical topic, which is outside the scope of the dictionary or because the user has entered some form of abbreviations or slang, which is not included in the dictionary (Oard, 1997). To solve this problem, an automatic *compensation* is introduced, via synonym dictionary or existing thesaurus in the concerned language. This case requires an extra step to look up the query term in the thesaurus or synonym dictionary, find equivalent terms or synonyms of the targeted source term, thus performing a query translation. In addition, short queries of one term are concerned by this phase. The third task, *disambiguation of target translations*, selects best translations related to each source query term. Finally, documents are retrieved in target language.

Figure 1 shows thee overall design of the proposed information retrieval system. Query expansion will be applied prior to and/or after the translation/disambiguation process. Among the proposed expansion strategies are, relevance feedback and thesaurus-based expansion, which could be interactive or automatic.

3.1. Organization of Source Query Terms

All possible combinations of source query terms are constructed and ranked depending on their mutual co-occurrence in a training corpus. A type of statistical process called *co-occurrence tendency* (Maeda et al., 2000; Sadat et al., 2001) can be used to accomplish this task. Methods such as Mutual Information MI (Church and Hanks, 1990), the Log-Likelihood Ratio LLR (Dunning, 1993), the Modified Dice Coefficient or Gale's method (Gale and Church, 1991) are all candidates to the co-occurrence tendency.

3.2. Co-occurrence Tendency

If two elements often co-occur in the corpus, then these elements have a high probability of being the best translations among the candidates for the query terms. The selection of pairs of source query terms to translate as well as the disambiguation of translation candidates in order to select target ones, is performed by applying one of the statistical methods based on co-occurrence tendency, as follows:

• *Mutual Information (MI)*

This estimation uses *mutual information* as a metric for significance of word co-occurrence tendency (Church and Hanks, 1990), as follows:

$$MI(w_1, w_2) = \log \frac{Prob(w_1, w_2)}{Prob(w_1)Prob(w_2)}$$

Here, $Prob(w_i)$ is the frequency of occurrence of word w_i divided by the size of the corpus N, and $Prob(w_i, w_j)$ is the frequency of occurrence of both w_i and w_j together in a fixed window size in a training corpus, divided by the size of the corpus N.

• *Log-Likelihood Ratio (LLR)*

The Log-Likelihood Ratio (Dunning, 1993) has been used in many researches. LLR is expressed as follows:

$$-2\log \lambda = K_{11}\log \frac{K_{11}N}{C_1R_1} + K_{12}\log \frac{K_{12}N}{C_1R_2} + K_{21}\log \frac{K_{21}N}{C_2R_1} + K_{22}\log \frac{K_{22}N}{C_2R_2}$$

Where, $C_1 = K_{11} + K_{12}$, $C_2 = K_{21} + K_{22}$, $R_1 = K_{11} + K_{21}$, $R_2 = K_{12} + K_{22}$, $N = K_{11} + K_{12} + K_{21} + K_{22}$,
K_{11} = *frequency of common occurences of word* w_i *and word* w_j , K_{12} = *corpus frequency of word* w_i - K_{11} ,
K_{21} = *corpus frequency of word* w_j - K_{11} , $K_{22} = N - K_{12} - K_{22}$.

3.3. Disambiguation of Target Translations

A word is *polysemous* if it has senses that are different but closely related. As a noun, for example, *right* can mean something that is morally acceptable, something that is factually correct, or one's entitlement. A two-terms disambiguation of translation candidates can be applied (Maeda et al., 2000; Sadat et al., 2001) is required, following a dictionary-based method. All source query terms are generated, weighed, ranked and translated for a disambiguation through co-occurrence tendency. The classical procedure for a *two-term disambiguation*, is described as follows:

1. Construct all possible combinations of pairs of terms, from the translation candidates.
2. Request the disambiguation module to obtain the co-occurrence tendencies. The window size is set to one paragraph of a text document rather than a fixed number of words.
3. Choose the translation, which shows the highest co-occurrence tendency, as the most appropriate.

As illustrated in Figure 2, the disambiguation procedure is used for two-term queries due to the computational cost (Maeda et al., 2000). In addition, the primary problem concerning long queries, involves the selection of pairs of terms, as well as the order for disambiguation. We propose and compare two methods for *n-term disambiguation,* for queries of two or more terms. The first method is based on a ranking of pairs of source query terms before the translation and disambiguation of target translations. The key concept in this step is to maintain the ranking order from the organization phase and perform translation and disambiguation starting from the most informative pair of source terms, i.e. a pair of source query terms having the highest co-occurrence tendency. Co-occurrence tendency is involved in both phases,

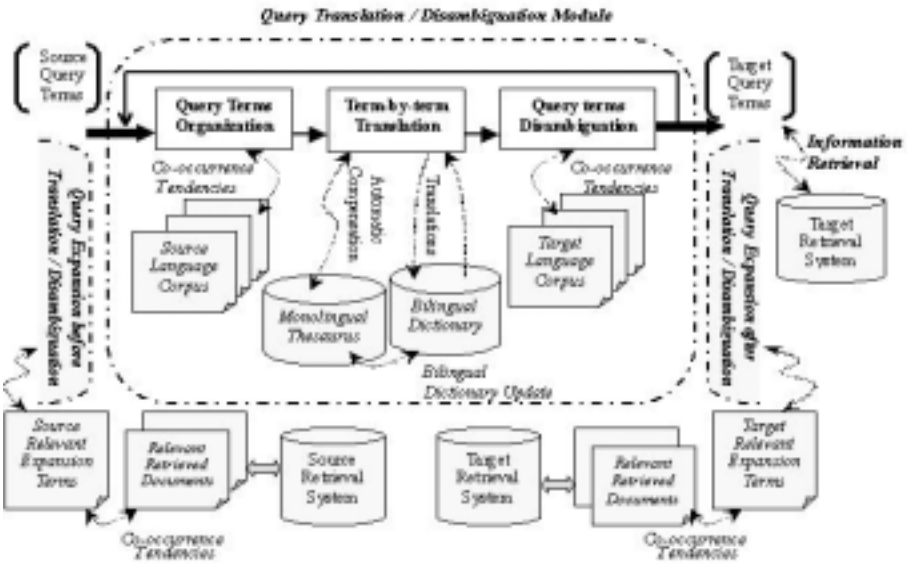


Figure 1: An overview of the Proposed Information Retrieval System
(In this research, source/target languages are French/English)

organization for source language and disambiguation for target language. The second method is based on a ranking of target translation candidates. These methods are described as follows: Suppose, Q represents a source query with n terms $\{s_1, s_2, \ldots, s_n\}$.

First Method: *(Ranking source query terms and disambiguation of target translations)*

1. Construct all possible combinations of terms of one source query: (s_1, s_2), (s_1, s_3), … (s_{n-1}, s_n).
2. Rank all combinations, according to their co-occurrence tendencies[2] toward highest values.
3. Select the combination (s_i, s_j), having the highest co-occurrence tendency, where at least one translation of the source terms has not yet been fixed.
4. Retrieve all related translations to this combination from the bilingual dictionary.
5. Apply a two-term disambiguation process to all possible translation candidates,
6. Fix the best target translations for this combination and discard the other translation candidates.
7. Go to the combination having the next highest co-occurrence tendency, and repeat steps 3 to 7 until every source query term's translation is fixed.

Second Method: *(Ranking and disambiguation of target translations)*

1. Retrieve all possible translation candidates for each source query term s_i, from the bilingual dictionary.
2. Construct sets of translations T_1 , T_2, …, T_n related to each source query term s_1 , s_2, …, s_n, and containing all possible translations for the concerned source term. For example, $T_i = \{t_{i1}, \ldots, t_{in}\}$ is the translation set for term s_i.
3. Construct all possible combinations of elements of different sets of translations. For example, (t_{11}, t_{21}), (t_{11} , t_{22}), …. (t_{ij} , t_{nk}),
4. Select the combination having the highest co-occurrence tendency[2].
5. Fix these target translations, for the related source terms and discard the other translation candidates.
6. Go to the next highest co-occurrence tendency and repeat step 4 through 6, until every source query term's translation is fixed.

Examples using the two proposed disambiguation methods are shown in Figures 3 and 4 for source English queries and target French translations.

4. Query Expansion in CLIR

Following the research reported by (Ballesteros and Croft, 1997) on the use of local feedback, the addition of terms that emphasize query concepts in the pre and post-translation phases improves both precision and recall. In the present study, we have proposed the combined automatic query expansion before and after translation through a relevance feedback. Original queries were modified, using judgments of the relevance of a few highly ranked documents, obtained by an initial retrieval, based on the presumption that those documents are relevant. However, query expansion must be handled very carefully. Simply selecting any expansion term from relevant retrieved documents could be risky. Therefore, our selection is based on the co-occurrence tendency in conjunction with all terms in the original query, rather than with just one query term. Assume that we have a query Q with n terms, $\{term_1 \ldots term_n\}$, then a ranking factor based on the co-occurrence frequency between each term in the query and an expansion term candidate, already extracted from the top retrieved relevant documents, is evaluated as:

$$Rank(expterm) = \sum_{i=1}^{n} co - occur(term_i, exp\ term)$$

[2] Co-occurrence tendency is based on one of the statistical methods: Mutual Information or Log-Likelihood Ratio, …

where, *co-occur(term$_i$, expterm)* represents the co-occurrence tendency between a query term *term$_i$* and the targeted expansion candidate *expterm*. *Co-occur(term$_i$, expterm)* can be evaluated by any estimation technique, such as mutual information or the log-likelihood ratio. All co-occurrence values were computed and then summed for all query terms (*i =1 to n*). An expansion candidate having the highest rank was then selected as an expansion term for the query Q. Note that the highest rank must be related to at least the maximum number of terms in the query, if not all query terms. Such expansion may involve several expansion candidates or just a subset of the expansion candidates.

5. Experiments and Evaluation

Experiments to evaluate the effectiveness of the two proposed disambiguation strategies, as well as the query expansion, were performed using an application of French-English information retrieval, i.e. French queries to retrieve English documents.

5.1. Linguistics Resources

Test Data: In the present study, we used test collection 1 from the TREC[3] data collection. Topics 63-150 were considered as English queries and were composed of several fields. Tags <num>, <dom>, <title>, <desc>, <smry>, <narr> and <con> denote topic number, domain, title, description, summary, narrative and concepts fields, respectively. Key terms contained in the title field <title> and description field <desc>, an average of 5.7 terms per query, were used to generate English queries. Original French queries were constructed by a native speaker, using manual translation.

Monolingual Corpora: The Canadian Hansard corpus (parliament debates) is a bilingual French-English parallel corpus, which contains more than 100 million words of English text as well as the corresponding French translations. In the present study, we used Hansard as a monolingual corpus for both French and English languages.

Bilingual Dictionary: COLLINS French-English dictionary was used for the translation of source queries.

Monolingual Thesaurus: EuroWordNet (Vossen, 1998) a lexical database was used to compensate, for possible limitations in the bilingual dictionary.

Stemmer and Stop Words: Stemming was performed using the English *Porter*[4] *Stemmer*. A special French stemming was developed and used in these experiments.

Retrieval System: The *SMART Information Retrieval System*[5] was used to retrieve both English and French documents. SMART is a vector model, which has been used in several studies concerning Cross-Language Information Retrieval.

5.2. Experiments and Results

A retrieval using original English/French queries was represented by *Mono_Fr/Mono_Eng* methods, respectively. We conducted two types of experiments. Those related to the query translation/disambiguation and those related to the query expansion before and/or after translation. Document retrieval was performed using original and constructed queries by the following methods. *All_Tr* is the result of using all possible translations for each source query term, obtained from the bilingual dictionary. *No_DIS* is the result of no disambiguation, which means selecting the first translation as the target translation for each source query term. We tested and evaluated two methods fulfilling the disambiguation of

[3] http://trec.nist.gov/data.html
[4] http://bogart.sip.ucm.es/cgi-bin/webstem/stem
[5] ftp://ftp.cs.cornell.edu/pub/smart

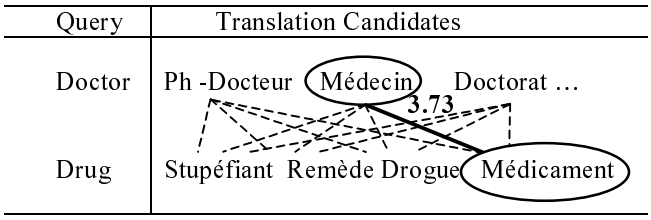


Figure 2: Two-Term Disambiguation Process

Highest co-occurrence tendencies for combinations of target translation candidates are as follows: (médecin, médicament), (médecin, remède), (médecin, drogue) …
Source French query: "doctor drug". Translated query to English: "médecin mèdicament".

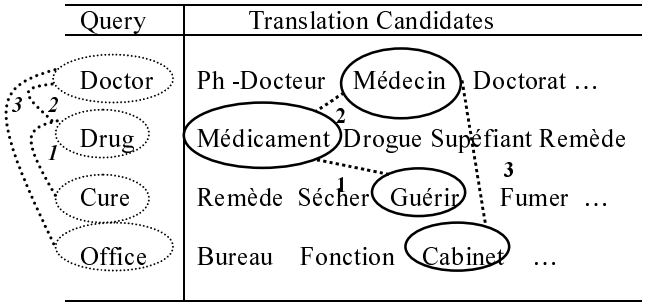


Figure 3: N-Term Disambiguation *(First Method)*: Ranking Source Query Terms and Disambiguation of Target Translations

Highest co-occurrence tendencies related to pairs of source query terms are as follows: (drug, cure), (doctor, drug), (doctor, office), (doctor, cure)…
Source French query:" doctor drug cure office". Translated query to English: "médecin mèdicament guérir cabinet".

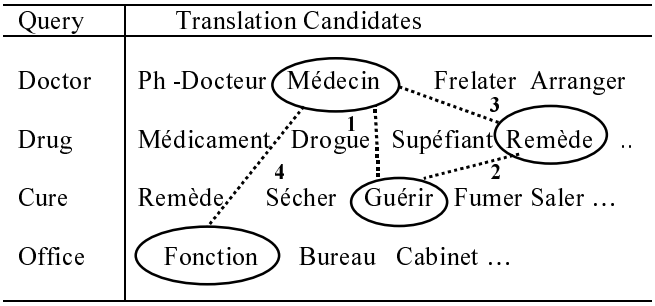


Figure 4: N-Term Disambiguation *(Second Method)*: Ranking and Disambiguation of Target Translations

Highest co-occurrence tendencies related to target translation candidates are as follows: (médecin, guérir), (guérir, remède), (remède, médecin) (médecin, fonction)…
Source French query: "doctor drug cure office". Translated query to English: "médecin remède guérir fonction".

translated queries (after translation) and the organization of source queries (before translation), using the co-occurrence tendency and the following estimations: Log-Likelihood Ratio (LLR) and Mutual Information (MI). LLR was used for *Bi_DIS*, disambiguation of consecutive pairs of source terms, without any ranking or selection (Sadat, 2001), for *LLR_DIS.bef*, the result of the first proposed disambiguation method (ranking source query terms, translation and disambiguation of target translations) and *LLR_DIS.aft*, the result of the second proposed disambiguation method (ranking and selecting target translation). In addition, MI estimation was applied to *MI_DIS.bef* and *MI_DIS.aft*, for the first and second proposed disambiguation methods. Query expansion was completed by the following methods: *Feed.bef_LLR*, which represents the result of adding a number of terms to the original queries and then performing a translation and disambiguation via *LLR_DIS.bef. Feed.aft*, is the result of query translation, disambiguation via *LLR_DIS.bef* method and then expansion. Finally, *Feed.bef_aft*, is the result of combined query expansion both before and after the translation and disambiguation via *LLR_DIS.bef.* In addition, we tested a query expansion before and after the disambiguation method MI_DIS.bef, together with the following methods: Feed.bef_MI, Feed.aft_MI and Feed.bef_aft_MI. Results and performance of these methods are described in Table 1. Figures 5 and 6 show the query translation/disambiguation using LLR and MI. Figures 7 and 8 show the query expansion for different combinations and estimations for the co-occurrence tendency (LLR or MI).

5.3. Discussion

The first column of Table 1 indicates the method. The second column indicates the number of retrieved relevant documents, and the third column indicates the precision averaged at point 0.10 on the Recall/Precision curve. The fourth column is the average precision, which is used as a basis for the evaluation. The fifth column is the R-precision and the sixth column represents the difference in term of average precision of the monolingual counterpart. Compared to the retrieval using original queries (English or French), *All_Tr* and *No_DIS* showed no improvement in term of precision, recall or average precision, whereas the simple two-term disambiguation *Bi_DIS* (disambiguation of consecutive pairs of source query terms) has increased the recall, precision and average precision by +1.71% compared to the simple dictionary translation without any disambiguation. On the other hand, the first proposed disambiguation method (ranking and selecting target translations) *LLR_DIS.aft,* showed a potential precision enhancement, 0.5012 at 0.10 and 90.82% average precision; however, recall was not improved (4131 relevant documents retrieved). The best performance for the disambiguation process was achieved by the second proposed disambiguation method (ranking source query terms and selecting target translations) *LLR_DIS.bef*, in average precision, precision and recall. The average precision was 101.51% of the monolingual counterpart, precision was 0.5144 at 0.10 and 436 relevant documents were retrieved. This suggests that ranking and selecting pairs for source query terms, is very helpful in the disambiguation process to select best target translations, especially for long queries of at least three terms. Results based on mutual information were less efficient compared to those using log-likelihood ratio. However, ranking source query terms before the translation and disambiguation resulted in an improvement in average precision, 100.91% of the monolingual counterpart. Although, query expansion before translation via *Feed.bef_LLR*/*Feed.bef_MI*, gave an improvement in average precision compared to the non-disambiguation method *No_DIS*, a slight drop in precision (0.4507/0.4394) and recall (413/405 relevant retrieved documents) was observed compared to *LLR_DIS.bef* or *MI_DIS.bef.* However, *Feed.aft_LLR/Feed.aft_MI* showed an improvement in average precision, 101.33%/101.25% compared to the monolingual counterpart and improved the precision (0.5153/0.5133 at 0.10) and the recall (433/ 430 retrieved relevant documents). Combined feedbacks both before and after translation yielded the best result, with an improvement in precision (0.5242 at 0.10), recall (434 retrieved relevant documents) and average precision, 102.89% of the monolingual counterpart when using LLR estimation. A disambiguation using MI for co-occurrence tendency yielded a good result, 103.53% of the monolingual counterpart for average precision. These results suggest that combined query expansion both before and after the proposed translation/disambiguation method improves the effectiveness of an information retrieval, when using a co-occurrence tendency based on MI or LLR.

	Rel Docs	at 0.10	A. Prec	R. Prec	% Mono
Mono_Fr (origin)	434	0.4178	0.2629	0.2925	100
Mono_En g (origin)	**433**	**0.4437**	**0.262**	**0.2663**	**100**
All_Tr	406	0.4285	0.2160	0.2573	82.19
No_DIS	429	0.4129	0.2214	0.2431	84.24
Bi_DIS	418	0.4115	0.2259	0.2769	85.95
LLR_DIS. Aft	431	0.5012	0.2387	0.2813	90.82
LLR_DIS. Bef	**434**	**0.5144**	**0.2679**	**0.3118**	**101.94**
MI_DIS.A ft	414	0.4507	0.2325	0.2556	88.47
MI_DIS.B ef	**429**	**0.5125**	**0.2652**	**0.3116**	**100.91**
Feed.bef_ LLR	413	0.4507	0.2309	0.2593	87.86
Feed.aft_ LLR	433	0.5153	0.2663	0.3165	101.33
Feed.bef_ aft_LLR	**436**	**0.5242**	**0.2704**	**0.3201**	**102.89**
Feed.bef_ MI	405	0.4394	0.2264	0.2521	86.14
Feed.aft MI	430	0.5133	0.2661	0.3074	101.25
Feed.bef aft_MI	**430**	**0.5160**	**0.2721**	**0.3077**	**103.53**

Table 1: Evaluations of the Translation, Disambiguation and Expansion Methods *(Different combinations with LLR and MI co-occurrence frequencies)*

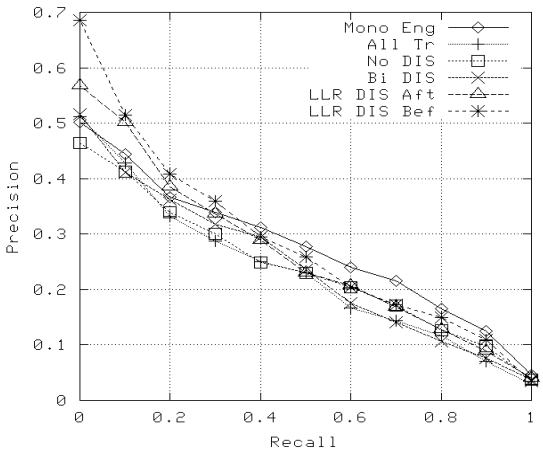


Figure 5: Recall/Precision Curves for the Query Translation/Disambiguation using LLR estimation

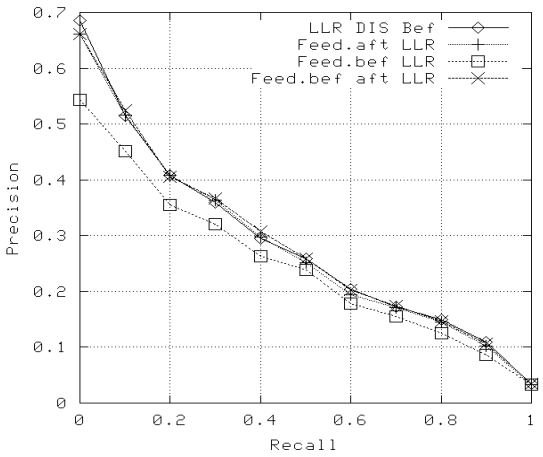


Figure 7: Recall/Precision Curves for the Query Expansion before and after the Translation/ Disambiguation using LLR estimation

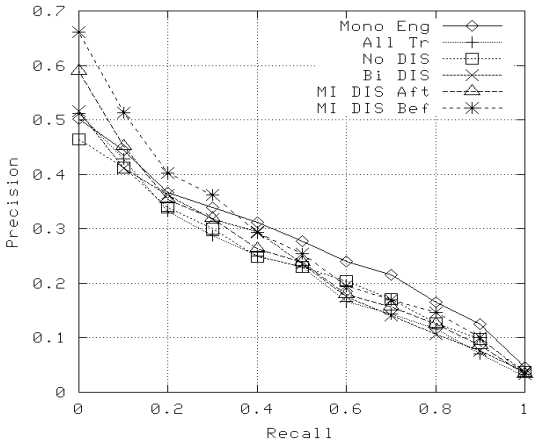


Figure 6: Recall/Precision Curves for the Query Translation/Disambiguation using MI estimation

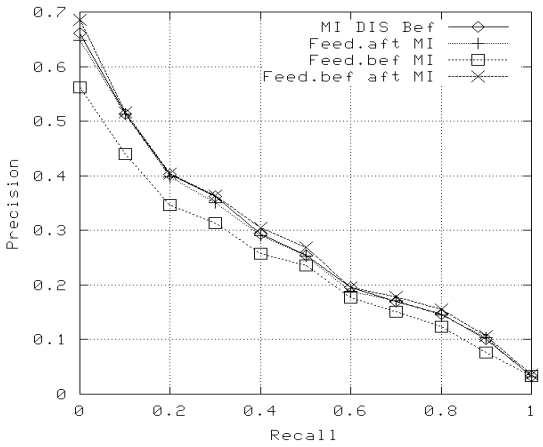


Figure 8: Recall/Precision Curves for the Query Expansion before and after the Translation/ Disambiguation using MI estimation

Thus, techniques of primary importance to this successful method can be summarized as follows:

- A statistical disambiguation method based on the co-occurrence tendency was applied first prior to translation, in order to eliminate misleading pairs of terms for translation and disambiguation. Then after translation, the statistical disambiguation method was applied in order to avoid incorrect sense disambiguation and to select best target translations.
- Ranking and careful selection are fundamental to the success of the query translation, when using statistical disambiguation methods.
- A combined statistical disambiguation method before and after translation provides a valuable resource for query translation and thus information retrieval,
- Log-Likelihood Ratio was found to be more efficient for query disambiguation than Mutual Information,
- A co-occurrence frequency to select an expansion term was evaluated using all terms of the original query, rather than using just one query term.
- Each type of query expansion has different characteristics and therefore combining various types of query expansion could provide a valuable resource for use in query expansion. This technique offered the greatest performance in average precision.
- These results showed that CLIR could outperform the monolingual retrieval. The intuition of combining different methods for query disambiguation and expansion, before and after translation, has confirmed that monolingual performance is not necessarily the upper bound for CLIR performance (Gao et al., 2001). One reason is that those methods have completed each other and that the proposed query disambiguation had a positive effect during the translation and thus retrieval. Combination to query expansion had an effect on the translation as well, because related words could be added.

The proposed combined disambiguation method prior to and after translation, was based on a selection of one target translation in order to retrieve documents. Setting a threshold in order to select more than one target translation is possible using weighting scheme for the selected target translations in order to eliminate misleading terms and construct an optimal query to retrieve documents.

6. Word Sense Disambiguation (WSD)

Word sense ambiguity is a pervasive characteristic of natural language and information retrieval. It is considered as one of the major causes of poor performance in Information Retrieval systems. We believe that a relationship between disambiguation, word sense ambiguity and IR, exists (Sanderson, 1994). Our proposed disambiguation method makes use of statistics data based on co-occurrence between words, which can be extracted from large language corpora. The motivation for this type of approach is the assumption that the used data will provide enough information to resolve most of word sense ambiguities encountered in practical applications. The acquisition of statistical data relies on the availability of training corpora, which is easier to acquire than parallel or aligned corpora. This approach could be well incorporated into Word Sense Disambiguation (WSD) when using dictionary-based translation. Moreover, it is easy to implement and cost effective. We believe that resolving word senses is worthwhile and could have a great impact on the recall and precision, especially, when training corpora are related to particular or different subject areas (Krovetz and Croft, 1992).

7. Conclusion

Dictionary-based method is attractive for several reasons. This method is cost effective and easy to perform, resources are readily available and performance is similar to that of other Cross-Language Information Retrieval methods. Ambiguity arising from failure to translate queries is largely responsible for large drops in effectiveness below monolingual performance (Ballesteros and Croft, 1997). The proposed disambiguation approach of using statistical information from language corpora to overcome limitation of simple word-by-word dictionary-based translation has proved its effectiveness, in the context of information retrieval. A co-occurrence tendency based on a log-likelihood ratio has showed to be more efficient than the one based on mutual information. The combination of query expansion techniques, both before and after translation through relevance feedback improves the effectiveness of simple word-by-word dictionary translation. We believe that the proposed disambiguation and expansion methods will be useful for simple and efficient retrieval of information across languages.

Ongoing research includes a search for additional methods that may be used to improve the effectiveness of information retrieval. Such methods may include the combination of different resources and techniques for optimal query expansion across languages. In addition, thesauri and relevance feedbacks will be studied in greater depth. A good word sense disambiguation model will incorporate several types of data source that complete each other, such as a part-of-speech tagger into statistical models. Finally, an approach to learning from documents categorization and classification in order to extract relevant expansion terms will be examined in the future.

Acknowledgments

The present study is supported in part by the Ministry of Education, Culture, Sports, Science and Technology of Japan, under grants 11480088, 12680417 and 12208032, and by the CREST program of the JST Corporation (Japan Science and Technology). We would like to thank Dr Claude de Loupy and all reviewers for their helpful comments on the earlier version of this paper.

References

Ballesteros, L. and Croft, W. B. 1998. Resolving Ambiguity for Cross-Language Retrieval. In proceedings of the 21st ACM SIGIR Conference.P:64-71.

Church, K. W. and Hanks, P. 1990. Word association Norms, Mutual Information and Lexicography. Computational Linguistics, Vol 16 No1. P: 22-29.

Dunning, T. 1993. Accurate methods for the statistics of surprise and coincidence. Computational linguistics, Vol.19.,No.1. P: 61-74.

Gale, W. A. and Church, K. 1991. Identifying word correspondences in parallel texts. In proceedings of the 4th DARPA Speech and Natural Language Workshop. P: 152-157.

Gao, J., Nie,J.Y., Xun, E., Zhang, J., Zhou, M., Huang, C. 2001. Improving query translation for Cross-Language Information Retrieval using statistical models. In proceedings of the 24st ACM SIGIR Conference. P: 96-104.

Hull, D. and Grefenstette, G. 1996. Querying across languages. A dictionary-based approach to Multilingual Information Retrieval. In proceedings of the 19th ACM SIGIR Conference. P:49-57.

Hull, D. 1998. A weighted boolean model for Cross-Language text Retrieval. In G. Grefenstette editor: Cross-Language Information Retrieval, chapter 10. Kluwer Academic Publishers.

Hutchins, J. and Sommers, J. 1992. Introduction to Machine Translation. Academic Press.

Krovetz, R. and Croft, W. 1992. Lexical ambiguity and information retrieval. ACM Transactions on Information Systems, 10 (2). P: 115-141.

Loupy, C., Bellot, P., El-Beze, M. and Marteau, P.-F. 1998. Query expansion and classification of retrieved documents. In Proceedings of TREC-7. NIST Special Publication.

Maeda, A., Sadat, F., Yoshikawa, M. and Uemura, S. 2000. Query term disambiguation for Web Cross-Language Information Retrieval using a search engine. In Proceedings of the 5th International Workshop on Information Retrieval with Asian Languages. P: 25-32.

Oard, D.W. 1997. Alternative approaches for Cross-Language Information Retrieval. In Working notes of the AAAI Symposium on Cross-Language Text and Speech Retrieval. Stanford University, USA. http://www.glue.umd.edu/~oard/research.html

Sadat, F., Maeda, A., Yoshikawa, M. and Uemura, S. 2001. Query expansion techniques for the CLEF bilingual track. In Working Notes for the CLEF 2001 Workshop. P: 99-104.

Sanderson, M. 1994. Word Sense Disambiguation and Information Retrieval. ACM Special Interest Group on Information Retrieval. P: 142-151.

Yamabana, K., Muraki, K., Doi, S. and Kamei, S. 1996. A language conversion Front-End for Cross-Linguistic Information Retrieval. In Proceedings of SIGIR Workshop on CLIR, Zurich, Switzerland. P: 34-39.

Vossen, P. EuroWordNet. 1998. A Multilingual database with lexical semantic networks. Kluwer Academic Publishers.

Semantic enrichment for information extraction using word sense disambiguation

Bernard Jacquemin, Caroline Brun and Claude Roux

Xerox Research Centre Europe
6, chemin de Maupertuis, 38 240 Meylan, France
{Bernard.Jacquemin,Caroline.Brun,Claude.Roux}@xrce.xerox.com

Abstract

External linguistic resources have been used for a very long time in information extraction. These methods enrich a document with data that are semantically equivalent, in order to improve recall. For instance, some of these methods use synonym dictionaries. These dictionaries enrich a sentence with words that have a similar meaning. However, these methods present some serious drawbacks, since words are usually synonyms only in restricted contexts. The method we propose here consists of using word sense disambiguation rules (WSD) to restrict the selection of synonyms to only these that match a specific syntactico-semantic context. We show how WSD rules are built and how information extraction techniques can benefit from the application of these rules.

1. Introduction

In today's world, the society of communications is gaining in importance every day. The amount of electronic documents – mainly by Internet, but not only – grows more and more. With this increase, no one is able to read, classify and structure those documents so that the requested information can be reached when it is needed. Therefore we need tools that reach a shallow understanding of the content of these texts to help us to select the requested data.

The process of understanding a document consists in identifying the concepts of the document that correspond to requested information. This operation can be performed with linguistic methods that permit the extraction of various components related to the data that are requested.

Since the beginning of the '90s, several research projects in information extraction from electronic text have been using linguistic tools and resources to identify relevant elements for a request. The first ones, based on domain-specific extraction patterns, use hand-crafted pattern dictionaries (CIRCUS (Lehnert, 1990)). But systems were quickly designed to build extraction pattern dictionaries automatically. Among these systems, AutoSlog (Riloff, 1993; Riloff and Lorenzen, 1999) builds extraction pattern dictionaries for CIRCUS. CRYSTAL (Soderland et al., 1995) creates extraction patterns lists for BADGER, the successor of CIRCUS. These learners use hand-tagged specific corpora to identify structures containing the relevant information. The syntactic structure used by CRYSTAL is more subtle than the one used by AutoSlog. CRYSTAL is able to make the most of semantic classes. WHISK (Soderland, 1999) is one of the most recent information extraction system. WHISK has been designed to learn which data to extract from structured, semi-structured and free text [1]. A parser and a semantic tagger have been implemented for free text. This system is the only one to process all of these three categories of text.

These methodologies need domain-specific pattern dictionaries that must be built for each different kind of information. However, none of these methods can be directly applied to generic information. Thus we decide to bypass these two obstacles: our approach is based on the utilization of an existing electronic dictionary, in order to expand the data in a document to equivalent forms extracted from that dictionary.

Our method deals with the identification of semantic contents in documents through a lexical, syntactic and semantic analysis. It then becomes possible to enrich words and multi-word expressions in a document with synonyms, synonymous expressions, semantic information etc. extracted from the dictionary.

2. Problems and Prospects

As for a lot of methodologies developed for natural language processing, the results of a method of information extraction are evaluated by two measures: precision and recall. Precision is the ratio of correctly extracted items to the number of items both correctly and erroneously extracted from the text; noise is the ratio of the faulty extracted items to all the achieved extractions. Recall is the ratio of correctly extracted items to the number of items actually present in the text. The problem consists in improving both precision and recall.

2.1. Recall improvement

A usual technique to improve the recall consists of enriching a text with a list of synonyms or near-synonyms for each word of that text. For example, all the synonyms of "climb" would be added to the document, even though some of those meanings have a remote semantic connection to the text. By this kind of enrichment, all the ways to express the same token (but not the same meaning) are taken into account.

This type of enrichment can be extended to synonymous expressions with a robust parser: syntactic dependencies and their arguments (the tokens belonging to the selected expression) are enlarged to dependencies that are generated out of the corresponding synonymous expressions.

[1] We use the term "structured text" to refer to what the database community calls semi-structured text; "semi-structured text" is ungrammatical and often telegraphic text that does not follow any rigid format; "free text" is simply grammatical text (Soderland, 1999).

The recall is usually optimised to the detriment of the precision with those techniques, since most words within a set of synonyms are themselves polysemous and are seldom equivalent for each of their meanings. Thus, a simply adding of all those polysemous synonyms in a document introduces meaning inconsistencies. Noise may stem from these inconsistencies.

2.2. Reduction of noise – Precision improvement

We notice that improving the recall using synonyms may often increase the noise. Although identified in the domain of IE, this problem is not yet solved and it has a negative influence on the system effectiveness. Our purpose is to use the linguistic context of the polysemous tokens to identify their meanings and select contextual synonyms or synonymous expressions. This approach should improve the precision in comparison with adding all the synonyms.

Sentences in the text:

La température **grimpe**.
(The temperature is climbing.)

Corresponding set of synonyms:

escalader	*monter*
(to climb)	*(to go up)*
sauter	*augmenter*
(to jump)	*(to increase)*
se hisser sur	
(to heave oneself up onto)	

Sentences resulting from the enrichment:

La temperature escalade.
La temperature ***monte****.*
La temperature saute.
La temperature ***augmente****.*
La temperature se hisse sur (???).

Figure 1: Enrichment by a list of synonyms.

For example, the dictionary [2] entry for the word *grimper* contains a set of 5 synonyms. If we use these synonyms to enrich the original text, we obtain five variations of the original sentence. Only the second and the fourth of the enriching variations are accurate in this context. The meteorological context associated with the word *température* in the dictionary should correctly discriminate the synonyms in this context: in the dictionary, each synonym of a lemma is associated with a meaning of this lemma and with the typical linguistic context of the lemma in this sense.

Consequently, we decided to use the linguistic context of the words that can be enriched to discriminate which synonyms should be used and which should not. The synonyms are stored in the dictionary according to the sense of each lemma. So, the task amounts to performing a lexical semantic disambiguation of the text and using synonymous expressions in the selected meanings to enrich the document.

3. Enrichment method by WSD

3.1. Our experience in WSD

We previously have developed a range of tools and techniques to perform Word Sense Disambiguation (WSD), for French and English. The basic idea is to use a dictionary as a tagged corpus in order to extract semantic disambiguation rules, (Brun et al., 2002; Brun, 2000; Brun and Segond, 2001; Dini et al., 1998; Dini et al., 2000). Since electronic dictionaries exist for many languages and they encode fine-grained reliable sense distinctions, be they monolingual or bilingual, we decided to take advantage of this detailed information in order to extract a semantic disambiguation rule database [3]. The disambiguation rules associate each word with a sense number taking the context into account. For bilingual dictionaries the sense number is associated with a translation, for monolingual dictionaries with a definition. WSD is therefore performed according to sense distinctions of a given dictionary. The linguistic rules have been created using functional dependencies provided by an incremental shallow parser (IFSP, (Ait-Mokhtar and Chanod, 1997)), semantic tags from an ontology (45 classes from WordNet (Feldbaum, 1998) for English) as well as information encoded in SGML tags of dictionaries. This method comprises two stages, rule extraction and rule application.

- Rule extraction process: for each entry of the dictionary, and then for each sense of the entry, examples are parsed with the IFSP shallow parser. The shallow parsing task includes tokenization, morphological analysis, tagging, chunking, extraction of functional dependencies, such as subject and object (SUBJ(X, Y), DOBJ (X, Y)), etc. For instance, parsing the dictionary example attached to one particular sense $\mathbf{S}_i$ of *drift* :

  1)The country is drifting towards recession.

  Gives as output the following chunks and dependencies :

  [SC [NP The country NP]/SUBJ :v is drifting SC] [PP towards recession PP] SUBJ(country, drift) VMOD-OBJ(drift, towards, recession)

  Using both the output of the shallow parser and the sense numbering from the dictionary we extract the following semantic disambiguation rule: When the ambiguous word "drift" has *country* as subject and/or *toward recession* as modifier, it can be disambiguated with its sense $\mathbf{S}_i$. We repeat this process as all dictionary example phrases in order to extract the word level rules, so called because they match the lexical context.

[2] The dictionary we use is a French electronic one (Dubois and Dubois-Charlier, 1997). We will give a more detailed information about it later.

[3] The English dictionary contained 39 755 entries and 74 858 senses, ie a polysemy of 1.88; the French dictionary contained 38 944 entries and 69 432 senses, ie a polysemy of 1.78

Finally, for each rule already built, we use semantic classes from an ontology in order to generalize the scope of the rules. In the above example the subject "country" is replaced in the semantic disambiguation rule by its ambiguity class. We call ambiguity class of a word, the set of WordNet tags associated with it. Each word level rule generates an associated class level rule, so called because it matches the semantic context: when the ambiguous word "drift" has a word belonging to the WordNet ambiguity class *noun.location* and *noun.group* as subject and/or a word belonging to the WordNet ambiguity class *noun.shape*, *noun.act*, and *noun.state* as modifier, it disambiguates with its sense $\mathbf{S}_i$. Once all entries are processed, we can use the disambiguation rule database to disambiguates new unseen texts. For French, semantic classes (69 distinctive characteristics) provided by the *AlethDic* dictionary (Gsi-Erli, 1994) have been used with the same methodology.

- Rule application process: The rule applier matches rules of the semantic database against new unseen input text using a preference strategy in order to disambiguate words on the fly. Suppose we want to disambiguate the word drift, in the sentence:

  2) In November 1938, after Kristallnacht, the world drifted towards military conflict.

  The dependencies extracted by the shallow parser, which might lead to a disambiguation, i.e., which involve *drift*, are:

  SUBJ(world, drift)
  VMODOBJ(drift, towards, conflict)

  The next step tries to match these dependencies with one or more rules in the semantic disambiguation database. First, the system tries to match lexical rules, which are more precise. If there is no match, then the system tries the semantic rules, using a distance calculus between rules and semantic context of the word in the text [4]. In this particular case, the two rules previously extracted match the semantic context of *drift*, because *world* and *country* shares semantic classes according to WordNet, as well as *conflict* and *recession*.

The methodology attempts to avoid the data acquisition bottleneck observed in WSD techniques. Thanks to this methodology, we built all-words (within the limits of the used dictionary) unsupervised Word Sense Disambiguator for French (precision: 65%, recall: 35%) and English (precision: 79%, recall: 34%).

3.2. Xerox Incremental Parser (XIP)

IFSP, which was used in the first experiments on semantic disambiguation at Xerox, has been implemented with transducers. Transducers proved to be an interesting formalism to implement quickly an efficient dependency parser, as long as syntactic rules would only be based on POS. The difficulty of using more refined information, such as syntactic features, drove us to implement a specific platform that would keep the same strategies of parsing as in IFSP, but would no longer rely on transducers.

This new platform (Ait-Mokhtar et al., 2001; Roux, 1999) comprises different sorts of rules that chunk and extract dependencies from a sequence of linguistics tokens, which is usually but not necessarily a sentence. The grammar of French that has been developed computes a large number of dependencies such as *Subject, Object, Oblique, NN* etc. These dependencies are used in specific rules, the disambiguation rules, to detect the syntactic and semantic information surrounding a given word in order to yield a list of words that are synonyms according to that context. Thus, a disambiguation rule manipulates together a list of semantic features originating from dictionaries, and a list of dependencies that have been computed so far. The result is a list of contextual synonyms.

If (Dependency$_0$(t, t^0) & ...& Dependency$_n$(t,t^k) & ... attribute$_p$(t^j)=v^u)

synonym(t) = s^0,...,s^n.

where

t^0,...,t^n is a list of token

s^0,...,s^n a list of synonyms.

Example:

- La température grimpe.
  (the temperature is climbing)
- La température augmente.
  (the temperature is rising)
- L'alpiniste grimpe le mont Ventoux.
  (the alpinist climbs the mount Ventoux)
- ???L'alpiniste augmente le mont Ventoux.
  (???the alpinist raises the mount Ventoux)

Figure 2: Application of a disambiguation rule for enrichment.

The contextual synonymy between *grimper* and *augmenter* can be defined with the following rule. The feature *MTO* is one of the semantic features that are associated with the entries of the Dubois dictionary. This feature is associated with each word that is connected to meteorology, such as *chaleur, froid, température* (heat, cold, température).

if (Subject(*grimper*, X) AND feature(X, *domain*)=MTO) synonym(*grimper*) = *augmenter*.

This rule applies on the above first example, *La température grimpe*, but fails to apply on the third sentence, *L'alpiniste grimpe le mont Ventoux*, since the subject does not bear the MTO feature.

[4] The first parameter of this metric is the intersection of the rule classes and the context classes; the second one is the union of the rule classes and the context classes. Distance equals the ratio of intersection to union.

3.3. Which WSD for which enrichment?

3.3.1. A very rich dictionary information

The new robust parser offers a flexible formalism and the possibility to handle semantic or other features. In addition to this parser, the semantic disambiguation now uses a monolingual French dictionary (Dubois and Dubois-Charlier, 1997). This dictionary contains many kind of information in the lexical field as well as in the syntactic or the semantic one. From the 115 229 entries of this dictionary, we can only use the 38 965 ones that are covered by the morphological analyser. These entries represent 68 588 senses, ie a polysemy of 1.76.

We build lexico-syntactic WSD rules using the methodology presented above (cf. section 3.1.): examples of the dictionary are parsed; extracted syntactic relations and their arguments are used to create the rules. We also make the most of the domain indication (171 different domains) to generalize the example rules (see later for details) – as previously done using WordNet for the English WSD and by AlethDic for the French one (Brun et al., 2002).

We use the specificity of the dictionary to improve the disambiguation task as far as possible in order to maximize the enrichment of the documents. The information of this dictionary is divided into several fields: domain, example, morphological variations, derived or root words, synonyms, POS, meaning, estimate of use frequency in the common language; in the verbal part of the dictionary only, syntactico-semantic class and subcategorization patterns of the arguments of the verb. Resulting WSD rules are spread over three levels reflecting the abstraction register of the dictionary fields.

3.3.2. Disambiguation rules at various levels

We build a disambiguation rule database at three levels: rules at word level (23 986), rules at domain level (22 790) and rules at syntactico-semantic level (40 736).

Word level rules use lexical information from the examples. They correspond to the basic rules in the previous system, which use constraints on words and syntactic relations. These dependencies are extracted from the illustrative examples from the dictionary.

L'avion de la société **décrit** un large cercle avant de (…)
*(The company's plane **describes** a wide circle before (…))*
SUBJECT(décrire,avion)
OBJECT(décrire,cercle)

Example in the dictionary for the entry "décrire":
L'avion décrit un cercle.
(The plane describes a circle.)
SUBJECT(décrire,avion)
OBJECT(décrire,cercle)

Figure 3: WSD at word level.

Rules at domain level are generalized from word level rules: instead of using the words of the examples as arguments of the syntactic relations in the rules, we replace them by the domains they belong to. These rules correspond to the class level rules in the previous system, but an improvement in comparison with them is that in some cases, we can discriminate the right domain if the argument is polysemous. This is mainly due to the internal consistency of the dictionary that enables the correspondences of domain across different arguments of a dependency. The consistency should help to reduce the noise.

L'escadrille décrit son approche vers l'aéroport où (…)
(The squadron describes its approach to the airport where (…))
SUBJECT(décrire,escadrille[dom:AER])
OBJECT(décrire,approche[dom:LOC])

Example in the dictionary for the entry "décrire":
L'avion décrit un cercle.
(The plane describes a circle.)
SUBJECT(décrire,avion[dom:AER])
OBJECT(décrire,cercle[dom:LOC])

Figure 4: WSD at domain level.

We don't rule out the possibility of using other lexico-semantic resources to generalize or expand this kind of rules, as we did previously using French EuroWordNet or AlethDic. These lexicons present the advantage of a hierarchical structure that doesn't exist for the domain field in the Dubois dictionary. Nevertheless, we will encounter the problem of the mapping of the various resources used by the system to avoid inconsistencies between them, as shown in (Ide and Véronis, 1990; **?**; Brun et al., 2002).

The third level of the rules currently in use in the semantic disambiguator is the syntactico-semantic one. The abstraction level of these rules is even higher than in the domain level. They are built from a syntactic pattern of subcategorization that indicates the typical syntactic construction of the current entry in its current meaning. Although the distinction between the arguments is very general – they are differentiated from human, animal and inanimate – our examination of the verbal dictionary indicates that, for 30% of the polysemous entries, this kind of rules is sufficient to choose the appropriate meaning.

3.4. Enrichment at various levels

WSD is not an end in itself. In our system, it is a means to select appropriate information in the dictionary to enrich a document. The quality and the variety of this enrichment vary according to the quality and the richness of the information in the dictionary. The variety of information allows several kind of enrichment.

For the specific task of information extraction, an index of the documents whose information is likely to be extracted is built. It allows the classification of all the linguistic realities extracted from text analysis. These realities are listed according to the XIP-formalism: syntactic relations,

L'escadrille décrit son approche vers l'aéroport where (...)
(The squadron describes its approach to the airport where(...))
SUBJECT(décrire,escadrille[dom:AER])
OBJECT(décrire,approche[dom:LOC])

Subcategorisation for the entry "décrire":
Transitive verb;
Subject inanimate.
SUBJECT(décrire,?[subcat:inanimate]) & OBJECT(décrire,?)

Figure 5: WSD at lexico-semantic level.

arguments, and features attached to the arguments. The enrichment is done inside the index because dependencies can be added without affecting the original document.

3.4.1. Lexical level

Replacing a word by its contextual synonyms is the easiest way to perform enrichment. This method of recall improvement is very common in IE, but in our system, the enrichment is targeted according to the context thanks to the semantic disambiguation. This process often reduces the noise. The enrichment is achieved by copying the dependencies containing the disambiguated word and by replacing this word by one of its synonyms.

La température grimpe.
(The temperature is climbing.)

Original index:
SUBJECT(grimper,température)

Set of targeted synonyms:
monter, augmenter.

Enriched index:
SUBJECT(grimper,température)
SUBJECT(monter,température)
SUBJECT(augmenter,température)

Figure 6: Enrichment at lexical level.

3.4.2. Lexico-syntactic level

The lexico-syntactic level of enrichment is more complex to achieve. The task consists in replacing a word by a multi-word expression (more than 14 000 synonyms are multi-word expressions in our dictionary) or in replacing a multi-word expression by a word, taking into account the words (lexical) and the dependencies between them (syntactic):

- Replacing a word by a multi-word expression (see figure 7):
  - Parse the multi-word expression to obtain dependencies;
  - Match the corresponding dependencies in the text;
  - Instantiate the missing arguments with the text arguments.
- Replacing a multi-word expression by a word:
  - Identify the POS of the word;
  - Select dependencies implying one and only one word of the multi-word expression;
  - Eliminate dependency where this word has a different POS;
  - Replace this word with its synonym in the remaining dependencies.

Le spécialiste a édité un manuscrit très abîmé.
(The specialist published a very damaged manuscript.)

Original index:
SUBJECT(éditer,spécialiste)
OBJECT(éditer,manuscrit)

Targeted synonymous expression:
établir l'édition critique de

Extracted dependencies from the expression:
SUBJECT(établir,?)
OBJECT(établir,édition)
EPITHET(édition,critique)
PP(édition,de,?)

Enriched index:
SUBJECT(éditer,spécialiste)
OBJECT(éditer,manuscrit)
SUBJECT(établir,spécialiste)
OBJECT(établir,édition)
EPITHET(édition,critique)
PP(édition,de,manuscrit)

Figure 7: Enrichment at lexico-syntactic level.

Since our work is based on the Dubois dictionary – whose entries are single words – most of the enrichment is one-to-one word. When a multi-word expression appears in the synonyms list, a single word has to be replaced by a multi-word expression, and the inverse process can be achieved if necessary. The complex case of replacing a multi-word expression by another multi-word expression could arise, but we never encounter this situation. The replacement of a multi-word expression by another is not yet implemented because of the complexity of the process. Nevertheless, the system relies on relations and arguments that are easy to handle, very simple and modular. These characteristics should allow us to bypass the inherent complexity of these structures.

3.4.3. A semantic level example

Syntactico-semantic fields in the dictionary allow a third enrichment level. The syntactico-semantic class structure contains very useful information that makes it possible to link verbs that are semantically related but lexically and syntactically very different. It might be interesting to semantically link *vendre* ("to sell", class D2a) and *acheter* ("to buy", class D2c) even though their respective actors are inverted. For example, *le marchand vend un produit au client* (the trader sells a product to the customer) bears the same meaning as *le client achète un produit au marchand* (the customer buys a product from the trader). The semantic class gives a general meaning of the verb(D2, meaning *donner, obtenir*, to give, to obtain), while the syntactic pattern (a for *vendre*: *fournir qc qn*, to supply so with sth, transitive with a oblique compliment, c for *acheter*: *prendre qc qn*, to take sth to so, transitive with a oblique compliment) yields the semantic realization.

Le papa offre un cadeau à sa fille.
(The father is giving a present to his daughter.)

Original index:
SUBJECT(offrir,papa)
OBJECT(offrir,cadeau)
OBLIQUE(offrir,fille)

offrir 01: D2a (to give sth to sb)
D2a corresponds to D2e (receive, obtain sth from sb).
recevoir 01: D2e

Enriched index:
SUBJECT(offrir,papa)
OBJECT(offrir,cadeau)
OBLIQUE(offrir,fille)
SUBJECT(recevoir,fille)
OBJECT(recevoir,cadeau)
????(recevoir,de,papa)

Figure 8: Enrichment at semantic level.

In a same perspective, a syntactico-semantic class constitutes another synonym set. Since this set is too general and too imprecise, it cannot be used to enrich a document. Still, it can be used as a last resort to enrich the query side when other methods have failed. We will not use this set as enrichment, but only to match a query by the class if the enrichment fails.

4. Evaluation

Though the method presented in this article is based on previous works, the use of other tools and lexical resource may have extended the potential of WSD rules. In particular, it is possible that the number of domains increase precision, and the use of subcategorization patterns may ensure more general rules to increase recall.

The partial evaluation we performed concerns 604 disambiguations in a corpus of 82 sentences from the French newspaper *Le Monde*. Precision in WSD is ratio of correct disambiguations to all disambiguations performed; recall is ratio of correct disambiguations to all possible disambiguations in the corpus. We distinguish the mistakes due to the method and the ones linked to our analysis tools in order to identify what we have to improve in order to increase the performance. These results are promising since both precision and recall are better than in the previous system.

Tokenization mistakes	44	7.28%
Tagging mistakes	19	3.15%
Parsing mistakes	9	1.49%
WSD mistakes	84	13.91%
Precision	**448**	**74.17%**
Recall		**43.61**%

Table 1: WSD method evaluation.

We note some remarks about this evaluation:

1. The lexicon used to perform tokenization has been modified in order to include additional information from the dictionary. We noticed during this evaluation some problems of coverage;

2. For this first prototype, we do not yet establish a strategy for cases in which multiple rules match. If more than one rule can be applied to the context, the sense is randomly chosen among the ones suggested by the matching rules [5];

3. Conversely, we do not yet try a strategy using the domain of disambiguated words as a general context to choose the corresponding meaning of a word to disambigate.

During the evaluation, we also notice that when a result was correct, the suggested synonymous expressions were always correct for the disambiguated word in this context. Our method for an optimized enrichment is validated.

5. Conclusion

In this paper, we present an original method for processing documents, preparing the text for information extraction. The goal of this processing is to expand each concept by the largest list of contextualy synonymous expressions in order to match a request corresponding to this concept.

Therefore, we implement an enrichment methodology applied to words and multi-word expressions. In order to perform the enrichment task, we have decided to use WSD to contextually identify the appropriate meaning of the expressions to expand. Inconsistent enrichment by synonyms is currently known as a major cause of noise in Information Extraction systems. Our strategy lets the system target the enriching synonymous expressions according to the semantic context. Moreover, this enrichment is achieved not

[5] This random choice is only performed for this evaluation and not in a IE perspective, since noise is better than silence in this field.

only with single synonymous words, but also with multi-word expressions that might be more complex than simple synonyms.

The WSD task and the resulting enrichment stage are achieved using syntactic dependencies extracted by a robust parser: the WSD is performed using lexico-semantic rules that indicate the preferred meaning according to the context. The linguistic information extracted from the analysis of the documents is indexed for the IE task. This index also stores additional new dependencies stemming from the enrichment process.

The utilization of a unique, all-purpose dictionary to achieve WSD and enrichment ensures the consistency of the methodology. Nevertheless, the information quality and richness of the dictionary might determine the system effectiveness.

The evaluation validates the quality of our method, which allows a great deal of lexical enrichment with less noise than is introduced by other enrichment methods. We have also indicated some ways our method could be expanded and our analysis tools could be improved. Our next step will be to test the effect of the enrichment in an IE task.

The method is designed to achieve a generic IE task, and the tools and resources are developed to process text data at a lexical level as well as at a syntactic or semantic level.

6. References

Salah Ait-Mokhtar and Jean-Pierre Chanod. 1997. Subject and object dependency extraction using finite-state transducers. In *Workshop on automatic Information Extraction and the Building of Lexical Semantic Resources, ACL*, pages 71–77, Madrid, Spain.

Salah Ait-Mokhtar, Jean-Pierre Chanod, and Claude Roux. 2001. A multi-input dual-entry point dependency parser. In *Proceedings of the International Workshop of Parsing Technology*, Beijing, China. IWPT-01.

Caroline Brun and Frédérique Segond. 2001. Semantic encoding of electronic documents. *International Journal of Corpus Linguistic*, 6:1:79–97.

Caroline Brun, Bernard Jacquemin, and Frédérique Segond. 2002. Exploitation de dictionnaires électroniques pour la désambiguisation sémantique lexicale. *TAL, special issue on Lexiques Sémantiques*, 42:3:to appear.

Caroline Brun. 2000. A client/server architecture for word sense disambiguation.

Luca Dini, Vittorio Di-Tomaso, and Frédérique Segond. 1998. Error driven word sense disambiguation. In *proceedings of COLING/ACL98*, pages 320–324, Montreal, Canada.

Luca Dini, Vittorio Di-Tomaso, and Frédérique Segond. 2000. Ginger II: an example-driven word sense disambiguato. *Computer and the Humanities, special issue on Senseval*, 34:121–129.

Jean Dubois and Françoise Dubois-Charlier. 1997. *Dictionnaire des verbes français*. Larousse, Paris. This dictionary exists in an electronic version and is accompanied by the corresponding electronic Dictionnaire des mots français.

Christiane Feldbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press, (MA).

Gsi-Erli. 1994. *Le dictionnaire AlethDic*. Erli.

Nancy Ide and Jean Véronis. 1990. Mapping dictionaries: A spreading activation approach. In *Proceedings of the 6th Annual Conference of the Centre for the New Oxford English Dictionary*, pages 52–64, Waterloo, Ontario.

Wendy Lehnert. 1990. Symbolic/subsymbolic sentence analysis: Exploiting the best of two worlds. In J. Barnden and J. Pollack, editors, *Advances in Connexionist and Natural Computation Theory*, volume 1, pages 135–164. Ablex Publishers, Norwood, NJ.

Ellen Riloff and Jeffrey Lorenzen. 1999. Extraction-based text categorization: generating domain-specific role relationships automatically. In T. Strzalkowski, editor, *Natural Language Information Retrieval*. Kluwer Academic Publisher.

Ellen Riloff. 1993. Automatically constructing a dictionary for information extraction tasks. In *Proceedings of the Eleventh National Conference on Artificial Intelligence*, pages 811–816. AAAI Press / MIT Press.

Claude Roux. 1999. Phrase-driven parser. In *Proceedings of VEXTAL'99*, Venezia, Italia. VEXTAL'99.

Stephen Soderland, David Fisher, Jonathan Aseltine, and Wendy Lehnert. 1995. Crystal: Inducing a conceptual dictionary. In *Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, pages 811–816. IJCAI-95.

Stephen Soderland. 1999. Learning information extraction rules for semi-structured and free text. *Machine Learning*, 34:233–272.

Word Sense Disambiguation Using Semantic Sets based on WordNet

Ganesh Ramakrishnan

Computer Sc. & Engg.
Indian Institue of Technology
Mumbai - 400076

Pushpak Bhattacharyya

Computer Sc. & Engg.
Indian Institue of Technology
Mumbai - 400076

Abstract

This paper presents an automatic method for resolving the lexical ambiguity of nouns in any free-flowing text. The method exploits the noun taxonomy present in the WordNet and also the relative position of nouns in the given text, to construct semantic sets from the text. The semantic set has been defined as a collection of senses of words in given text that are related through the WordNet. Two different concepts of semantic distance between words have been explored and used for disambiguation. Hand-tagging of text and training are not required by the method presented in this paper. The method has been tested against SemCor, the tagged version of the Brown corpus and compared with previous unsupervised WSD algorithms. The method is supported by good empirical results.

1 Introduction

Any language uses words with multiple meanings. Before Information Retrieval or Semantic analysis of texts, it is essential to determine the true senses of those words. The problem of determining the right sense of words, in a context, is called *Word Sense Disambiguation* (WSD).

The typical approaches to the problem of WSD can be classified into 3 types: (1) *Supervised*, (2) *Unsupervised* and (3) *Cross-Lingual*.

Supervised Methods require resources like semantically annotated corpora to train the WSD system, and lexical resource like WordNet which provides the sense numbers using which the annotations are made. These algorithms, like the ones considered in [1], [2] and [3] use the corpora like Grolier's encyclopedia [1] or private sense-tagged data-sets [2]. However, the semantically annotated corpora are Laborious to construct and expensive, since tagging is done manually or at most semi-automatically.

Unsupervised Methods consider the statistically relevant co-occurance of individual keywords as classes and generate a class based model to predict which will be the most likely class to follow a particular keyword. The class is treated as an equivalent of sense. Unsupervised WSD methods can be further classified into two types, viz. WSD that makes use of the information provided by machine readable dictionaries: this is the case with the work reported by [10], [14], [4], [12] and [11]. And WSD that uses information gathered from raw corpora (unsupervised training methods); [1] and [13] presented unsupervised WSD methods using raw corpora.

From a multilingual point of view, word sense disambiguation is nothing more than determining the appropriate *translation* of a word or lexical item. Thus, translation presupposes word sense disambiguation. Word translation only requires only that the words should be expressing the same meaning. However, it is not necessary to know the exact meaning of the words. See [7] for further details.

2 WordNet

WordNet[9] is an online lexical reference system whose design is inspired by current psycholinguistic theories of human lexical memory. English nouns, verbs, adjectives and adverbs are organized into synonym sets, each representing one underlying lexical concept. Different relations link the synonym sets. WordNet was developed by the Cognitive Science Laboratory at Princeton University.

The WordNet consists of synsets arranged in semantic relationships with one another, through *hypernymy, hyponymy, holonymy, meronymy, synonymy* and *antonymy* relationships. In our discussion, we use WordNet as the only lexical resource and all the *senses* are with respect to the WordNet.

3 Semantic Set

Below is a sample text of 100 words, from the *Brown Corpus*, with some nouns underlined.

In the WordNet sub-graph in figure 2, the relationship between these nouns is shown. The words marked in ellipses are words that actually occur in the text.

The Fulton_County_Grand_Jury said Friday an investigation of Atlanta 's recent primary_election produced no *evidence* that any irregularities took_place. The jury further said in *term* end presentments that the City_Executive_Committee which had over-all charge of the election deserves the praise and thanks of the City_of_Atlanta for the manner in which the election was conducted The September-October *term* jury had been charged by Fulton Superior_Court_Judge_Durwood_Pye to investigate reports of possible irregularities in the hard-fought primary which was won by Mayor-nominate_Ivan_Allen_Jr ..... It recommended that Fulton legislators act to have these laws studied and revised to the end of modernizing and improving them. The grand_jury commented on a number of other *topics* among them the Atlanta and Fulton_County purchasing_departments which it said are well operated and follow generally accepted practices which inure to the best interest of both governments However the jury said it believes these two offices should be combined to achieve greater efficiency and reduce the cost of administration ... Implementation of Georgia 's automobile title *law* was also recommended by the outgoing jury It urged that the next Legislature provide enabling funds and re-set the effective date so_that an orderly implementation of the *law* may be effected. ... This is one of the major items in the Fulton_County general assistance *program* the jury said but the State_Welfare_Department has seen_fit to distribute these funds through the welfare departments of all the counties in the state with the exception of Fulton_County which receives none of_this money The jurors said they realize a proportionate distribution of these funds might disable this *program* in our less populous counties. The jurors said Failure to do this will continue to place a disproportionate *burden* on Fulton taxpayers.

Figure 1: Sample text from SemCor, br-a01 with the word *program* (word number 93) in consideration

The number in the brackets, by the side of the word, is its WordNet sense number. The numbers mentioned in the square brackets are the textual positions. For example, the word *law* appears in textual positions 66 and 72. The arrows going up-down show the *hyponymy* relations. Thus, 2 hyponyms of *sense number 1* of *cognition* are shown.

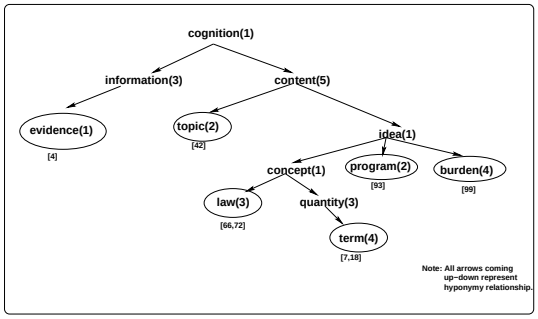


Figure 2: An extract of the WordNet graph, corresponding to the nouns underlined in figure 1

In the same way, one can consider the *holonymy*, *meronymy*, *synonymy* and *antonymy* relationships from the WordNet to capture all the nouns in a given piece of text. Consider the resultant WordNet subgraph. Also, suppose that distances are measured over edges, with every edge of unit distance and the distances are additive. Consider all the words that occur in the graph, within a distance of 4 from the 2^{nd} sense of the word *program*. We call the set of word-senses, within a fixed distance from the chosen synset as the *semantic set* corresponding to that synset. Fig. 3 is an example. The notations and the definitions are given in section 4.

program<80,2,<0,0,0,0,0,0>>, portion<95,1,<0,3,1,0,1,0>>, policy<58,2,<0,2,0,0,0,0>>, term<7,4,<0,3,1,0,0,0>>, topic<42,2,<0,1,2,0,0,0>>, law<66,1,<0,3,1,0,0,0>>, end<8,3,<0,4,0,1,0,0>>, term<18,4,<0,3,1,0,0,0>>, end<8,4,<0,1,0,1,0,1>>, practice<45,5,<0,1,3,0,0,0>>, manner<14,3,<0,4,1,0,0,0>>, law<66,3,<0,2,1,0,0,0>>, end<39,3,<0,4,0,1,0,0>>, end<39,4,<0,1,0,1,0,1>>, law<72,1,<0,3,1,0,0,0>>, burden<99,4,<0,1,1,0,0,0>>, city<59,1,<0,2,3,0,0,0>>, law<72,3,<0,2,1,0,0,0>>, city<31,1,<0,2,3,0,0,0>>, city<56,1,<0,2,3,0,0,0>>, evidence<4,1,<0,2,3,0,0,0>>

Figure 3: Semantic set corresponding to sense number 2 of the word *program*

4 Terminology

We want to find the correct senses of the words in a text T. Let W be a window in T having n nouns,

(w_1, w_2, w_3, ...w_n). For every w_i its s_i senses are σ_{i_1},.....$\sigma_{i_{s_i}}$. Let P_i be the position of w_i in the text.

Semantic Graph

Let G be that minimal sub-graph of the WordNet, which includes all the noun-senses σ_{i_k}, $1 \leq k \leq s_i$ and $1 \leq i \leq n$, from T. We call G, the *Semantic Graph* for the text T.

Let σ_{i_p} and σ_{j_q} be two noun-senses in the sub-graph G. Consider the shortest path from σ_{i_p} to σ_{j_q}. Let $\eta_1(\sigma_{i_p}, \sigma_{j_q})$, $\eta_2(\sigma_{i_p}, \sigma_{j_q})$, $\eta_3(\sigma_{i_p}, \sigma_{j_q})$, $\eta_4(\sigma_{i_p}, \sigma_{j_q})$, $\eta_5(\sigma_{i_p}, \sigma_{j_q})$ and $\eta_6(\sigma_{i_p}, \sigma_{j_{q_6}})$ respectively be the number of *hyponymy*, *hypernymy*, *meronymy*, *holonymy*, *synonymy* and *antonymy* arcs on this path.

Semantic vector

The *semantic vector* between two noun senses σ_{i_p} and σ_{j_q} in the graph G is the sequence

$< \eta_1(\sigma_{i_p}, \sigma_{j_q}),$ ..... $\eta_4(\sigma_{i_p}, \sigma_{j_q}),$ $\eta_5(\sigma_{i_p}, \sigma_{j_q}),$ $\eta_6(\sigma_{i_p}, \sigma_{j_q}) >$,

where the $\eta_i(\sigma_i, \sigma_{j_q})$'s are as given in previous definition. We denote the semantic vector by $N(\sigma_{i_p}, \sigma_{j_q})$.

Semantic distance

The concept of semantic distance has been explored in [15]. Broadly, two concepts of semantic distance have been mentioned there. They are *semantic similarity* and *semantic relatedness*. In this paper, we talk of semantic relatedness, as explored in [16]. But the measures of semantic distance that we adopt are little variants of what has been proposed by [16]. The first measure of the *semantic distance* of a noun-sense σ_{j_q} from σ_{i_p} in G corresponds to the minimum number of arcs that *must* be traversed in order to reach σ_{j_q} from σ_{i_p}.

From the fact that *hypernymy*, *hyponymy* and *meronymy*, *holonymy* are complementary, and that *synonymy* and *antonymy* are *symmetric*, it follows that the semantic distance is *commutative*.

The second measure of semantic distance will be given in section 5.2.

Semantic form

Recall the definition that P_i is the position of w_i in the text. The expression, $w_j < P_j, q, < N(\sigma_{i_p}, \sigma_{j_q} >>$ is called the *semantic form* for σ_{j_q} *with respect to* σ_{i_p}. We will denote it by $F(\sigma_{i_p}, \sigma_{j_q})$.

Semantic set

Consider every noun-sense σ_{j_q} in G, within a maximum semantic distance of R from σ_{i_p}. The collection of all the semantic forms $F(\sigma_{i_p}, \sigma_{j_q})$ is called the *semantic set* S_{i_p} for σ_{i_p}, with radius R. σ_{i_p} is called the *reference noun-sense* for S_{i_p}.

A semantic set S_{i_p} is of the form given in equation 1.

$$S_{i_p} = F(\sigma_{i_p}, \sigma_{i_p}), F(\sigma_{i_p}, \sigma_{i_{1_{p_1}}}) .... F(\sigma_{i_p}, \sigma_{i_{k_{p_k}}}) \quad (1)$$

where k is the length of the semantic set. For word w_i we have s_i semantic sets S_{i_1}, S_{i_2} , ... $S_{i_{s_i}}$. Also, for the word sense σ_{i_p} we define the position vector $\overline{P_{i_p}}$ and $\overline{M_{i_p}}$ as in equations 2 and 3.

$$\overline{P_{i_p}} =< P_{i_1} ... P_{i_k} > \quad (2)$$

$$\overline{M_{i_p}} =< N(\sigma_{i_p}, \sigma_{i_{1_{p_1}}}), ... N(\sigma_{i_p}, \sigma_{i_{k_{p_k}}}) > \quad (3)$$

An Example

Consider again the figure 1 which shows a sample from the text br-a01 of SemCor. The wordsenses of the underlined nouns in the text, form a semantic graph, part of which has been depicted in figure 2. For instance consider the word *program* which has position number 93 in br-a01 and *burden* which has position number 99 in br-a01. IN figure 2, it is shown that sense number 4 of *burden* and sense number 2 of *program* have the same hypernym - the sense number 1 of *idea*. Thus, the semantic distance between *program(2)* and *burden(4)* is 2. The semantic vector from *program(2)* to *burden(4)*, keeping *program(2)* as the reference word is $< \eta_1(\sigma_{93_2}, \sigma_{99_4})$,.. $\eta_6(\sigma_{93_2}, \sigma_{99_4}) >$=<1,1,0,0,0,0>

The distances traversed along the different relation arcs, in the figure 2 from *program(2)* to *burden(4)* are as given in the table 1.

The semantic form $F(\sigma_{93_2}, \sigma_{99_4})$ is given as *burden<99,4,<1,1,0,0,0,0>>*. σ_{99_4} is within a semantic distance of 4 from σ_{93_2}. The collection of all $F(\sigma_{93_2}, \sigma_{j_q}), 1 \leq q \leq s_j \forall$ words $w_j, j \neq i$ in the text T such that, σ_{j_q} is within a semantic distance 4 from σ_{93_2} is called the semantic set for σ_{93_2}, $S(\sigma_{93_2})$. This semantic set is given in figure 3.

5 The Approach

The problem of finding the appropriate sense for w_i can be transformed to the problem of choosing the corresponding appropriate semantic set for w_i. This means we intend to find a measure function $M(S_{i_p}) =$

Table 1: The distance along the different relation arcs, between *program(2)* and *buden(4)* as depicted in 2

Relation	Notation for dist.	Distance
hyponymy	$\eta_1(\sigma_{93_2}, \sigma_{99_4})$	1
hypenymy	$\eta_2(\sigma_{93_2}, \sigma_{99_4})$	1
meronymy	$\eta_3(\sigma_{93_2}, \sigma_{99_4})$	0
holonymy	$\eta_4(\sigma_{93_2}, \sigma_{99_4})$	0
synonymy	$\eta_5(\sigma_{93_2}, \sigma_{99_4})$	0
antonymy	$\eta_6(\sigma_{93_2}, \sigma_{99_4})$	0

m_{i_p} such that $\text{argmax}_{1 \le p \le s_i} M(S_{i_p})$ gives the correct sense for the word w_i.

The idea is that, a word-sense in the text indicates the presence of other word-senses in the piece of text in such a way that **semantically close word senses should also appear textually close**. Therefore, a word-sense in the text is affected by another word-sense in the text in two ways. First is that, *as the semantic distance between them increases, the influence should decrease*. Secondly, *as the textual distance between them increases, the influence should decrease.*

Intuitively, the first factor plays a predominant role in determining the sense of the word under consideration. This follows from the fact that **slight variation in textual position of a word-sense should not influence the sense of the passage as such. But a slight variation in semantic distance should considerably alter the sense of the passage.**

Based on these two observations, we state the hypothesis in section 5.1

5.1 Simple Manhattan measure

Hypothesis

The measure $M(S_{i_p})$ is of the form $M(\overline{P_{i_p}}, \overline{M_{i_p}})$. The contribution of each word σ_{i_j} in the semantic set to the score $M(S_{i_p})$ decreases exponentially its the semantic distance from w_i and decreases inversely with its textual distance from w_i.

Semantic distance (defined in section 4) can be restated as the *Manhattan distance*, $H(\sigma_{i_p}, \sigma_{j_q})$ in equation 4. Note that this measure, in contrast to the measure of semantic distance as given in [16], does not reduce the distance if the path connecting the two concepts changes 'direction too often'. (*e.g* of such a change is when the path connecting the two synsets, changes from say hypernymy to meronymy relation).

$$H(\sigma_{i_p}, \sigma_{j_q}) = \Sigma_{m=1}^{6} \eta_m(\sigma_i, \sigma_{j_q}) \tag{4}$$

According to the hypothesis mentioned above, the expression for the measure function is as given in equation 5.

$$M(\overline{P_{i_p}}, \overline{M_{i_p}}) = \Sigma_{F(\sigma_{i_p}, \sigma_{j_q}) \in S(\sigma_{i_p})} \frac{1}{|P_{j_q} - P_i|} \times e^{-H(\sigma_{i_p}, \sigma_{j_q})} \tag{5}$$

For a word w_i, the appropriate sense number is p and the second most appropriate sense number is $\overline{p}$ iff the conditions given in equations (6) and (7) are satisfied.

$$p = \underset{0 \le j \le s_i}{\text{argmax}} M(S_{i_j}) \tag{6}$$

$$\overline{p} = \underset{0 \le j \le s, j \ne k}{\text{argmax}} M(S_{i_j}) \tag{7}$$

5.2 Eucledian measure

Instead of using the *Manhattan distance*, one can use the *Eucledian distance* . The intuition is given in the figure 4

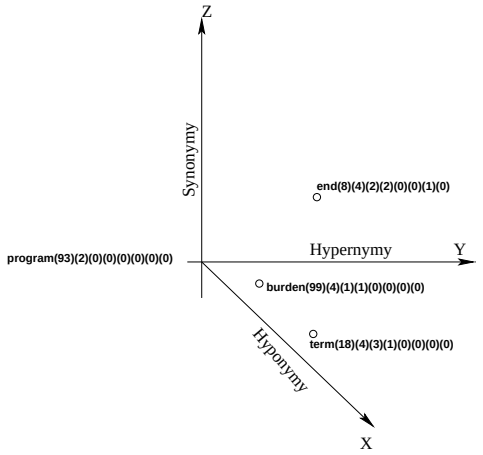


Figure 4: 3-D Graph showing the relative positions of three words with respect the the word *program*

We can look upon the words as being arranged in a six dimensional space, with each space corresponding to one of the 6 relations (*hypernymy etc*). The figure 4 for instance, shows the word-senses *end*(4), *burden*(4) and *term*(4), with respect to the word-sense *program*(2) in 3-D space of *hypernymy, hyponymy* and *synonymy.*

Instead of using the distance measure as in equation 4, we can use the measure $H(\sigma_{i_p}, \sigma_{j_q})$ as in equation 8. Again, this meaure of distance is different from that sugested in [16], because, instead of considering

change of direction along the path, we consider each of the 6 WordNet relations to be along orthogonal directions.

$$H(\sigma_{i_p}, \sigma_{j_q}) = \sqrt{\Sigma_{m=1}^{6}(\eta_m(\sigma_i, \sigma_{j_q})^2)} \quad (8)$$

The appropriate sense for the word w_i can be found as before, using equation 5 and 6. In the measure in equation 8, we give uniform weight-age to all the six relations - *hypernyms* etc. One can instead, give more weight-age to the *hypernymy* and *synonymy* relations as compared to the other relations (say, by taking cubes instead of squares), since, they determine the context of a passage of text, to a greater extent. This gives us the equations 9 and 10 for $H(\sigma_{i_p}, \sigma_{j_q})$.

$$E(\sigma_{i_p}, \sigma_{j_q}) = (\eta_2(\sigma_{i_p}, \sigma_{j_q})^3) + (\eta_5(\sigma_{i_p}, \sigma_{j_q})^3) \quad (9)$$

$$H(\sigma_{i_p}, \sigma_{j_q}) = \sqrt{\Sigma_{m=1,m\neq 2,5}^{6}(\eta_m(\sigma_i, \sigma_{j_q})^2) + E(\sigma_{i_p}, \sigma_{j_q})} \quad (10)$$

Again, one can employ equations 5 and 6 to find the appropriate sense for w_i.

Mutual Reinforcement

We may note that **a word w which has a unique sense in WordNet, helps disambiguate other words related to it**. That is, if word w_j has only one WordNet sense, we would like to give special attention to this information, in all the sets that contain σ_{j_1}. For instance, if the p^{th} semantic set for w_i, i.e S_{i_p} has the word w_j, with w_j having only onse sense in the WordNet, giving more weightage to w_j, sense number 1, will add additional emphasis on the p^{th} sense of w_i.

Moreover, we would like that this effect on σ_{i_p} be reflected on all the sets that contain σ_{i_p}.in turn. To ensure that this happens, we make the following changes to equation 5. Initially, we set the score for each semantic set to 1. Next, within the semantic sets for a word, we normalise the scores. Not that sets corresponding to unambiguous word senses (i.e word senses for the words having just one WordNet sense) will have a score of 1 initially. Then we find the new measure for each semantic set using equations 11 and 12:

$$I(\sigma_{i_p}, \sigma_{j_q}) = M(S_{j_q}) \times \frac{1}{|P_{j_q} - P_i|} \times e^{-H(\sigma_{i_p}, \sigma_{j_q})} \quad (11)$$

$$M(\overline{P_{i_p}}, \overline{M_{i_p}}) = \Sigma_{F(\sigma_{i_p}, \sigma_{j_q}) \in S(\sigma_{i_p})} I(\sigma_{i_p}, \sigma_{j_q}) \quad (12)$$

After updating all set measures for w_i using equation 12, we normalise the measures for the sets correspionding to w_i using equation 13.

$$M(S_{i_p}) = M(\overline{P_{i_p}}, \overline{M_{i_p}}) = \frac{M(S_{i_p})}{\Sigma_{r=1}^{s_i} M(S_{i_r})} \quad (13)$$

Note that in the equation 12 we have scaled the entry for each term σ_{j_q} in the set S_{i_p}, by the measure $M(S_{j_q})$ for the corresponding set S_{j_q}. This means that, if in a particular iteration, sense number q of w_j is found to be more probable than the other senses of w_j, then it's contribution to the scores of other sets is more than the other senses of w_j.

The pseudocode is summarised in figures 5 (INITIALISATION) and 6 (MUTUAL REINFORCEMENT).

1. **INTIALISATION**

2. Incrementally construct semantic chains S_{i_p}, $1 \leq p \leq i_s$, for each of the i_s WordNet senses of σ_i, $1 \leq i \leq n$.

3. for all $1 \leq i \leq n$ do

   (a) for all $1 \leq p \leq s_i$ do

      i. $M(S_{i_p}) = \frac{1}{s_i}$ /* Note that we have combined 2 steps into 1; setting $M(S_{ip})$ to 1 and then normalising */

Figure 5: The INITIALISATION Pseudocode for the method

6 Experiments and results

Experiments were performed over nouns in Brown corpus and checked against SemCor for correctness. As an example case, consider the 93^{rd} noun, *program* in the text in figure 1. It is tagged with sense number 2 in SemCor. Figure 7 shows the 8 semantic sets for the word *program*.

Using equations 4 and 5, we get the scores for the different sets as indicated by the bold number to the right of each set in the figure. *The scores stabilse after around 10 iterations.* We find highest score for the

1. do till the scores stabilise
   (a) **MUTUAL REINFORCEMENT**
   (b) for all $1 \leq i \leq n$ do
      i.
      ii. for all $1 \leq p \leq s_i$ do
         A. $H(\sigma_{i_p}, \sigma_{j_q}) = \Sigma_{m=1}^{6} \eta_m(\sigma_i, \sigma_{j_q})$ /*This could be replaced by the Eucledian measure.*/
         B. $M(\overline{P_{i_p}}, \overline{N_{i_p}}) = \Sigma_{F(\sigma_{i_p}, \sigma_{j_q}) \in S(\sigma_{i_p})} M(S_{j_q}) \times \frac{1}{|P_{j_q} - P_i|} \times e^{-H(\sigma_{i_p}, \sigma_{j_q})}$
   (c) **NORMALISATION**
   (d) for all $1 \leq i \leq n$ do
      i. for all $1 \leq p \leq s_i$ do
         A. $M(S_{i_p}) = M(\overline{P_{i_p}}, \overline{N_{i_p}}) = \frac{M(S_{i_p})}{\Sigma_{r=1}^{s_i} M(S_{ir})}$

Figure 6: The MUTUAL REINFORCEMENT Pseudocode for the method

program<93,1,<0,0,0,0,0,0>>, ... evidence<4,1,<0,2,2,1,0,0>> = 0.312, **0.023**
program<93,2,<0,0,0,0,0,0>>, ... portion<95,1,<0,3,1,0,1,0>> = 1.129, **0.088**
program<93,3,<0,0,0,0,0,0>>, ... election<35,2,<0,3,1,0,1,0>> = 0.144, **0.009**
program<93,4,<0,0,0,0,0,0>>, ... report<24,1,<0,1,1,0,0,0>> = 0.899, **0.611**
program<93,5,<0,0,0,0,0,0>>, ... title<65,1,<0,1,1,1,0,0>> = 0.186, **0.017**
program<93,6,<0,0,0,0,0,0>>, . distribution<91,3,<0,1,3,1,0,0>> = 1.107, **0.053**
program<93,7,<0,0,0,0,0,0>>, ... laws<38,1,<0,2,2,0,0,1>> = 0.629, **0.045**
program<93,8,<0,0,0,0,0,0>>, ... city<56,1,<0,1,3,0,0,1>> = 0.473, **0.022**

Figure 7: Example of 8 semantic sets for the word *program*

second set - thus indicating *sense number 2*. Thus, as per our expectation, the algorithm correctly disambiguated the word *program*. On the other hand, using equations 8 and 5, we get the scores as the underlined number, to the right of each set, in the figure 7. As far as the *Eucledian distance* was concerned, it did not make a big difference, whether we used the measure as suggested in equation 8 or 10. The experiments were carried out on the fist 100 nouns for each of 10 documents from the *Brown corpus*. 2 tests were done - (1) comparing the top ranked sense p and (2) comparing the 2 top ranked senses, p and $\overline{p}$ derived using equation 6. The results for 5 of them are tabulated below.

The average *precision* obtained using the *Eucledian measure* was $3-4\%$ lower than that obtained using

Table 2: Results with top sense for each of 10 *brown corpus* documents

Text	Coverage (%)	Precision (%)	Recall (%)
a01	99	70	69.3
a02	98	69	67.6
a11	96	63	60.5
a12	95	65.0	61.8

Table 3: Results with top 2 senses for each of 10 *brown corpus* documents

Text	Coverage (%)	Precision (%)	Recall (%)
a01	99	83.8	83.0
a02	98	75.5	74.0
a11	96	79.2	76.0
a12	95	74.7	71.0

the *Manhattan measure*. The comparison of our algorithm was done with [4], one of the best known *Unsupervised* WSD algorithms.The comparison was performed on the entire text of *br-a01*. The results were as mentioned in table 4

Table 4: Comparison with [4]

	Aigrre		Our algo	
	precision	recall	precision	recall
br-a01	66.4	58.8	76.9	68.2
br-a02	-	-	70.9	68.8
br-b13	-	-	77.8	75.5
br-c04	-	-	67.3	64.10

The window size $|W| = n$, for all the above tests was chosen as 100. Changing it to 150 produced improvement by $5-7\%$.

7 Conclusions

The algorithm discussed in this paper is unsupervised. Currently, it is designed only for disambiguating nouns. All it needs is WordNet, an extensively used lexical database. It can disambiguate any free running text, provided that the *part of speech tags* are provided. The idea behind the algorithm is theoretically well supported. It has many special features compared to previous unsupervised algorithms. Even though a window of words is used for disambiguation, all the nouns in the window are not considered with equal importance for disambiguating a word in the

text - the importance decreases with increasing distance in the text as well as with increasing *Manhattan* or *Eucledian* distance in the WordNet. Also note that the same word, occurring in different parts of the window is disambiguated in a different way - it considers separately, the multiple occurrences of same word in the same window.

With slight modification, this algorithm can be used for disambiguating *verbs*, *adjectives* in any text. The corresponding *verb* and *adjective* taxonomies in the WordNet can be used for these purposes - in a most similar way.

The algorithm can be improved by choosing a different measure function or choosing different measures of semantic distance, than the two mentioned in this paper. Also, consideration of collocation of words and verb-noun collocations, should give additional clues for disambiguation.

References

[1] David Yarowsky. "Word-sense disambiguation using statistical models of Roget's categories trained on large corpora", In *Proceedings of the* 14^{th} *International Conference on Computational Linguistics (COLING-92)*, pages 454-460, Nantes, France, 1992.

[2] Hwee Tou Ng and Hian Beng Lee, "Integrating Multiple Knowledge Sources to Disambiguate Word Sense: An Exemplar-Based Approach", In *Proceedings of ACL96, 1996.*

[3] Adam Kilgarriff, "Gold Standard Data-sets for Evaluating Word Sense Disambiguation Programs", *Computer Speech and Language 12 (4), Special Issue on Evaluation*, 1998.

[4] Agirre.E and Rigau.G, "Word sense disambiguation using conceptual density", *Proceedings of COLING'96.*

[5] Phil Resnik, "Selectional preference and sense disambiguation", *Proceedings of ANLP-97 Workshop on Tagging Text with Lexical Semantics: Why, What, and How?*, Washington D.C., USA, 1997.

[6] Hinrich Schtze, "Automatic Word Sense Discrimination", *Computational Linguistics*, Volume 24, Number 1, 1998.

[7] Nancy Ide, "Parallel Translations as Sense Discriminators", *Proceedings of SIGLEX99*, Washington D.C, USA, 1999.

[8] Green, S.J., "Automatically Generating Hypertext by Computing Semantic Similarity" *Ph.D. Thesis, University of Toronto*, 1997.

[9] Fellbaum, Christiane, ed., "WordNet: An Electronic Lexical Database" *MIT Press*, May 1998.

[10] Cowie.J, Guthrie.L and Guthrie.J, *L*exical Disambiguation using simulated annealing. Proceedings of the 5th International Conference on Computational Linguistics. COLING-93 (1992), pp157-161.

[11] McRoy.S, *U*sing multiple knowledge sources for Word Sense Disambiguation. Computational Linguistics 18.1 (1992), pages 1-30.

[12] Li.X, Szpakowicz.S and Matwin.M, *A* WordNet based algorithm for word sense disambiguation. Proceedings of the 14^{th} Joint International Conference on Artificial Intelligence.

[13] Resnik.P, *S*electional preference and sense disambiguation. In proceedings of the ACL Singlex Workshop on Tagging Text with Lexical Semantics, Why, What and How? (Washington DC, April 1997).

[14] Miller.G, Chodorow.M, Landes.S, Leacock.C and Thomas.R, *U*sing a semantic concordance for sense indentification. Proceedings of the 37^{th} Annual Meeting of the Association for Computational Linguistics (ACL-99).

[15] Alexander Budanitsky, Graeme Hirst, *S*emantic Distance in WordNet: An experimental, application-oriented evaluation of five measures Proceedings of WordNet and Other Lexical Resources Workshop, Second meeting of the North American Chapter of the Association for Computational Linguistics, Pittsburgh, June 2001.

[16] Graeme Hirst, David St-Onge, *L*exical chains as representations of context for the detection and correction of malapropisms, In: Christiane Fellbaum (editor), WordNet: An electronic lexical database, Cambridge, MA: The MIT Press, 1998, 305–332.

Towards Sense-Disambiguated Association Thesauri

Hiroyuki Kaji and Yasutsugu Morimoto

Central Research Laboratory, Hitachi, Ltd.
1-280 Higashi-Koigakubo, Kokubunji-shi, Tokyo 185-8601, Japan
{kaji, y-morimo}@crl.hitachi.co.jp

Abstract

We developed a method for generating a sense-disambiguated association thesaurus, in which word senses are distinguished according to the related words, from a bilingual comparable corpus. The method aligns pairs of related words translingually by looking up a bilingual dictionary. To overcome both the problem of ambiguity in the translingual alignment of pairs of related words and that of disparity of topical coverage between corpora of different languages, we devised an algorithm for calculating the correlation between the senses of a polysemous word and its related words iteratively according to the set of words related to both the polysemous word and each of the related words. A preliminary experiment using Wall Street Journal and Nihon Keizai Shimbun corpora demonstrated that the method produces a sense-disambiguated association thesaurus successfully. We expect the sense-disambiguated association thesaurus will play essential roles in information retrieval and filtering. Namely, it enables word sense disambiguation of documents and queries as well as effective query expansion. It also functions as an effective user interface for translingual information retrieval.

1 Introduction

An association thesaurus, that is, a collection of pairs of related words, plays an essential role in information retrieval. Query expansion using a corpus-dependent association thesaurus improves recall and/or precision (Jing and Croft 1994; Schuetze and Pedersen 1994; Mandala et al. 1999). Navigation in an association thesaurus allows users to efficiently explore information through a large text corpus even when their information needs are vague (Kaji et al. 2000).

Association thesauri have the advantage of being possibly generated from corpora automatically. However, they have a drawback that they cannot distinguish between the senses of a polysemous word; namely, although each word that is related to a polysemous word is usually relevant to a specific sense of the polysemous word, the association thesauri list all related words regardless of sense. Query expansion using words irrelevant to the sense of user's interest decreases the precision of retrieval. A mixed list of related words relevant to different senses of a polysemous word prevents users from navigating smoothly in the association thesaurus.

In order to solve this problem, we propose a method for generating a sense-disambiguated association thesaurus, in which the senses of a polysemous word are distinguished. More specifically, the words related to a polysemous word are classified according to the sense of the polysemous word to which they are relevant.

2 Approach

The high cost of sense-tagging a corpus prohibits us from collecting pairs of related "senses" directly from a corpus. Accordingly, we adopt a strategy to extract pairs of related "words" from a corpus and then transform each of them to a pair of related senses. This transformation is done through translingual alignment of pairs of related words, as shown in Figure 1. The underlying assumptions are:

(1) The senses of a polysemous word in a language are lexicalized differently in another language (Resnik and Yarowsky 2000).
(2) Translations of words that are related in one language are also related in the other language (Rapp 1995).

According to the first assumption, we define each sense of a polysemous word x of the first language by a synonym set consisting of x itself and one or more of its translations $y_1, y_2, \ldots$ into the second language. The synonym set is similar to that in WordNet (Miller 1990) except that it is bilingual, not monolingual. Examples of some sets are given below.

{tank, タンク<*TANKU*>, 水槽<*SUISO*>, 槽<*SO*>}

{tank, 戦車<*SENSHA*>}

These synonym sets define the "container" sense and the "military vehicle" sense of "tank" respectively.

According to the second assumption, our method aligns first-language pairs of related words with second-language pairs of related words via a bilingual dictionary. An alignment of a first-language pair of a polysemous word and its related word with its counterpart in the second language is transformed into a pair of a sense of the polysemous word and a clue. A word related to the polysemous word is called a clue, because it helps to determine the sense of the polysemous word. For example, the alignment of (tank, gasoline) with (タンク<*TANKU*>, ガソリン<*GASORIN*>) results in a sense-clue pair ({tank, タンク<*TANKU*>, 水槽<*SUISO*>, 槽<*SO*>}, gasoline), and the alignment of (tank, soldier) with (戦車<*SENSHA*>, 兵士<*HEISHI*>) results in a sense-clue pair ({tank, 戦車<*SENSHA*>}, soldier).

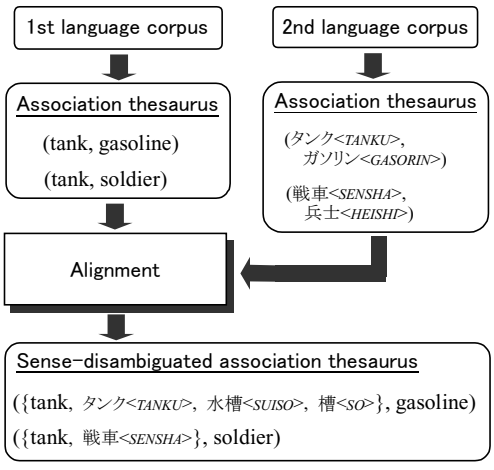


Figure 1: Proposed framework for producing a sense-disambiguated association thesaurus

3 Proposed Method

3.1 Problems and solution

In the framework of aligning pairs of related words translingually, we encounter two major problems: the ambiguity in alignment of pairs of related words, and the disparity of topical coverage between the corpora of the two languages. The following subsections discuss how to overcome these problems.

3.1.1 Coping with ambiguity in alignment

Matching of pairs of related words via a bilingual dictionary often suggests that a pair in one language can be aligned with two or more pairs in the other language (Dagan and Itai 1994; Kikui 1998). To cope with this ambiguity, we evaluate the plausibility of alignments according to the following two assumptions.

(a) Correct alignments are those with pairs of strongly related words.
(b) Correct alignments are accompanied by a lot of common related words that can be aligned with each other.

Then, according to the plausibility of alignments, we calculate the correlation between the senses of a polysemous word and the clues, i.e., words related to the polysemous word.

To precisely estimate the plausibility of alignments according to assumption (b), we should use the correlation between senses and clues. Therefore, we developed an algorithm for calculating the correlation between senses and clues iteratively (see Subsection 3.2.2 for details).

3.1.2 Coping with disparity between corpora

Matching of pairs of related words via a bilingual dictionary often results in a number of pairs not being aligned with any pair. One reason for this is the disparity of topical coverage between the corpora of two languages; another reason is the insufficient coverage of the bilingual dictionary.

To make it possible to acquire the correlations between senses and a clue, even from a first-language pair of related words that cannot be aligned with any second-language pair of related words, we introduce a "wild card" pair. The wild-card pair is a virtual pair related to every word of the second language and implies every sense of the polysemous word of the first language. When a pair cannot be aligned with any other pair, we align it with the wild-card pair compulsorily. We apply the iterative algorithm mentioned in Subsection 3.1.1 to all alignments including alignments with the wild-card pair. Although an alignment with the wild-card pair produces no distinction among the senses of the polysemous word in the first iteration, it produces distinction after the second iteration (An example is given in Section 3.3).

3.2 Algorithm

Our method consists of two steps: translingual alignment of pairs of related words and iterative calculation of correlation between senses and clues. The following subsections give a detailed description of these steps.

3.2.1 Alignment of pairs of related words

An association thesaurus is a collection of pairs of related words with a measure of association between them. In this section, R_X and R_Y denote association thesauri of the first and second languages, respectively. We use mutual information, which is calculated according to co-occurrence statistics, as a measure of association; $MI(x,x')$ denotes the mutual information value of a pair of related words (x,x') ($\in R_X$), and $MI(y,y')$ denotes that of a pair of related words (y,y') ($\in R_Y$), respectively. It should be noted that the measure of association is not limited to the mutual information.

Alignments of pairs of related words between R_X and R_Y, each of which is accompanied by a set of common related words, are extracted through the following procedure.

(1) Extraction of possible alignments

First, for each polysemous word x of the first language, we extract the clue set $X(x)$, which is defined as the set of words related to x, i.e.,

$$X(x)=\{x' \mid (x,x') \in R_X\}.$$

Henceforth, we denote the j-th clue of x as $x'(j)$. Then, for each pair of x and $x'(j)$ ($\in X(x)$), we extract the counterpart set $Y(x, x'(j))$, which is defined as the set of second-language pairs with which the first-language pair $(x, x'(j))$ is possibly aligned, i.e.,

$$Y(x, x'(j))=\{ (y, y') \mid (y, y') \in R_Y, (x, y) \in D, (x'(j), y') \in D\}.$$

Where D denotes a bilingual dictionary, i.e., a collection of pairs consisting of a first-language word and a second-language word that are translations of each other.

(2) Extraction of sets of common related words

(a) In case the counterpart set $Y(x, x'(j))$ is nonempty, for each alignment of $(x, x'(j))$ with (y, y') ($\in Y(x, x'(j))$), we extract a set of common related words $Z((x, x'(j)), (y, y'))$, which is defined as a set of first-language words related to the first-language pair $(x, x'(j))$ and with at least one translation related to the second-language pair (y, y'), i. e.,

$$Z((x, x'(j)), (y, y'))=\{x'' \mid (x, x'') \in R_X, (x'(j), x'') \in R_X\} \cap \{x'' \mid \exists y'' (x'', y'') \in D, (y, y'') \in R_Y, (y', y'') \in R_Y\}.$$

(b) In case the counterpart set $Y(x, x'(j))$ is empty, or the set of common related words $Z((x, x'(j)), (y, y'))$ extracted in the step (a) is empty for all counterparts (y, y') ($\in Y(x, x'(j))$), we align the first-language pair $(x, x'(j))$ with the wild-card pair (y_0, y_0') and construct a set of common related words as follows:

$$Z((x, x'(j)), (y_0, y_0'))=\{x'' \mid (x, x'') \in R_X, (x'(j), x'') \in R_X\}.$$

3.2.2 Calculation of correlation between senses and clues

We define the correlation between each sense of a polysemous word and a clue as the mutual information between them multiplied by the maximum plausibility of alignments that imply the sense. That is,

$$C_n(S(i),x'(j)) = MI(x,x'(j)) \cdot \frac{\max_{y(\in S(i)\cup\{y_0\}),y'} \left(MI(y,y') \cdot \sum_{x'' \in Z((x,x'(j)),(y,y'))} C_{n-1}(S(i),x'') \right)}{\max_k \left\{ \max_{y(\in S(k)\cup\{y_0\}),y'} \left(MI(y,y') \cdot \sum_{x'' \in Z((x,x'(j)),(y,y'))} C_{n-1}(S(k),x'') \right) \right\}},$$

where n denotes the iteration number, and $S(i)$ denotes the i-th sense of the polysemous word x, precisely, the synonym set that defines the i-th sense of x.

The numerator of the second term in the above formula is the maximum of plausibility of alignments that imply the sense, and the denominator is introduced to normalize the plausibility of alignments. The first term of the plausibility of alignment, the mutual information of the second-language pair of related words, corresponds to assumption (a) in Subsection 3.1.1. We assign an arbitrary value larger than zero to the mutual

Alignment	Set of common related words	Sense(s) implied
((tank, troop), (水槽<*SUISO*>, 群れ<*MURE*>))	{air, area, fire, government}	{tank, タンク<*TANKU*>, 水槽 <*SUISO*>, 槽<*SO*>}
((tank, troop), (槽<*SO*>, 多数<*TASU*>))	{area, army, control, force}	
((tank, troop), (戦車<*SENSHA*>, 群<*GUN*>))	{area, army, battle, commander, force, government}	{tank, 戦車<*SENSHA*>}
((tank, troop), (戦車<*SENSHA*>, 多数<*TASU*>))	{Serb, area, army, battle, force, government}	
((tank, troop), (戦車<*SENSHA*>, 隊<*TAI*>))	{Russia, Serb, air, area, army, battle, commander, defense, fight, fire, force, government, helicopter, soldier}	
((tank, gallon), wild card)	{Ford, Institute, car, explosion, fuel, gas, gasoline, leak, natural-gas, oil, pump, toilet, treaty, truck, vehicle, water}	{tank, タンク<*TANKU*>, 水槽 <*SUISO*>, 槽<*SO*>}, {tank, 戦車<*SENSHA*>}

(a) Alignments and accompanying sets of common related words

information of the wild-card pair (y_0, y_0'). Note that the value of the mutual information of the wild-card pair does not have an effect on the results. The second term of the plausibility of alignment, the sum of the correlations between the sense and the common related words, corresponds to assumption (b) in Subsection 3.1.1.

We set the initial values of the correlations between senses and clues as follows:

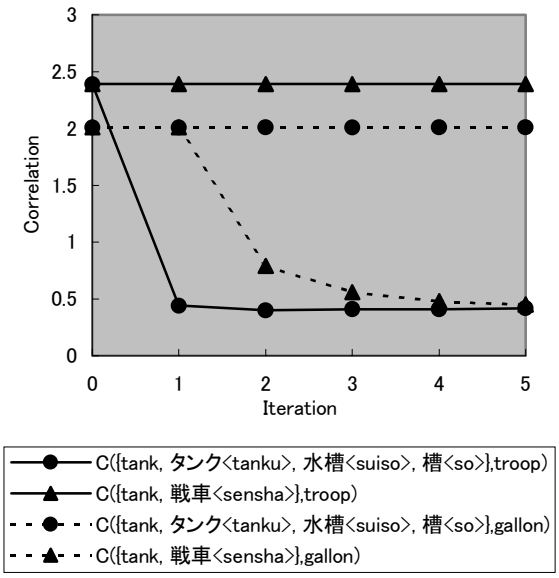
$$C_0(S(i), x'(j)) = MI(x, x'(j)).$$

In the present implementation, we iterate the calculation five times, which makes the correlation values converge. The iteration results in a correlation matrix between the senses of the polysemous word x and the clues. We do not determine the only sense that each clue suggests, but leave using the sense-vs.-clue correlation matrix to application systems.

3.3 Example of calculation

An example of calculating sense-vs.-clue correlations for an English polysemous word "tank" is shown in Figure 2. An English pair of related words (tank, troop) is aligned with five Japanese pairs of related words (水槽<*SUISO*>, 群れ<*MURE*>), (槽<*SO*>, 多数<*TASU*>), (戦車<*SENSHA*>, 群<*GUN*>), (戦車 <*SENSHA*>, 多数<*TASU*>), and (戦車<*SENSHA*>, 隊<*TAI*>). The five sets of common related words that accompany these alignments are shown in Figure 2(a). On the contrary, another English pair of related words (tank, gallon) cannot be aligned with any Japanese pair of related words and, therefore, is aligned with the wild-card pair. The set of common related words that accompanies the alignment of (tank, gallon) with the wild-card pair is also shown in Figure 2(a).

Figure 2(b) shows how the correlation values between the senses of "tank" and the two clues "troop" and "gallon" converge. The correlations with irrelevant senses approach certain small values as the iteration proceeds, while the correlations with relevant senses are kept constant. Note that the correlation value between {tank, タンク<*TANKU*>, 水槽 <*SUISO*>, 槽<*SO*>} and "gallon" and that between {tank, 戦車 <*SENSHA*>} and "gallon", both of which are based on the alignment with the wild-card pair, begin to diverge after the second iteration.



(b) Convergence of correlations

Figure 2: Example of calculating sense-vs.-clue correlations

4 Experiment

4.1 Experimental method

We conducted an experiment to study the feasibility of our method. In this experiment, the first and second languages were English and Japanese, respectively.

First, input data were prepared as follows.

(i) Association thesauri

An English association thesaurus was generated from a Wall Street Journal corpus (July, 1994 to Dec., 1995; 189 Mbytes), and a Japanese association thesaurus was generated from a Nihon Keizai Shimbun corpus (Dec., 1993 to Nov., 1994; 275 Mbytes). The procedure used is outlined as follows (Kaji et al. 2000). Mutual information was calculated for each pair of words according to the frequency of co-occurrence in a window, and pairs of words having a mutual information value larger than a threshold were selected. The words were restricted to nouns and unknown words, which are probably nouns. The size of the window was set to 25 words excluding function words, and the threshold of mutual information value was set to 0.

(ii) Test words

60 English polysemous nouns, whose different senses appear in newspapers, were selected as the test words, and their senses were defined by using their translations into Japanese. The frequencies of the test words in the corpus ranged from 39,140 ("share", the third noun in descending order of frequency) to 106 ("appreciation", the 2,914th noun).

The number of senses defined per test word ranged from 2 to 8, and the average was 3.4.

(iii) Bilingual dictionary

An English-Japanese noun dictionary was compiled from the EDR (Japan Electronic Dictionary Research Institute) English-to-Japanese and Japanese-to-English dictionaries. The resulting dictionary included 269,000 English nouns and 276,000 Japanese nouns.

Then, a sense-vs.-clue correlation matrix was produced for each test word by the method described in Section 3. Finally, the clues were classified according to their correlation with the senses. Namely, the sense having the largest correlation value was selected for each clue on the assumption that a clue is relevant to only one sense (Yarowsky 1993). Although this assumption is not always true, we did so because it is most important to distinguish the most relevant sense from the others.

4.2 Experimental results

Table 1(a) is a classified list of clues obtained for a test word "tank", and Table 1(b) is that obtained for another test word "intelligence". In these lists, clues are sorted in descending order of a score, which is defined as the minimum difference between the correlation with the sense and those with the other senses, i.e.,

$$Score(c) = \min_{S' \neq S} \left[C_5(S,c) - C_5(S',c) \right],$$

where *Score(c)* denotes the score of a clue *c* in the list corresponding to a sense *S*. The score indicates the capability of the clue distinguishing the most relevant sense from the others.

Note that Table 1 lists the top 50 clues for each sense. The total number of clues obtained for each sense of "tank" was as follows:

{tank, タンク<*TANKU*>, 水槽<*SUISO*>, 槽<*SO*>}: 86

{tank, 戦車<*SENSHA*>}: 89

As for "intelligence", two senses were defined: the "ability to learn" sense and the "information" sense. The total number of clues obtained for each sense was as follows:

{intelligence, 知能<*CHINO*>, 知性<*CHISEI*>}: 64

{intelligence, 情報<*JOHO*>, 諜報<*CHOHO*>}: 153

The experiment demonstrated the effectiveness of our method. At the same time, it revealed a few problems. First, when it happens that the second-language association thesaurus includes one or more counterparts of a first-language pair of related words but all of them are incorrect ones, the method causes an error. A sense-clue pair ({tank, タンク<*TANK*>, 水槽<*SUISO*>, 槽<*SO*>}, Poland) included in Table 1(a) is an example. The Japanese association thesaurus included an incorrect counterpart of (tank, Poland), i.e., (水槽<*SUISO*>, 波<*NAMI*>), but it did not include any correct counterpart of (tank, Poland), e.g., (戦車<*SENSHA*>, ポーランド<*PORANDO*>). Consequently, (tank, Poland) was aligned only with (水槽<*SUISO*>, 波<*NAMI*>), which resulted in the incorrect sense-clue pair.

Second, the experimental results show that it is difficult to distinguish a generic or non-topical sense from the other senses. An example is given below. Three senses of "measure" were defined: the "amount, size, weight, etc." sense, the "action taken to gain a certain end" sense, and the "law" sense. The number of clues obtained for each sense was as follows:

{measure, 量<*RYO*>, 尺度<*SHAKUDO*>, 指数<*SHISU*>}: 39

{measure, 対策<*TAISAKU*>, 手段<*SHUDAN*>, 処置<*SHOCHI*>}: 1

{measure, 法案<*HOAN*>, 議案<*GIAN*>, 法令<*HOREI*>}: 93

The method failed to obtain effective clues for selecting the second sense, which is extremely generic, although "measure" in this sense occurred frequently in the corpus.

5 Future Extensions

5.1 From sense-vs.-clue correlation to sense-vs.-sense correlation

The sense-vs.-clue correlation matrix is an intermediate form of sense-disambiguated association thesaurus. It should be transformed further into a sense-vs.-sense correlation matrix. This transformation can be done straightforwardly.

Let's take a pair of related words (tank, troop) as an example. The sense-vs.-clue correlation matrix produced for a polysemous word "tank", which is denoted as *M*(tank), includes the following pairs of a sense and a clue.

({tank, タンク<*TANKU*>, 水槽<*SUISO*>, 槽<*SO*>}, troop)

({tank, 戦車<*SENSHA*>}, troop)

Likewise, the sense-vs.-clue correlation matrix produced for another polysemous word "troop", which is denoted as *M*(troop), includes the following pairs of a sense and a clue.

({troop, 群れ<*MURE*>, 群<*GUN*>, 多数<*TASU*>}, tank)

({troop, 軍隊<*GUNTAI*>, 隊<*TAI*>, 部隊<*BUTAI*>}, tank)

So a pair of senses is produced by combining two pairs of a sense and a clue, one from *M*(tank) and the other from *M*(troop). The correlation value of the pair of senses is defined as the minimum of the correlation values of the combined pairs of a sense and a clue. For example,

C({tank, 戦車<*SENSHA*>}, {troop, 軍隊<*GUNTAI*>, 隊<*TAI*>, 部隊<*BUTAI*>}) = min [*C*({tank, 戦車<*SENSHA*>}, troop), *C*({troop, 軍隊<*GUNTAI*>, 隊<*TAI*>, 部隊<*BUTAI*>}, tank)].

5.2 Use of syntactic co-occurrence

We have conducted another experiment to evaluate word sense disambiguation using the sense-vs.-clue correlation matrix, which will be reported in detail at another opportunity. Although the overall results have been promising, our method has its limitations.

The present method deals with only nouns, and it extracts clues for word sense disambiguation according to co-occurrence in a window. However, it is obvious that doing this is not suitable for all polysemous words. Syntactic co-occurrence is more useful for disambiguating some sorts of polysemous words (Lin 1997). It is an important and interesting research issue to extend our method so that it can extract clues according to syntactic co-occurrence. This extended method does not replace the present method; however, we should combine both methods or use the one suitable for each polysemous word.

The framework of our method is compatible with syntactic co-occurrence. Basically, we only have to incorporate a parser into the association thesaurus generator. A parser of the first language is indispensable, but a parser of the second language is not. As for the second language, we may use co-occurrence in a small-sized window instead of syntactic co-occurrence.

6 Discussion

(a) List of clues relevant to each sense of "tank"

{tank, タンク<*TANKU*>, 水槽<*SUISO*>, 槽<*SO*>} *		{tank, 戦車<*SENSHA*>} **	
Clue	Score	Clue	Score
Walbro	5.13	artillery	4.04
ammonia	4.83	Grozny	2.98
static electricity	4.45	commander	2.65
Mrs. Tramm	4.15	Chechen	2.63
gasket	4.13	Chechnya	2.56
Jon-Luke	3.91	Mr. Yeltsin's	2.54
vapor	3.85	Patton	2.43
fuel tank	3.74	Serb	2.42
Aruba	3.55	Bosnian government	2.40
Zeus	3.24	missile	2.28
kangaroo	3.24	Cutiron	2.27
fuel	2.95	ball	2.17
pickup truck	2.87	treaty	2.17
leak	2.76	Yeltsin's	2.16
toilet	2.74	ammunition	2.14
tank barge	2.61	Polish method	2.03
fish	2.56	helicopter	2.01
Spar	2.43	soldier	2.00
tide	2.42	Mr. Gaffney	1.97
truck	2.34	Gaffney	1.95
pump	2.26	troop	1.92
liquid	2.25	thud	1.87
underground	2.24	weapon	1.84
Pena	2.23	civilian	1.82
concrete	2.22	Belarus	1.80
pickup	2.21	assault	1.73
gasoline	2.19	Bosnian	1.71
static	2.17	method	1.71
float	2.12	rebel	1.70
ozone	2.05	Yeltsin	1.68
temperature	1.94	NATO	1.66
recall	1.93	Mr. Yeltsin	1.64
electricity	1.90	parliament	1.51
tank car	1.85	Russian	1.48
plastic	1.84	army	1.39
explosion	1.82	U.N.	1.33
GM	1.78	bomb	1.25
rush	1.76	Army	1.25
safety	1.73	Polish	1.19
Poland	1.71	military	1.17
Mercedes	1.69	Rutkowski	1.15
emission	1.68	Pentagon	1.11
barge	1.60	defense	1.09
gallon	1.55	battle	1.07
design	1.46	force	1.05
fragment	1.42	Progress	1.02
bottom	1.39	Heritage Foundation	1.00
road	1.39	ton	1.00
Shell	1.35	column	0.97
blue	1.30	Force	0.92

* a large container for storing liquid or gas
** an enclosed heavily armed, armored vehicle

(b) List of clues relevant to each sense of "intelligence"

{intelligence, 知能<*CHINO*>, 知性<*CHISEI*>} ***		{intelligence, 情報<*JOHO*>, 諜報<*CHOHO*>} ****	
Clue	Score	Clue	Score
trait	3.76	CIA	5.19
curve	3.43	spy	4.55
domain	3.03	mole	4.49
secret	1.89	Pyongyang	3.12
shoot	1.88	U.S. military	3.10
consequence	1.78	palace	3.01
Hamlet	1.73	Directorate of Operation	2.91
Mainstream Science	1.67	intelligence budget	2.75
human	1.60	secret service	2.75
community	1.50	rod	2.74
domain name	1.50	satellite	2.61
capability	1.49	double agent	2.52
understanding	1.47	Defense Intelligence Agency	2.45
outcome	1.44	Woolsey	2.44
writer	1.43	Deutch	2.39
conclusion	1.42	U.S. intelligence	2.38
score	1.39	agent	2.37
IQ test	1.28	Intelligence Committee	2.35
book	1.28	Shalikashvili	2.32
IQ	1.27	intelligence community	2.31
author	1.26	Mr. Deutch	2.31
analysis	1.20	intelligence agency	2.27
knowledge	1.12	Kalugin	2.26
difference	1.07	weapon	2.25
Bell Curve	1.02	Mr. Woolsey	2.23
story	0.96	defector	2.19
study	0.95	intelligence service	2.17
child	0.93	Ames	2.11
test	0.90	espionage	2.09
Curve	0.89	Aspin	2.01
psychologist	0.88	Torricelli	1.98
society	0.88	analyst say	1.98
Mainstream	0.81	Seoul	1.98
woman	0.79	policy maker	1.97
research	0.71	Serb	1.91
white	0.67	assertion	1.90
academic	0.65	TI	1.89
fluid	0.64	fraction	1.81
tool	0.63	terrorism	1.81
life	0.63	annual budget	1.79
extreme	0.62	North Korean	1.78
Murray	0.59	KGB	1.73
gathering	0.59	State Department	1.70
man	0.57	military service	1.70
way	0.51	middle	1.69
view	0.49	Mr. Wolf	1.65
Science	0.47	East German	1.64
good	0.45	launder	1.64
discussion	0.42	Defense	1.64
source	0.39	Cold War	1.63

*** ability to learn, reason, and understand
**** information about an enemy

Table 1: Excerpt from the produced sense-disambiguated association thesaurus

6.1 Usefulness of sense-disambiguated thesaurus

The usefulness of the sense-disambiguated association thesaurus for information retrieval and filtering is discussed below. First, when it is shared by a system and users, the sense-disambiguated association thesaurus enables users to input

unambiguous queries. The system does not need to sense-disambiguate queries, since they are already disambiguated.

Second, the sense-disambiguated association thesaurus definitely improves the performance of query expansion. Because it enables a query to be expanded with related words relevant to the sense of user's interest, not with related words regardless of sense.

Third, the sense-disambiguated association thesaurus can be effectively used to sense-disambiguate documents. The sense of a polysemous word in a document is determined by comparing the context with the clues of each sense.

Finally, the sense-disambiguated association thesaurus, in which a sense is defined by a set of bilingual synonyms, functions as a user interface for translingual information retrieval. A user, who may not understand the second language, recognizes senses via the clues of the first language, and the system obtains second-language translation(s) from the synonym set specified by the user.

6.2 Word sense disambiguation and bilingual corpora

Word sense disambiguation using bilingual corpora has an advantage in that it enables unsupervised learning. However, the previous methods, which align instances of words (Brown et al. 1991), require a parallel corpus and, therefore, are applicable to limited domains. On the other hand, our new method requires a comparable corpus. The comparability required by the new method is very weak: any combination of corpora of different languages in the same domain, e.g., Wall Street Journal and Nihon Keizai Shimbun, is acceptable as a comparable corpus. Thus the new method has an advantage over the previous methods in being applicable to many domains.

Word sense disambiguation using bilingual corpora has a limitation because the senses of a first-language polysemous word are not always lexicalized differently in the second language. Second-language translations that preserve the ambiguity cause erroneous disambiguation. To avoid this problem, we eliminate translations that preserve the ambiguity from the synonym sets defining senses.

An example is given below.

{title, 肩書き<*KATAGAKI*>, 称号<*SHOGO*>, ~~タイトル<*TAITORU*>~~, 敬称<*KEISHO*>}

{title, 題名<*DAIMEI*>, 題目<*DAIMOKU*>, 表題<*HYODAI*>, 書名<*SHOMEI*>, ~~タイトル<*TAITORU*>~~}

{title, ~~タイトル<*TAITORU*>~~, 選手権<*SENSHUKEN*>}

These synonym sets define three senses of "title", the "person's rank or profession" sense, the "name of a book or play" sense, and the "championship" sense. A Japanese translation "タイトル<*TAITORU*>", which represents all these senses, is eliminated from all these synonym sets.

The method of eliminating ambiguous translations is effective as far as we can find alternative translations. However, it is not always the case. An essential approach to solving this problem is to use two or more second languages (Resnik and Yarowsky 2000).

7 Conclusion

Sense-disambiguated association thesauri, in which word senses are distinguished according to the related words, were proposed. It is produced through aligning pairs of related words between association thesauri of different languages. To overcome both the problem of ambiguity in the translingual alignment of pairs of related words and that of disparity of topical coverage between the association thesauri of different languages, an iterative algorithm for calculating the correlation between the senses of a polysemous word and its related words according to the set of words related to both the polysemous word and each of the related words was developed. An experiment using English and Japanese association thesauri, both of which were generated from newspaper article corpora, demonstrated that the algorithm produces a sense-disambiguated association thesaurus successfully. The usefulness of the sense-disambiguated association thesauri for information retrieval and filtering was also discussed.

Acknowledgments

This research was sponsored in part by the Telecommunications Advancement Organization of Japan and the New Energy and Industrial Technology Development Organization of Japan.

References

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1991. Word-sense disambiguation using statistical methods. In *Proceedings of the 29th Annual Meeting of the ACL*, pages 264-270.

Dagan, Ido and Alon Itai. 1994. Word sense disambiguation using a second language monolingual corpus. *Computational Linguistics*, 20(4): 563-596.

Jing, Yufeng and W. Bruce Croft. 1994. An association thesaurus for information retrieval. In *Proceedings of a Conference on Intelligent Text and Image Handling "RIAO'94"*, pages 146-160.

Kaji, Hiroyuki, Yasutsugu Morimoto, Toshiko Aizono, and Noriyuki Yamasaki. 2000. Corpus-dependent association thesaurus for information retrieval, In *Proceedings of the 18th International Conference on Computational Linguistics*, pages 404-410.

Kikui, Genichiro. 1998. Term-list translation using monolingual word co-occurrence vectors. In *Proceedings of the 17th International Conference on Computational Linguistics*, pages 670-674.

Lin, Dekang. 1997. Using syntactic dependency as local context to resolve word sense ambiguity. In *Proceedings of the 35th Annual Meeting of the ACL / the 8th Conference of the EACL*, pages 64-71.

Mandala, Rila, Takenobu Tokunaga, and Hozumi Tanaka. 1999. Combining multiple evidence from different types of thesaurus for query expansion. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development of Information Retrieval*, pages 191-197.

Miller, George A. 1990. WordNet: an on-line lexical database. *International Journal of Lexicography*, 3(4): 235-312.

Rapp, Reinhard. 1995. Identifying word translations in non-parallel texts. In *Proceedings of the 33rd Annual Meeting of the ACL*, pages 320-322.

Resnik, Philip and David Yarowsky. 2000. Distinguishing systems and distinguishing senses: new evaluation methods for word sense disambiguation. *Natural Language Engineering*, 5(2): 113-133.

Schuetze, Hinrich and Jan O. Pedersen. 1994. A cooccurrence-based thesaurus and two applications to information retrieval. In *Proceedings of a Conference on Intelligent Text and Image Handling "RIAO'94"*, pages 266-274.

Yarowsky, David. 1993. One sense per collocation. In *Proceedings of the ARPA Human Language Technology Workshop*, pages 266-271.

Designing Text Filtering Rules: Interaction between General and Specific Lexical Resources

Antonio Balvet
Université Paris X-Nanterre / UMR MoDyCo
200, avenue de la République 92001 Nanterre
antonio.balvet@u-paris10.fr

Abstract

In this paper, we present a modular linguistic wizard for information retrieval applications based on explicit rules. We focus on the main features of the present version of the linguistic wizard: extracting rough verb subcategorization frames from existing corpora and querying a large coverage, corpus-independent semantic network (i.e. Memodata's Dictionnaire Intégral). We also provide performance evaluation measures computed on the basis of a rules-based text filtering system, in order to quantify the gain achieved by making use of the linguistic wizard. The performance evaluation figures are therefore based on a manual run and a "random" run, which provide, respectively, the maximum and minimum quality bounds for a system filtering texts through explicit rules.

1. Introduction

How to provide the right information to the right person at the right time ? This question has become all the more crucial in automatic Information Retrieval (IR) systems, which have to deal with ever-increasing volumes of data. The question at hand, which is, in fact, about relevancy, also applies to the field of Information Filtering (IF). Automatic IF systems, let them be statistics-based or rules-based, are rapidly confronted to the issue of enhancing their initial performance.

In this paper, we show how to integrate both corpus-driven and corpus-independent resources in order to provide more relevant information to the final user.

We first give a historical background of the field of IF, from H.P. Luhn's initial specifications to the current TREC[1] definition. Then, we justify our approach to IF, which is based on explicit categorization rules. In the following section, we present the main features of a LInguistic wiZARD and the gain which can be attained by integrating the LIZARD into the text categorization process, compared to a manual approach.

1.1. From Selective Dissemination of Information to Text Filtering

Providing relevant information is a standard requirement for information systems, let them be human or computer-assisted. This requirement was formally stated in (Luhn, 1958), in the initial framework of public libraries. Luhn was one of the first authors to specify the task which was later to be known as "Information Filtering". The then called "Selective Dissemination of Information" (SDI) activity specified every aspect of a process aimed at fulfilling a full-scale information service, from profiles (information needs) to social filtering (collaborative filtering).

1.2. Filtering Texts: a TREC Definition

The TREC international evaluation conferences, sponsored mainly by the United States' federal government, have taken Luhn's initial specifications to their farthest point, providing the field of Information Retrieval (IR) with standard evaluation procedures as well as standardized tasks and data (gigabytes of text corpora).

1.2.1. Text Filtering as a "Push" Activity

Within the general framework of IR, the IF task was first formalized in 1995. The IF "track", as specified in (Lewis, 1995), is defined as belonging to the range of "push" activities, as opposed to "pull" ones. This means IF is a task where queries (profiles) are stable while the textual data are dynamic (high update rate).

1.2.2. A Binary Selection Decision

The TREC conferences also defined IF as implementing a "binary text classification". The emphasis laid on the binary (YES/NO) aspect of the selection decision distinguishes IF from other push activities such as routing[2], where texts are classified according to a relevance rate computed mainly on the basis of the occurrence probability of a given set of terms (continuous selection decision).

We state that the TREC definition of IF implies an approach to the problem of automatic text classification based on explicit rules, while the routing definition implies a machine-learning, or even statistics-based, one, as explicit rules directly implement binary pattern-matching.

2. Categorizing Text with Rules

2.1. Why Use Rules ?

2.1.1. Explicit *vs.* Implicit Categorization Rules

Machine-learning approaches rely on large amounts of learning material and on the fine-tuning of the often time and space-consuming learning algorithms used. These characteristics make the machine-learning approaches suitable to the classification of stable data repositories, and for activities that do not require -even close to- real-time processing. That is to say that these approaches are particularly well suited to *pull* activities, where data are stable and queries are transient.

[1] Text REtrieval Conference, see (Harman, 1993).

[2] See (Robertson & Hull, 2001) for an overview of the filtering track's subtask (adaptive and batch filtering, routing) specifications.

These approaches are also well adapted to the evaluation procedures defined in the TREC conferences, which are based on a two-stage process[3] for defining reference corpora. The first phase collects all the evaluated systems' outputs, for precedent editions of the evaluation conference[4], from which a portion is extracted, proof-read by human assessors in the second phase[5]. This portion of the original collection is considered as the reference (test) corpus for all evaluated systems.

2.1.2. Real-Scale Data and Explicit Rules

Real-world IF does not fit well in the frame of the TREC conferences, though. As will be seen later in the paper, the available data in actual applications (both "learning" and "testing" corpora) are sometimes quite scarce, amounting to the maximum to megabytes rather than gigabytes of text, thus ruling out *de facto* data-intensive approaches. Furthermore, most of the relevant text units have very low occurrence rates[6], to such extent that detecting these "low signals" appears fundamental to the task of filtering documents. This constitutes yet another indirect justification for the use of symbolic rules, inherently independent from occurrence rates.

2.2. What Rules to Use ?

2.2.1. Keywords-Based Pattern-Matching

In the field of rules-based IF systems, keyword-based pattern-matching approaches are the most common ones. Most of the keyword-based systems are but instances of the renown "grep" command found on Unix-like systems. In keywords-based systems, filters are constituted of search strings, and profiles are Boolean operations on individual filters (NOT, AND, OR). Matching, thus filtering, is limited to exact match of a given string.

2.2.2. Regular Expressions-Based Pattern-Matching

Regular expressions-based IF systems are more flexible than keywords-based ones, in the sense that wildcards (+ and * operators), Boolean (&, |, !) and range (e.g. [a-z]) operators allow for extended search patterns[7]. Those basic features are the building blocks for efficient IF systems. Nevertheless, regular expressions-based IF systems are limited by their syntax, which naïve users are not always willing to master.

[3] See (Voorhees & Harman, 2001) for more details.

[4] This procedure is known as the "pooling method".

[5] The pooling method appears common to all text-related tasks, even the filtering track. Given that most of the evaluated systems rely on implicit categorization rules, this evaluation procedure clearly disfavors alternative approaches, such as explicit rules-based ones.

[6] Named entities (e.g. person/products/company names) register very low occurrence rates compared to other text units; in some cases, non-ambiguous persons/products/companies are only mentioned once.

[7] For example, the following search pattern retrieves all conjugated forms of the French verb "manger": mang*, together with "mangue", "mangeoire" etc..

Neither isolated keywords nor regular expressions appear appropriate for filtering texts: the cost of developing text categorization rules based solely on those basic elements appears overwhelmingly high. Therefore, once stated the necessity of using explicit rules for filtering texts, we need to investigate alternative explicit rules.

3. Local Grammars as Text Filtering Rules

In this section, we introduce corpus-processing oriented symbolic rules: "local grammars" as defined in (Gross, 1975). We show how these local grammars can be used for specific tasks such as text filtering, following the approaches introduced in (Grefenstette, 1996) and (Roche, 1993), who use cascades of Finite State Transducers (FST) for Natural Language Processing-related tasks, in an iterative fashion[8].

3.1. The Local Grammars Approach

Alongside the chomskyan "classical" paradigm for Natural Language Processing (NLP), alternative approaches exist, focusing more on the phrase than on the sentence level, even though pursuing the same goal of arriving at a complete description of human natural language.
Harris's "link grammar"[9] and Gross's "local grammars" are instances of such alternative approaches.

3.1.1. Describing Complex Lexical Units

We focus on the concept of local grammars, such as illustrated in the work of the LAboratoire d'Automatique Documentaire et Linguistique (LADL), and implemented through the Intex platform[10].
Local grammars rely heavily on a distributional analysis of a given corpus. They describe linguistic constituents which are closer to idiomatic phrases than to general sentences, which other distributionalist authors such as B. Habert have named "complex lexical units". Most of the time, local grammars capture very contextual properties of lexical items. Thus, the local grammars approach appears very productive for specialized domains/fields of expertise and terminology-oriented tasks.
The Intex system is based on local grammars, expressed as Finite-State Transducers (FST), which are used as a formalism, as a parsing technique and as a data structure for linguistic knowledge representation. Preprocessing rules (sentence boundaries detection, input normalization), tagging dictionaries (simple and compound words, frozen expressions, named entities etc.) and parsing rules are thus represented as FSTs. This has the effect of ensuring optimal consistency in both data and processes, together with processing efficiency (speed) and extensibility[11]. Moreover, Intex comes with standard large-coverage lexical resources for French: simple and compound words dictionaries, lexicon-

[8] Information processed in earlier stages constrain subsequent analyses. See (Abney, 1996) for other applications such as parsing.

[9] Introduced in (Harris, 1968) and developed ever since.

[10] See (Silberztein, 1993).

[11] Extending/revising a set of local grammars boils down to editing symbolic rules, expressed in a graphical format for better readability (see Figure 1).

grammar tables for frozen expressions, and specialized local grammars (occupational nouns, toponyms, dates, roman numerals etc.).
Figure 1 below shows an example of a very simple local grammar, used to describe, parse and translate roman to modern numerals (transducer output). This very simple local grammar allows for parsing and transformation of input strings: the pattern to match is described in the boxes (e.g. I, II, IX …), the output of the transformation is written in bold (e.g. 1, 2, 9 …).

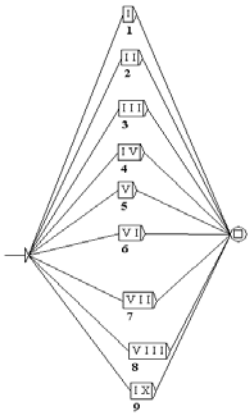


Figure 1: a local grammar used to parse and transform roman numerals into modern numerals

The Intex system also allows for multiple embedding of local grammars, ensuring sufficient computational power for the most common cases by extending FSTs to Augmented Transition Networks.

3.1.2. Describing "Topical Signatures" as local grammars

Our approach to text filtering aims at:

- isolating typical complex lexical units of a given domain/field of expertise, which we call "topical signatures", through a distributional analysis of reference corpora, close to terminological studies in its philosophy,
- describing those expressions as a set of local grammars,
- use this set of local grammars in the process of text categorization.

Typical expressions are thus mainly taken from reference corpora, nevertheless we also make use of thesaurus-like resources in order to provide better coverage for our topical signatures. The approach described here is close to Riloff's[12] in its philosophy, except that topical signatures range from single (e.g. non-ambiguous person names) to complex units (typical phrases such as "monter au capital de"), rather than word pairs exclusively.

3.2. Profiles and Filters as Local Grammars

Filtering textual information involves at least two objects: the user's personal information need, which will be referred to as a "profile", and the individual filters matching relevant parts of documents.
In a rules-based approach, a profile is a conjunction/disjunction or negation of existing filters. In our approach, both filters and profiles can be expressed as local grammars: profiles are conjunctions/disjunctions or negations of existing local grammars matching textual sequences considered relevant by experts of the field.
For example, in order to automatically retrieve relevant documents about the "Mad Cow Disease" epidemics, local grammars for detecting phrases stating the following facts could be designed: typical symptoms have been found on animals, animals have been put down in order to prevent contagion, then perform a Boolean conjunction (AND) operation on those filters in order to implement a "Mad Cow Disease" profile.
Translating filters and profiles into local grammars is consistent with the Intex system's convention. Nevertheless, it implies rendering users' knowledge of the field explicit, which is an inherent source of limitation in coverage of the problem. In some cases, finding categorization rules based on textual cues would even seem awkward, as users rely on implicit, rather than explicit, knowledge and synthetic, rather than analytic, categorization strategies. In those cases, messages are understood in a global fashion and users rely more on their experience of the field than on the actual textual cues contained in the messages. Therefore, the local grammars approach is inherently limited in coverage, even though it complies fully with the TREC specifications[13].

3.3. Problems with Designing Local Grammars by Hand

3.3.1. Experience from a Functional Prototype

A functional prototype of an information filtering system based on local grammars has been designed at a French corporate research laboratory[14]. The prototype, connected to the Agence France Press (AFP) newswire, has demonstrated the feasibility and usability of a rules-based approach to text categorization, together with processing efficiency on French news extracts (ranging from 1 to 10 Kbytes): average processing time (input normalization, filtering and routing) was estimated to 30 seconds per document, which is inferior to the AFP newswire update frequency (1 document per minute).
Nevertheless, the prototype has also shown the necessity to semi-automatically expand user-designed filters, as users cannot explicitly predict future utterances related to a particular domain/area of expertise. In other words, the operational prototype lacked "linguistic calculus" features.

3.3.2. Managing "Flat" Local Grammars

In day-to-day practice, users are quickly confronted to resources management issues due to the proliferation of very specialized (context-dependent) local grammars.

[12] See (Riloff, 1994), where the author presents a strategy focused on extracting non-ambiguous pairs of words from text corpora for "portable" text classification systems.

[13] Our experience of the field has shown us that the TREC specifications for the text filtering task do not account for the complex cognitive (categorization) operations involved in human text filtering.

[14] See (Balvet *et al*., 2001) for more details.

Moreover man-made local grammars are often too restrictive: for example, common phrase alternations (passive/active voice, nominalization etc.) are not available as a standard resource, therefore users usually develop very rough and imperfect grammars for such alternations. Semantic expansion is not implemented in the Intex platform either. Thus, users are rapidly confronted to the problem of expanding their local grammars in a semi-automatic fashion for better coverage and reusability.

4. LIZARD, Main Features

In this section, we introduce the concept of expanded local grammars, and the tools available for French in order to achieve a kind of semi-automatic query expansion on user defined local grammars used as filters, through the LIZARD system.

LIZARD is a tool we have designed, allowing the integration of heterogeneous lexical resources. It was built using the Open Agent Architecture, which provides efficient agent and remote-access capability to heterogeneous systems: OAA allows the creation of Java/C/C++ and Prolog-based agents. The current version of the LInguistic wiZARD is still in alpha status, providing minimal expansion of local grammars: inclusion of synonyms and hyper/hyponyms of terms found in the user's local grammars is suggested, by querying a Memodata agent. Extension to semantically related verbs, together with their preference selection frames extracted from the reference corpora is made available by querying a verb selection preference database.

Syntactic variants are also made available through the following transformations, implemented via local grammars: passive/active form, nominalization with support-verb (e.g. augmenter son capital → procéder à l'augmentation de son capital), and multiple insertions (adjectives, adverbs, phrases etc.).

4.1. Overview of the Global Architecture

4.1.1. A System-Oriented Application

The figure below presents the general system-oriented architecture of the linguistic wizard. Each box in the system diagram represents a processing module. Each module offers standard linguistic and corpus-related facilities, based on existing components, following a "component off-the-shelf" philosophy: every module is thought as a service, therefore each particular component can be replaced by another equivalent component[15].

The Intex module's services are all FST-related text operations (text normalization, pattern-matching, local grammars editing).

The Memodata module's services are all semantics-related operations (retrieving semantically/morphologically related words and phrases, comparing pairs of words or phrases).

[15] For example, the Intex module can be replaced by AT&T's FSM package.

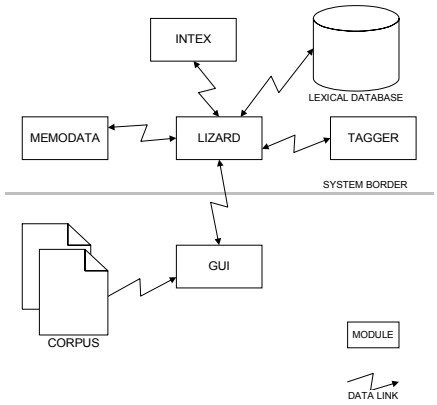


Figure 2: system-oriented architecture of the LIZARD

Communication paths (queries and responses) are represented by broken arrows. The gray line represents the "visible" limits of the whole LIZARD system: the only module accessible by the end-user is the Graphical User Interface (GUI).

The output of the system is a lexical database of domain-dependent typical expressions, which we call "topical signatures".

4.1.2. An Agents-Based System

Developing an agents-based system on top of the modular application shown in Figure 2 was rendered possible by the integration of Stanford Research Institute's Open Agent Architecture (OAA). Within this framework, turning a software component into an autonomous agent is rather straightforward: each module provides services and all agents communicate in a "blackboard" fashion via a central supervising agent called "Supervisor". The Supervisor centralizes all requests from all declared agents and routes them to the appropriate service-rendering agents.

Designing an agent-based NLP system allows the system to operate in a distributed (client/server) fashion over a network (intra/internet), so that memory-intensive applications, such as Memodata's Dictionnaire Intégral, can be run on a dedicated server.

4.2. Rough Verb Subcategorization Frames Extraction

The LIZARD system implements an interactive distributional analysis of reference corpora, in order to extract rough subcategorization frames for relevant verbal entries. For this task, the reference corpora need to be unambiguously tagged and lemmatized, so that only one tag per individual word remains. A first customizable generalization phase deletes most of the Adjectives, all the Adverbs, numbers and punctuation signs. This first phase only keeps those parts of speech generally considered informationally relevant, such as Nouns (part-of-speech information only), Determiners, Verbs (infinitive form), Prepositions and Pronouns.

A second generalization phase provides general subcategorization frames such as: V-Det-N, V-Prep-Det-N etc. Those frames form the core of the domain's set of topical signatures. Once the subcategorization frames have been extracted and validated by the user, all

selected topical signatures candidates are transformed in order to conform to the lexicon-grammar format[16], which the Intex system translates into local grammars[17].

4.3. Querying a Semantic Network

4.3.1. Integrating the Dictionnaire Intégral

Memodata's Dictionnaire Intégral (DI), a corpus-independent semantic network, is presented in detail in (Dutoit, 2000), therefore we only mention the features used by the LIZARD system. The DI comes with a Java API, allowing easy integration in existing systems. This API gives access to common word functions such as synonymy, hyper/hyponymy, morphological relatedness etc. It also gives access to less common features, such as phrase and sentence functions. Those functions are essential to our system, in that they allow easy retrieval of semantically related phrases, not just words. Those functions also allow rough semantic evaluation of two phrases based on a proximity algorithm developed by D. Dutoit.

4.3.2. Expanding Core Topical Signatures

The candidate topical signatures extracted from reference corpora in the previous phase are expanded by querying the DI for related words and phrases: hyper/hyponyms[18], morphologically related words[19] and related phrases are interactively integrated into the existing core topical signatures[20]. The general philosophy is to compensate lack of coverage of hand-designed local grammars by integrating common (extracted from the DI) as well as specific knowledge (extracted from reference corpora) into local grammars intended to be used for automatic text categorization tasks.

5. Performance Evaluation

In the following evaluation, we compare the performance of three text filtering systems[21], following an evaluation procedure aiming at emphasizing the gain attainable by integrating the LIZARD in a rules-based text filtering system.

The first one, the "manual" system, uses hand-designed local grammars[22] and sets the upper bound in quality for the evaluation runs.

The second one, the "computer-assisted" system, is based on the LIZARD and allows us to evaluate our local grammars expansion approach.

The third one, the "random" system, uses random filtering rules and simulates a black-box, automatic text categorization system. This system sets the lower bound for the evaluation runs: we expect our system to perform at least better in quality than the random system.

5.1. The Corpus

5.1.1. A Financial News Corpus

The corpus comes from a private company, Firstinvest, providing targeted financial news to its customers. The financial news extracts are routed by human operators to the appropriate clients in a binary fashion. Thus, the corpus constitutes a reference for an automatic IF system: the situation described matches the TREC definition for the document filtering track.

The reference corpus is organized as follows: 2.6 Mo of French financial news extracts in ASCII format, 19 topics (from Internet-related news to profit warning, rumors and interviews). We focus on topic 19, "corporate transactions", describing scenarios of companies buying or selling parts of their capital.

The performance evaluation measures we used (see below) are based on the number of matches **and** the number of incorrectly retrieved documents (i.e. negative examples) registered for the tested system. As the entire corpus has reached us completely sorted, providing us only with positive examples for each topic, we needed to provide a set of negative examples (noise). Therefore, 50 news extracts (66 Kbytes) of noise corpus, assigned to other topics than the one tested here, were extracted manually from the whole corpus for evaluation purposes.

5.1.2. Learning and Test Corpus

Topic 19 totals 303 documents, which we segment in two parts: 2/3 for the learning corpus (200 documents) and 1/3 (103 documents) for the test corpus. For each evaluation run, standard precision and recall rates (P/R, see below) were computed based on the comparison between each system's output and the reference corpus from Firstinvest.

As the reader will undoubtedly notice, these figures are very far from those of evaluation conferences such as TREC, even though they correspond to real-life data. In fact, the reference data we describe can not be compared to the reference corpora provided by TREC editions: the documents were sorted entirely by hand, they represent but a fraction, in size, of the TREC test suites, and they match an actual information need from users ready to pay for the service provided by Firstinvest.

5.2. Setting the Upper and Lower Bounds to Evaluate the LIZARD Approach

5.2.1. The Manual Run

S. Bizouard designed a set of local grammars for an information extraction (IE) system evaluation experiment undertaken at Thales RT. Following E. Riloff, we assert that IF and IE are complementary activities. Thus, IE local grammars can be used as IF profiles. Therefore, we took S. Bizouard's hand-designed local grammars as a reference for the manual run. Those

---

[16] Syntactic and semantic information, associated to a lexical entry, are expressed as a set of binary features (+/-). Lexicon-grammar tables also include lexical parameters such as the form of a typical complement.

[17] See (Silberztein, 1999) for more details on the lexicon-grammar feature of the Intex system.

[18] Specifics and generics in Memodata's terminology.

[19] For example: "achat" (Noun) which is morphologically related to "acheter" (Verb).

[20] The current version of the LIZARD does not make use of the semantic net navigation customization features, implemented in the DI, yet.

[21] Performing a form of "batch" filtering according to the definition of (Robertson & Hull, 2001).

[22] See (Bizouard, 2001) for more details.

resources were designed following the topical signature approach described above.
Precision and recall of S. Bizouard's grammars do not equal the theoretical 100%, even though they are the result of considerable effort[23]. Our hypothesis is that this apparent lack of coverage is mainly due to implicit knowledge used by experts of the field in classifying texts, which explicit approaches such as the one described in this paper can not capture. The apparent lack of coverage of the hand-designed local grammars also appears due to a lack of proper selection preference constraining: some rules remain too "open" by failing to provide a closed list of possible complements for some very common verbs[24].
Our implicit hypothesis is that manually-designed resources tend to rate high in precision, but low on recall, so the manual run will give the higher precision bound.

5.2.2. The LIZARD Run

The computer-assisted run shows the impact of the integration of both corpus-driven and corpus-independent resources on a text categorization task. In other words, the computer-assisted run implements a query expansion approach based on explicit resources (verb subcategorization frames, semantically related words ...).
The implicit hypothesis is that the natural low recall rates tendency of the manual approach can be compensated by elements (parameters) taken both from existing specialized corpora and general purpose semantic nets (i.e. Memodata's Dictionnaire Intégral).

5.2.3. The "Random" Run

This run is based on a fully automatic text filtering system, which randomly selects documents, independently from their content. The random run shows what can be achieved by a text filtering system which decision selection rules are hidden (black-box system). The implicit hypothesis is that the random run will set the lower bound for both recall and precision (around 50%), in other words, the minimal recall and precision rates expected from the computer-assisted system.

5.3. Figures

RUN	Matches	Noise
Manual	76	9
LIZARD	103	13
Random[25]	53.2	24.8

Figure 3: performance table for each run

These figures were computed on the test (103 documents) and noise corpus (50 documents). As the table shows, the LIZARD system retrieves all the relevant documents. Moreover, it only is responsible for about 1/3 additional noise (compared to the manual run).
The figures presented below give the standard precision/recall rates for each run[26]. As the figures show, the LIZARD system performs very good in recall (100%) and compares equally to the manual run in precision, despite of its "noise" rate being slightly higher.

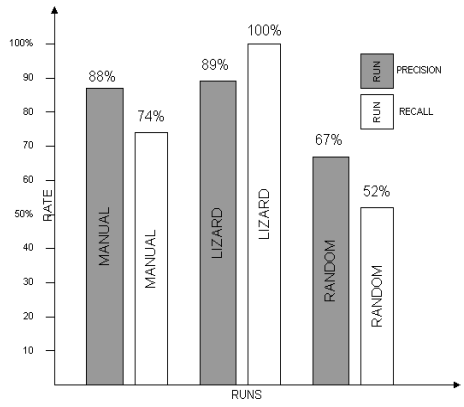


Figure 4: precision/recall rates of three text filtering approaches

5.3.1. Discussion of Figures

The figures presented above show the performance of three types of text filtering systems:

- a system relying exclusively on manually-designed categorization rules, centered on topical signatures,
- a system based on computer-assisted categorization rules (topical signatures), integrating mainly subcategorization frames extracted from the learning corpus, and suggestions from a thesaurus agent,
- a system relying on unknown categorization rules, which appear to be random.

The figures appear consistent with the implicit hypotheses: the "manual" system rates high in precision (88%) but rather low in recall (74%). The manual run validates our "topical signatures" approach, it also shows that explicit approaches fail at capturing part of the knowledge used by experts in a text categorization task.
The "random" system rates moderately in recall (around 50%: 52% in average over 10 runs) and rates rather well in precision (67%). This would appear surprising, should one not bear in mind the essential property of random processes, together with the binary nature of the selection decision evaluated here. In other words, faced with 2 possibilities (select/discard), the random system performs exactly as expected, as it would have for a coin-flipping output prediction simulation: it gives around 50% correct answers[27]. Still, the "rules" used in the decision selection process can not be traced back,

[23] Approximately 3 man-months.
[24] E.g. complements for the verb "céder" are not specified, while it can be found in phrases such as "céder sa filiale" but also in "céder à ses avances" which is not related to topic19.
[25] The rates for the random system were averaged over 10 runs, given the random nature of the system tested.
[26] Precision = Nb. of matches / Nb. of responses,
Recall = Nb. of matches / Total of expected responses.
[27] Respectively, incorrect.

while tracing and debugging capabilities are inherent to symbolic approaches. In other words, the "random" system would appear to perform surprisingly well in regard to its cost[28] if not for its opaque way of categorizing text, its fickle selection decision[29] and its "black box" nature. The random system also shows the relative efficiency of our approach: in the classical evaluation framework described, relying on external evidence (recall and precision rates), almost 50% of the problem are covered without any "intelligence" whatsoever.

Finally, the figures computed for the LIZARD run show the substantial gain attainable by integrating both common and specific knowledge in the text categorization process. The LIZARD approach thus provides the field of information filtering with a seemingly viable and efficient approach, even though complementary experiments should take place in order to evaluate more precisely the gain of the local grammars expansion approach.

5.4. Conclusion and Perspectives

In this paper, we have shown how the field of Information Retrieval, i.e. Information Filtering, could benefit from a symbolic approach to text classification tasks such as "batch filtering". Moreover, we have shown that real-life data, consisting of a corpus of short specialized texts (financial news), did not fit well in the frame of the international TREC evaluation conferences, providing gigabytes of textual data and evaluation procedures that favor data-intensive (machine-learning) approaches. Therefore, in order to evaluate the approach described, we have presented a procedure which compares our system's performance to a manual and a random one, rather than figures based on the official "utility" measures for text filtering systems' evaluation.

We have tried to show how the integration of hybrid resources - corpus-driven (specialized) and corpus independent (general) ones - in the design process of automatic categorization rules expressed as Finite-State Transducers could yield better results than rules designed solely by hand. The figures presented show the performance of LIZARD, a system based on interactively expanded symbolic rules for automatic text filtering, which rates high in recall and compares equally well in precision to a manual approach.

The experiments described in this paper have also shown us that even though human operators' expertise is crucial to the IF activity, it is not less prone to subjectivity than other categorization tasks. Therefore, any attempt to compare the performance of a given IF system to a human reference should take into consideration the problem of the inherent subjectivity attached to the IF/categorization task. In other words, we plan to follow qualitative (glass-box) evaluation procedures in the future, rather than purely quantitative (black-box) ones.

[28] Easy implementation, low space/memory load.

[29] The retrieved document set varies with every run.

6. References

Abney, S. (1996). Partial Parsing via Finite-State Cascades. *Proceedings of the ESSLLI'96 Robust Parsing Workshop*.

Balvet, A. Meunier, F. Poibeau, T. Viard, D. Vichot, F. Wolinski, F. (2001). Filtrage de Documents et Grammaires Locales : le Projet CORAIL. *Actes du congrès de l'ISKO (International Society for Knowledge Organisation)*, 5-6 juillet 2001. Université de Nanterre-Paris X.

Bizouard, S. (2001). *Évaluation d'Outils d'Acquisition de Ressources Linguistiques pour l'Extraction*. Mémoire de DESS en Ingénierie Multilingue. CRIM, INALCO.

Dutoit D., (2000). *Quelques Opérations Texte → Sens et Sens → Texte Utilisant une Sémantique Linguistique Universaliste Apriorique*. Ph.D. dissertation. Caen University.

Grefenstette, G. (1996). Light Parsing as Finite-State Filtering. *Workshop on Extended Finite State Models of Language, ECAI'96*.

Gross, M. (1975). *Méthodes en Syntaxe*. Paris, Hermann.

Harman, D. (1993). Overview of the First Text REtrieval Conference (TREC-1). *NIST Special Publications*. Gaithersburg, MD.

Harris, Z.S. (1968). *Mathematical Structures of Language*. Interscience Publishers, John Wiley & Sons.

Lewis D., Hill M. (1995). The TREC-4 filtering track. *NIST Special Publications*. Gaithersburg, MD.

Luhn, H.P. (1958). A Business Intelligence System. *IBM Journal of Research and Development*, Vol 2(4), pp. 314-319.

Riloff, E. (1994). *Information Extraction as a Basis for Portable Text Classification Systems*. Ph.D. dissertation. University of Massachussets Amherst.

Robertson, S. & Hull, D.A. (2001). The TREC-9 Filtering Track Final Report. *NIST Special Publications*. Gaithersburg, MD.

Roche, E. (1993). *Analyse Syntaxique Transformationnelle du Français par Transducteurs et Lexique-Grammaire*. Ph.D. dissertation. Paris VII University.

Silberztein, M. (1993). *Le Système INTEX, Dictionnaires Électroniques et Analyse Automatique des Textes*. Paris, Masson.

Silberztein, M. (1999). Traitement des Expressions Figées avec INTEX. *Linguisticae Investigationes*, tome XXII, pp. 425-449. John Benjamins Publishing Company.

Voorhees, E. & Harman, D. (2001). Overview of the Ninth Text REtrieval Conference (TREC-9). *NIST Special Publications*. Gaithersburg, MD.

Lexically-Based Terminology Structuring: a Feasibility Study

Natalia Grabar, Pierre Zweigenbaum

DIAM — STIM/DSI, Assistance Publique – Hôpitaux de Paris
& Département de Biomathématiques, Université Paris 6
{ngr,pz}@biomath.jussieu.fr

Abstract

Terminology structuring has been the subject of much work in the context of terms extracted from corpora: given a set of terms, obtained from an existing resource or extracted from a corpus, identifying hierarchical (or other types of) relations between these terms. The present work aims at assessing the feasibility of such structuring by studying it on an existing, hierarchically structured terminology. For the evaluation of the results, we measure recall and precision metrics, taking two different views on the task: relation recovery and term placement. Our overall goal is to test various structuring methods proposed in the literature and to check how they fare on this task. The specific goal in the present phase of our work, which we report here, is focussed on lexical methods that match terms on the basis on their content words, taking morphological variants into account. We describe experiments performed on the French version of the US National Library of Medicine MeSH thesaurus. This method proposes correct term placement for up to 26% of the MeSH concepts, and its precision can reach 58%.

1. Background

Terminology structuring, *i.e.*, organizing a set of terms through semantic relations, is one of the difficult issues that have to be addressed when building terminological resources. These relations include subsumption or hyperonymy (the *is-a* relation), meronymy (*part-of* and its variants), as well as other, diverse relations, sometimes called 'transversal' (*e.g.*, *cause*, or the general *see also*).

Various methods have been proposed to discover relations between terms (see (Jacquemin and Bourigault, 2002) for a review). We divide them into *internal* and *external* methods, in the same way as (McDonald, 1993) for proper names. Internal methods look at the constituency of terms, and compare terms based on the words they contain. Term matching can rely directly on raw word forms (Bodenreider et al., 2001), on morphological variants (Jacquemin and Tzoukermann, 1999), on syntactic structure (Bourigault, 1994; Jacquemin and Tzoukermann, 1999) or on semantic variants (synonyms, hyperonyms, etc.) (Hamon et al., 1998). External methods take advantage of the context in which terms occur: they examine the behavior of terms in corpora. Distributional methods group terms that occur in similar contexts (Grefenstette, 1994). The detection of appropriate syntactic patterns of cooccurrence is another method to uncover relations between terms in corpora (Hearst, 1992; Séguéla and Aussenac, 1999).

The present work aims at assessing the feasibility of such structuring by studying it on an existing, hierarchically structured terminology. Ignoring this existing structure and starting from the set of its terms, we attempt to discover hierarchical term to term links and compare them with the preexisting relations.

Our aim consists in testing various structuring methods proposed in the literature and checking how they fare on this task. The specific goal in the present phase of our work, which we report here, is focussed on lexical methods that match terms on the basis on their content words, taking morphological variants into account.

After the presentation of the data we used in our experiments, we present methods for generating hierarchical links between terms through the study of lexical inclusion and for evaluating their quality with appropriate recall and precision metrics. We then detail and discuss the results obtained in this evaluation.

2. Material

In this experiment we used an existing hierarchically structured thesaurus, a 'stop word' list, and morphological knowledge.

2.1. The MeSH biomedical thesaurus

The Medical Subject Headings (MeSH, MeS (2001)) is one of the main international medical terminologies (see, *e.g.*, Cimino (1996) for a presentation of medical terminologies).

It is a thesaurus specifically designed for information retrieval in the biomedical domain. It is used to index the international biomedical literature in the Medline bibliographic database. The French version of the MeSH (INS, 2000) contains a translation of these terms (19,638 terms) plus synonyms. It happens to be written in unaccented, uppercase letters.

As many other medical terminologies, the MeSH has a hierarchical structure: 'narrower' concepts (children) are related to 'broader' concepts (parents). The MeSH specifically displays a rich, polyhierarchical structure: each concept may have several parents. In total, the MeSH contains 26,094 direct child-to-parent links and (under transitive closure) 95,815 direct or indirect child-to-ancestor links.

2.2. Stop word list

The aim of using a 'stop word' list is to remove from term comparison very frequent words which are considered not to be content-bearing, hence 'non-significant' for terminology structuring.

The stop word list used in this experiment is a short one (15 word forms). It contains the few grammatical words which occur frequently in MeSH terms, articles and prepositions:

au, aux, d', de, des, du, en, et, l', la, le, les, ses, un, une

2.3. Morphological knowledge

Previous work has acknowledged morphology as an important area of medical language processing and medical information indexing (Pacak et al., 1980; Wingert et al., 1989; Grabar et al., 2002) and of term variant extraction (Jacquemin and Tzoukermann, 1999). In this work, we apply morphological knowledge to the terminology structuring task.

Three types of morphological relations are classically considered:

- *Inflection* produces the various forms of a same word such as plural, feminine or the multiple forms of a verb according to person, tense, etc.: *intervention – interventions*, *acid – acids*. The parts of speech of a lemma and its inflected forms are the same. Reducing an inflected form to its lemma is called lemmatization.

- *Derivation* is used to obtain, *e.g.*, the adjectival form of a noun (noun *aorta* ↔ adjective *aortic*, verb *intervene* ↔ noun *intervention*, adjective*human* ↔ adverb *humanely*). Derivation often deals with words of different parts of speech. Reducing a derived word to its base word is called stemming.

- *Compounding* combines several radicals, here often of greek or latin origin, to obtain complex words (*e.g.*, *aorta* + *coronary* yields *aortocoronary*).

The morphological knowledge we used consists of {*lemma*, *derived or inflected form*} pairs of word forms where the first is the 'normalized' form and the second a 'variant' form. The general principle is that both forms of such a pair have similar meaning.

In this work we rely on inflectional knowledge and derivations that do not change word meaning. We have left compounding aside for the time being, since the words it relates may have distant meanings.

2.3.1. Inflectional knowledge

For inflection, we have two lexicons of such word pairs. The first one is based on a general lexicon (ABU, `abu.cnam.fr/DICO`) which we have augmented with pairs obtained from medical corpora processed through a tagger/lemmatizer (in cardiology, hematology, intensive care, and drug monographs): it totals 219,759 pairs (where the inflected form is different from the lemma). The second lexicon is the result of applying rules acquired in previous work (Zweigenbaum et al., 2001) from two other medical terminologies (ICD-10 and SNOMED) to the vocabulary in the MeSH, ICD-10 and SNOMED (total: 2,889 pairs).

2.3.2. Derivational knowledge

For derivation, we also used resources from (Zweigenbaum et al., 2001) which, once combined with inflection pairs, result in 4,517 pairs.

These morphological resources will still need to be improved; but we believe that the results should not vary much from what is present here.

3. Methods

The present work induces hierarchical relations between terms when the constituent words of one term lexically include those of the second term (section 3.1.). We evaluate these relations by comparing them with the pre-existing relations, computing precision and recall both for links and concepts (section 3.2.).

3.1. Lexical Inclusion

The method we use here for inducing of hierarchical relations between terms is basically a test of *lexical inclusion*: we check whether a term P (*parent*) is 'included' in another term C (*child*). We assume that this type of inclusion is a clue of a hierarchical relation between terms, as in the following example: *acides gras* / *acides gras indispensables* (*fatty acids* / *fatty acids, essential*).

To detect this type of relation, we test whether all the content words of P occur in C. We test this on segmented terms with a gradually increasing normalization on word forms:

- basic normalization: conversion to lower case, removal of punctuation, of numbers and of 'stop words' (introduced in section 2.2.);

- normalization with morphological ressources (see section 2.3.): lemmatization (with the two alternative inflectional lexicons) and stemming with a derivational lexicon.

Terms are indexed by their words to speed up the computation of term inclusion over all term pairs of the whole MeSH thesaurus. When these normalizations are applied, terms are indexed by their normalized words: we assume that P is lexically included in C iff all normalized words in P occur in C.

3.2. Evaluation

We evaluated the results obtained with this approach by comparing them with the original structure in the MeSH. We considered two methods to evaluate this terminology structuring task:

- the first method is interested in the number of links found, and compares these links with those originally present in the MeSH thesaurus: do we obtain all the links that pre-exist in the MeSH?

- the second method considers the positioning of individual MeSH concepts (terms) in the hierarchical structure of the thesaurus: can we place each concept in at least one suitable position in the emerging hierarchy?

For both methods, we compute recall and precision metrics. The recall metric allows us to analyze the completeness of the results and to know whether all the expected links are induced and concepts positioned. The precision metric evaluates the correctness of induced results.

The recall and precision measures computed here have two versions:

- strict (only the links to direct parents of a given concept are considered satisfactory), and
- tolerant (a link to any ancestor is considered as correct).

We also tested a mixed scheme: the weight given to each link depends on the distance between the two concepts related with this link in the original hierarchical structure of the MeSH: the more distant these concepts, the lower the weight the induced link obtains. However, since the mixed scheme results are not very different from the tolerant one, we do not present them here.

The lexical inclusion methods and the evaluation procedure were implemented as Perl5 scripts.

4. Results

4.1. Lexical inclusions obtained

The method described in section 3.1. has been applied to the flat list of 19,638 terms ('main headings') of the MeSH thesaurus. The gradualy increasing normalizations we applied to this list of terms allow us to induce an increasing number of hierarchical links between these terms.

In table 1 we show quantitative results for the relations induced with the analysis of lexical inclusions and obtained with each type of morphological normalization tested. The first column introduces the types of normalization. The *raw* results were obtained with no morphological normalization. The *lem-gen* results were obtained with application of inflection pairs compiled from a general lexicon, and *lem-med* results with inflectional pairs acquired from medical terminologies (see section 2.3.1.). The *lem-stem-med* results correspond to the normalization done with derivational pairs (see section 2.3.2.). The basic normalization (conversion to lower case, removal of punctuation, numbers and stop words) is performed in all cases. The second column presents the number of links induced with each of the normalization methods tested. The third column recalls the number of hierarchical relations in the MeSH.

Type of normalization	Number of links	Reference
raw	9,189	95,815
lem-gen	12,963	95,815
lem-med	11,627	95,815
lem-stem-med	15,942	95,815

Table 1: Quantification of induced relations berween analyzed terms.

In table 2 we present the same type of information for the placement of terms. The second column contains the number of terms which have been linked with our methods. This number corresponds to the number of concepts that can be linked in the 'structured' terminology we induced. The third column recalls the number of linked terms in the MeSH hierarchy.

As expected, the number of links induced between terms increases when applying inflectional normalization and even more with derivational normalization. Inflectional knowledge compiled from the general lexicon (*lem-gen*) allows to link more terms than that only obtained from specialized terminologies (*lem-med*): 12,963 *vs* 11,627 links. But for the positionining of terms, we obtain better covering of terms when using specialized morphological knowledge (*lem-med*) than when using morphological knowledge from general lexicon (*lem-gen*): 10,949 *vs* 10,261 terms.

Type of normalization	Number of terms	Reference
raw	9,126	19,638
lem-gen	10,261	19,638
lem-med	10,949	19,638
lem-stem-med	11,752	19,638

Table 2: Quantification of positioned terms.

Lemmatization can be ambiguous when an inflected form can be obtained from several lemmas (*e.g.*, *souris* $\rightarrow$ *souris/N* (*mouse*) and *sourire/V* (*to smile*)). In that case, we have adopted a brute force approach which merges the two corresponding morphological families and chooses one lemma as unique representative for both.

Table 3 shows examples of lexically included terms which we obtained with this method. For each type of normalization, it shown pairs *parent* / *child* corresponding to direct, then indirect relations in the original MeSH structure.

4.2. Evaluation of these lexical inclusions

In section 3.2. we presented the methods designed to evaluate the structuring results we obtain with a lexical inclusion analysis of terms. These methods allow us to evaluate recall and precision metrics for both relations between terms and term positioning. In all the cases we take into account the nature of induced links (direct or indirect ones) by testing both strict and tolerant variants. The correctness of induced results is computed by comparing these results with the original MeSH structure.

Table 4 shows the evaluation results for the links, and table 5 for concept (term) placement.

The second column in table 4 contains the number of direct and indirect correct links; the third column shows the number of incorrect links (links which do not exist in the MeSH). The *Recall, direct* column presents the recall R_d of the direct links found d (weighted by the number of direct links $D = 26,094$ in the MeSH – see section 2.1.); the *Recall, all* column presents the recall R_a of all the links (weighted by the total number of links $D + I = 95,815$ in the MeSH):

$$R_d = \frac{d}{D}; R_a = \frac{d+i}{D+I}$$

The last column of this table presents the evaluation of the precision metric, taking into account both strict and tolerant appoaches; if d is the number of direct links found, i the number of indirect links found, and n the number of non-MeSH links found, strict precision P_s and tolerant precision P_t are:

$$P_s = \frac{d}{d+i+n}; P_t = \frac{d+i}{d+i+n}$$

Type of normalization	Parent P	Child C
raw $_{direct}$	accouchement	accouchement provoque
	delivery	*labor, induced*
raw $_{indirect}$	acides gras	acides gras indispensables
	fatty acids	*fatty acids, essential*
lem-gen $_{direct}$	intervention chirurgicale	interventions chirurgicales obstetricales
	surgical procedures, operative	*obstetric surgical procedures*
lem-gen $_{indirect}$	intervention chirurgicale	interventions chirurgicales voies biliaires
	surgical procedures, operative	*biliary tract surgical procedures*
lem-med $_{direct}$	agents adrenergiques	inhibiteurs captage agent adrenergique
	adrenergic agents	*adrenergic uptake inhibitors*
lem-med $_{indirect}$	chromosomes humains	chromosome humain 21
	chromosomes, human	*chromosomes, human, pair 21*
lem-stem-med $_{direct}$	aberration chromosomique, anomalies	aberrations chromosomes sexuels, anomalies
	chromosome abnormalities	*sex chromosome abnormalities*
lem-stem-med $_{indirect}$	eosinophilie	poumon eosinophile
	eosinophilia	*pulmonary eosinophilia*

Table 3: Examples of correct, lexically induced MeSH terms, and their English translations. Indirect means that the MeSH includes a path of length > 1 from the parent to the child.

Normalization	Correct links		Incorrect	Recall (%)		Precision (%)	
	direct	indirect	(non MeSH)	direct	all	strict	tolerant
raw	2688	1266	5235	10.3	4.1	29.3	43.0
lem-gen	3058	1779	6790	11.7	5.0	26.3	41.6
lem-med	3451	2171	7341	13.2	5.9	26.6	43.4
lem-stem-med	3580	2316	10046	13.7	6.2	22.5	37.0

Table 4: Recall and precision of lexically-induced links.

Normalization	Recall: correct advices / # MeSH nodes			Precision: correct advices / # advices		
	strict (%)	tolerant (%)	MeSH nodes	strict (%)	tolerant (%)	nodes linked
raw	10	18	19543	27	52	6969
lem-gen	10	23	19543	24	55	8078
lem-med	10	26	19543	24	58	8644
lem-stem-med	9	26	19543	18	55	9398

Table 5: Recall and precision of lexically-induced node placement advices.

The recall of links increases when applying more complete morphological knowledge (inflection then derivation). And, not surprisingly, we notice that the recall of relations between terms obtained with morphological knowledge acquired from medical terminologies (*lem-med*, *lem-stem-med*) is higher (13.2 and 13.7%) than the recall corresponding to the use of the morphological knowledge compiled from the general lexicon (*lem-gen*, 11.7%).

The evolution of precision is opposite: injection of more extensive morphological knowledge (derivation *vs* inflection) leads to taking more 'risks' for generating links between terms: *raw* results precision is 29.3% *vs* 22.5% for *lem-stem-med* precision.

When accepting both direct and indirect links (tolerant approach), the precision measure obtained is higher than when only direct links are considered (strict approach). For instance, with raw normalization, the tolerant approach gives a precision of 43.0% and the strict approach 29.3%. For the *lem-stem-med* normalization the tolerant precision is 37.0% and the strict precision is 22.5%.

Depending on the normalization and the weighting scheme, up to 29.3% of the links found are correct, and up to 13.7% of the direct MeSH links are found by lexical inclusion.

Up to 26% of the concepts are correctly placed under their ancestors; and the term positioning advices are correct in up to 58% of the cases.

5. Discussion

We presented in this paper an experiment of terminology structuring. We tested here some 'internal' methods for this task, which consist in the analysis of the lexical inclusions of terms. We consider that a term P is lexically included in a term C iff all words of P occur in C, and that this is a clue of its being a parent (ancestor) of C. To help this analysis we apply normalizations, both basic and making use of morphological knowledge.

Whereas raw lexical inclusion detects directly attainable relations between terms by matching identical words in these terms, lemmatization adds flexibility with inflectional variants. Morphological stemming allows to link terms which contain words that are graphically different but have a very close meaning. This allows to obtain hierarchical dependencies between terms that are more based on the 'meanings' of these terms. These semantic similarities are detected through the morphological analysis we apply.

To assess the induced results we compare them with the original structure of the MeSH. We evaluate both the induced links and the placed terms. Depending on the normalization and the weighting scheme, up to 29.3% of the links found are correct, and up to 13.7% of the direct MeSH links are found by lexical inclusion. Up to 26% of the terms are correctly placed under their ancestors; and the placement advices are correct in up to 58% of the cases.

The only expected and evaluated type of relation is the hierarchical one, as exists in the MeSH thesaurus. But we assume that the methods applied also allow to induce other types of relations, and maybe other hierarchical relations, which are not in the original MeSH hierarchy. Some 'new' relations can be found, for instance, in the incorrect ('extra'-) relations we induced. These additional relations have to be analysed in a detailed way to better evaluate the results obtained with these simple methods.

In summary, lexical inclusion caters for a non-negligible number of the hierarchical concept organization in the MeSH thesaurus; and the use of morphological knowledge, mainly for lemmatization, significantly increases this proportion. As could have been hypothesized, trying to place a concept at one position in the hierarchy is more successful than finding all the links from this concept to its parents in a polyhierarchical terminology.

A simple analysis of lexical inclusions shows that in many cases a hierarchical dependency between (medical) terms can be detected and allows to obtain an important number of hierarchical relations between these terms. This information is useful when dealing with the terminology structuring task.

To detect and evaluate more relations between terms, other methods for terminology structuring may be applied, such as those presented in section 1. We plan to test them in the same context as the morphological experiments presented here.

6. References

Olivier Bodenreider, Anita Burgun, and Thomas C. Rindflesch. 2001. Lexically-suggested hyponymic relations among medical terms and their representation in the UMLS. In URI INIST CNRS, editor, *TIA'2001 Terminologie et Intelligence artificielle*, pages 11–21, Nancy.

Didier Bourigault. 1994. Extraction et structuration automatiques de terminologie pour l'aide à l'acquisition de connaissances à partir de textes. In *RFIA'94*, pages 1123–1132. AFCET.

James J Cimino. 1996. Coding systems in health care. In Jan H. van Bemmel and Alexa T. McCray, editors, *Yearbook of Medical Informatics '95 — The Computer-based Patient Record*, pages 71–85. Schattauer, Stuttgart.

Natalia Grabar, Pierre Zweigenbaum, Lina Soualmia, and Stéfan J. Darmoni. 2002. A study of the adequacy of user and indexing vocabularies in natural language queries to a MeSH-indexed health gateway. *J Am Med Inform Assoc*, 8(suppl). *Submitted*.

Gregory Grefenstette. 1994. *Explorations in Automatic Thesaurus Discovery*. Natural Language Processing and Machine Translation. Kluwer Academic Publishers, London.

Thierry Hamon, Adeline Nazarenko, and Cécile Gros. 1998. A step towards the detection of semantic variants of terms in technical documents. In Christian Boitet, editor, *Proceedings of the 17 th COLING*, pages 498–504, Montréal, Canada, 10–14 August.

Marti A. Hearst. 1992. Automatic acquisition of hyponyms from large text corpora. In Antonio Zampolli, editor, *Proc 14 th COLING*, pages 539–545, Nantes, France, 23–28 July.

Institut National de la Santé et de la Recherche Médicale, Paris, 2000. *Thésaurus Biomédical Français/Anglais*.

Christian Jacquemin and Didier Bourigault. 2002. Term extraction and automatic indexing. In Ruslan Mitkov, editor, *Handbook of Computational Linguistics*. Oxford University Press, Oxford. *To appear*.

Christian Jacquemin and Évelyne Tzoukermann. 1999. NLP for term variant extraction: A synergy of morphology, lexicon, and syntax. In Tomek Strzalkowski, editor, *Natural language information retrieval*, chapter 2, pages 25–74. Kluwer Academic Publishers, Dordrecht & Boston.

David D. McDonald. 1993. Internal and external evidence in the identification and semantic categorization of proper names. In Branimir Boguraev and James Pustejovsky, editors, *Corpus Processing for Lexical Acquisition*, pages 61–76. MIT Press, Cambridge, MA.

2001. Medical Subject Headings. WWW page `http://www.nlm.nih.gov/mesh/meshhome.html`, National Library of Medicine, Bethesda, Maryland.

M. G. Pacak, L. M. Norton, and G. S. Dunham. 1980. Morphosemantic analysis of -ITIS forms in medical language. *Methods Inf Med*, 19:99–105.

Patrick Séguéla and Nathalie Aussenac. 1999. Extraction de relations sémantiques entre termes et enrichissement de modèles du domaine. In Régine Teulier, editor, *Actes de IC'99*, June.

F. Wingert, David Rothwell, and Roger A Côté. 1989. Automated indexing into SNOMED and ICD. In Jean Raoul Scherrer, Roger A. Côté, and Salah H. Mandil, editors, *Computerised Natural Medical Language Processing for Knowledge Engineering*, pages 201–239. North-Holland, Amsterdam.

Pierre Zweigenbaum, Stéfan J. Darmoni, and Natalia Grabar. 2001. The contribution of morphological knowledge to French MeSH mapping for information retrieval. *J Am Med Inform Assoc*, 8(suppl):796–800.

Query Expansion by a Contextual Use of Classes of Nouns

Gaël de Chalendar and Brigitte Grau

LIR group – LIMSI (CNRS)
BP 133 91403 Orsay Cedex
Gael.de.Chalendar@limsi.fr, Brigitte.Grau@limsi.fr

Abstract

We developed a system, SVETLAN', dedicated to the acquisition of classes of semantically close nouns from texts. We aim at constructing a structured lexicon for the general language, that is not for representing a specialized domain. Thus, texts are open-domain newspaper articles. The acquisition is based on a distributional method that groups the nouns that are related to a same verb with a same functional role. However, in order to deal with polysemy, classes are learned in context: they are built from text segments related to a same semantic domain. For that, we use results of ROSA, a system that clusters automatically segmented texts in order to build semantic domain defined by sets of weighted words. We will show how these classes can be used to expand queries, in comparison with an expansion realized by using WordNet.

1. Introduction

Information Retrieval systems often require semantic knowledge to improve their results. However, one can ask, "what type of semantics?". According to the application, it may differ. It can be only synonymous, or semantically close words, or words belonging to a same domain, either specific or general. One conclusion is that it is necessary to be able to bring together words with close signification. Moreover this gathering has to be done in a well defined context in order to take into account multiple meanings of words. For example, in the context of nuclear plants, one confronted to the sentences: *"… started to replace the fuel rods…"*, *"… started to replace the combustible of the reactor…"* and *"… to replace the films and the batteries of the camera… "*, should join together the words *combustible* and *rods* but should put aside the word *film*.

We are interested in robust applications aimed to cope with every domain, opposed to domain specialized systems. Those systems often use preexistent knowledge to find synonyms or related words but it remains difficult to select the right information. For instance, the noun *care* has 6 registered meanings in WordNet 1.6 (Fellbaum, 1998). If we are interested in medicine practice, we do not want to retrieve documents that use the word *care* with its 4th sense ("*a cause to feeling concerned*"), but maybe only those that use it with its first sense: "*the work of caring for […] someone […]*".

Our conclusion after these statements is that a general ontology or classification seeking for universality is an utopia and principally because of the word polysemy. So, the terminological aspect of general language has to be modeled by multiple overlapping classifications. The question we have to ask is then: "how can these classifications be acquired". We make three hypotheses. Firstly, at least a part of the semantic knowledge is encoded in the texts. Secondly, a part of this text-encoded knowledge can be automatically extracted and lastly, this extraction will be feasible only if semantics is considered in fine-grained contexts.

Work has been done during previous decades on general language but the encoding was mainly manual, as for scripts of Schank (Schank, 1982) that were defined for storing semantico-pragmatic representations of everyday situations. It has been proved very difficult to extend the scripts beyond the first few ones. Another example of manually encoded semantic knowledge is CYC (Lenat, 1986) that is supposed to be a universal semantic knowledge base. In reality, CYC has to be manually tuned in each application it is used in.

On the contrary, various methods have been used with success to acquire semantic knowledge on specialized domains: cooccurrences statistics (Zernik, 1991), distributional approaches following Harris ideas (Harris, 1968), classification techniques (Agarwal, 1995), linguistic indices (Roark & Charniak, 1998), etc. Our interrogation was on the possibility of adapting these successful techniques to general language. Our proposition is to determine automatically thematic domains and to apply a classical distributional method on texts belonging to a same domain. This approach allows our system to form classes of semantically close words.

The idea behind the distributional method is that the usage of a verb is directed by its sub-categorization frame. This frame specifies for example that the subject of the verb should be an instance of a particular concept. The set of real objects referred to by the words that are subjects of the verb in a particular domain represent this concept by extension. Thus, a description of this extension is the set of words used to refer to these objects. These sets of words are the semantic classes made by our system, SVETLAN' (Chalendar & Grau, 2000).

We will show how these classes can be used to expand queries, in comparison with an expansion realized by using WordNet.

2. Overview of the system

Input data of SVETLAN' (see Fig. 1) are semantic domains with the thematic units (TUs) that have given birth to them. Domains are sets of weighted words, relevant to represent a same specific topic. These domains are automatically learned by ROSA that aggregates similar thematic units, made of sets of words. TUs are built by a topic segmentation process relying on lexical cohesion. It processes texts such as newspaper articles.

The first step of SVETLAN' consists of a syntactic parsing of the corpus in order to produce the structured thematic units (STUs) corresponding to each TU. STUs are constituted by a set of triplets - a verb, the head noun of a phrase and its syntactic role - extracted from the parser results. The STUs related to a same semantic domain are aggregated altogether to learn a structured domain. Aggregation leads to group nouns playing the same syntactic role with a verb in order to form classes.

As these aggregations are made within STUs belonging to a same domain, classes are context sensitive, which ensures a better homogeneity. A filtering step, based on the weights of the words in their domain allows the system to eliminate nouns from classes when they are not very relevant in this context.

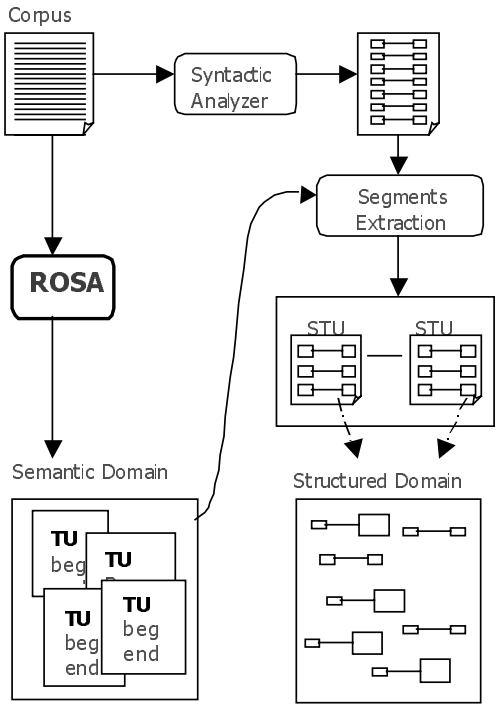


Figure 1: Schemata of Structured Domain learning

3. The ROSA system

We only give here a brief overview of the system that is made of two modules, SEGCOHLEX and SEGAPSITH. It is described more precisely in (Ferret & Grau, 1998). ROSA incrementally builds topic representations, made of weighted words, from discourse segments delimited by SEGCOHLEX (Ferret, 1998). It works without any *a priori* classification or hand-coded pieces of knowledge. Processed texts are typically newspaper articles coming from the *Los Angeles Times*. They are pre-processed to only keep their lemmatized content words (adjectives, single or compound nouns and verbs).

The topic segmentation implemented by SEGCOHLEX is based on a large collocation network, built from 24 months of the *Los Angeles Times* newspaper, where a link between two words aims at capturing semantic and pragmatic relations between them. The strength of such a link is evaluated by the mutual information between its two words. The segmentation process relies on these links for computing a cohesion value for each position of a text. It assumes that a discourse segment is a part of text whose words refer to the same topic, that is, words are strongly linked to each other in the collocation network and yield a high cohesion value. On the contrary, low cohesion values indicate topic shifts. After delimiting segments by an automatic analysis of the cohesion graph, only highly cohesive segments, named thematic units (TUs), are kept to learn topic representations. This segmentation method entails a text to be decomposed in small thematic units, whose size is equivalent to a paragraph. Because discourse segments, even related to the same topic, often develop different points of view of this topic, we enrich the particular description given by a text. We add to the TUs those words of the collocation network that are particularly linked to the words found in the corresponding segment.

Words	occ.	weight
examining judge	58	0.501
police custody	50	0.442
public property	46	0.428
charging	49	0.421
to imprison	45	0.417
court of criminal appeal	47	0.412
receiving stolen goods	42	0.397
to presume	45	0.382
criminal investigation department	42	0.381
fraud	42	0.381

Table 1: The most representative words of a domain about justice

Learning a complete description of a topic consists of merging all successive points of view, i.e. similar TUs, into a single memorized thematic unit, called a semantic domain. Each aggregation of a new TU increases the system's knowledge about one topic by reinforcing recurrent words and adding new ones. Weights on words represent the importance of each word relative to the topic and are computed from the number of occurrences of these words in the TUs (see Table 1 for an example of a domain). This method, implemented in SEGAPSITH, leads to learn specific topic representations as opposed to (Lin, 1997) for example whose method builds general topic descriptions as for economy, sport, etc.

4. Semantic Domain Structuring

Semantic domains are similar to classes formed by (Zernik, 1991). SVETLAN' purpose is then to delimit small classes inside these domains, and to associate them to the verbs they define, as it is made in distributional approaches (Faure & Nedellec, 1998) (Pereira & al., 1993). A class is defined by those nouns which play a same role relative to a same verb and that are supposed to be connected by a strong semantic link. Thus, even if they do not denote a same object, the objects denoted by them play a similar role in the tight context defined by the semantic domain.

4.1. Formation of The Structured Thematic Units

A syntactic parser processes texts in order to find the verbs and their arguments. For English, we used the link grammar (Grinberg & al., 1995). The system extracts all the triplets found by the analyzer, constituted by a verb, a syntactic relation and the head noun of the noun phrase. Relations are subject, direct and indirect objects, the preposition that introduces a prepositional phrase. The link grammar only gives one interpretation of the sentence.

After parsing the texts, SVETLAN' groups the triplets relatively to the delimited thematic units. So, we define a structured thematic unit as a set of <*Verb→syntactic*

relation→Noun> structures, i.e. a syntactic relation instantiated with a verb and a noun. We will refer to these structures as instantiated syntactic relations.

4.2. Aggregation

Structured thematic units related to a same domain are aggregated altogether to form the structured domains. Aggregating a structured thematic unit within a structured domain consists of:

- aggregating the instantiated syntactic relations that contain the same relation and the same verb, i.e. associating a set of words to an argument of a verb;
- adding new instantiated syntactic relations, i.e. adding new verbs with their arguments made of a syntactic relation and the lemmatized form of a noun.

Nouns are not weighted inside a class; they only keep the weight they had in their semantic domain. Thus, the criterion to define a class is that words appear with a same verb, in similar contexts. The similarity of contexts is a lexical similarity computed on the whole domain.

5. Results

Classes are built according to two levels of contextual use of the words: a global similarity of the thematic contexts and a local relevance inside a domain we added to discard irrelevant words. In order to illustrate the effect of topic similarity when building classes, we show in Table 2 a class regrouping all the direct objects found for the verb *to replace* in the whole corpus. We can see that there is no semantic proximity between those nouns. When the class is formed, for the same verb, inside a nuclear domain, the class is then homogeneous. So, even general verbs, as *to replace* (it is possible to replace a lot of things), are relevant criteria to group nouns when their appear in similar thematic units.

to replace	*object*	text, constitution, trousers, combustible, law, dinar, rod, film, circulation, judge, season, device, parliament, battalion, police, president, treaty
to replace	*object*	combustible, rod

Table 2: The effect of the thematic context on the kind of classes

However, classes of nouns contain a lot of words that disturb their homogeneity. These words often belong to parts of the different TUs at the origin of the semantic domain that are not very related to the described topic. They correspond to meanings of words scarcely used in the current context. As these words are weekly weighted in the corresponding domains, the data can be filtered: each noun that possesses a weight lower than a threshold is removed from the class. By this selection, we reinforce learning classes of words according to their contextual use.

to establish	*object*	base, zone
to answer	*to*	document, question, list
~~**to establish**~~	~~*object*~~	~~base, zone~~
to answer	*to*	document, question, ~~list~~

Table 3: Two filtered classes in a domain about nuclear weapons

Table 3 shows two aggregated links obtained without filtering in its upper part and the filtered counterparts in its lower part. The link for the verb '*to establish*' has been completely removed while the link of the verb '*to answer*' with the preposition 'to' has been reduced by the removing of '*list*'.

Table 4 shows some examples of classes obtained by SVETLAN'. Even when verbs are polysemous, which is the case for several verbs in the examples, the domain membership constraint leads the system to build relevant classes. We also can see that the various syntactic relations are relevant criteria to gather semantically linked words.

Domain	Verb	Relation	Class
War	to qualify	Direct Object	president, leader
Food assistance	to take refuge	Into	country, region
Tour de France	to cover	Direct Object	stage, tour
Sport	to face	In	match, final
Economy	to release	Direct Object	million, billion
Festival cinema	to tell	Subject	film-maker, film
Conflict Croatia	to resume	Direct Object	negotiation, discussion
Economy	to reduce	Direct Object	surplus, deficit

Table 4: Examples of noun classes

SVETLAN' originality relies on the constitution of classes given with their context of reference. As a context is explicitly defined by a set of words, it gives indices, when finding a word in a text or a sentence, to choose a class or another, and so to obtain neighbor words. We will show the application of this property when expanding a query.

Verb	Relation	Class
To accuse	Subject	Indictment, prosecutor
	By	Prosecutor, jury
To make	Subject	Prosecutor, indictment
	Direct Object	Jury, prosecutor
To show	Subject	Juror, defendant
	Direct Object	Jury, scheme
To tell	Subject	Magistrate, informant
	Direct Object	Juror, jury
To give	Direct Object	Sentence, prosecutor, trial
	From	Sentence, prosecution
	To	Jury, defendant

Table 5: Example of verbs with classes defining their arguments in a domain about justice

However, the constitution of classes is not the sole result of SVETLAN'. The structuring of semantic domains is another. Instead of bag of words, domains are

now described by verbs associated to classes defining their arguments. This kind of knowledge is a first step towards schema representation of pragmatic knowledge. Such an example is given in Table 5.

6. Experiments

6.1. Corpus Characteristics

We conducted an experiment with a corpus of English newspaper articles composed of 3 months of the "Los Angeles Times" newspaper. We used the following experimental settings: segmentation of the corpus and creation of the thematic memory (i.e. the set of semantic domains); syntactic analysis and syntactic links extraction; structured memory creation (i.e. the set of structured domains); and lastly, an evaluation of the results. We first counted the number of correct classes. A correct class is one that contains words sharing a direct semantic link. For the wrong classes, we counted the number of errors due to parse errors.

For our experiment, we only keep the TUs that lead to build stable domains, i.e. domains grouping at least 10 TUs.

The corpus we worked on is unanalyzed and SGML encoded. Its language level is high with a journalistic style and it tackles various topics. The size of corpus is 7.3 million words.

6.2. Results

The thematic memory created contains 138 stable domains. Table 8 shows results obtained with these domains. Within about 150 classes, about 60% are correct while 7% of wrong classes are due to parse errors.

Number	Correct	Syntactic Parser Errors	Other
149	58 %	7 %	35 %

Table 8: Results on English with a 0.1 threshold

Table 9 shows some examples of the classes contained in a structured domain whose topic is medicine.

Verb	Rel^{on}	Class
To take	Under	Home, residence
To meet	Object	Care, physician
To carry	Object	Virus, antibody
To get	Subject	Treatment, care

Table 9: Examples of classes in a structured domain on English

These examples show two classes with the word care. They instantiate two different kinds of semantic relation: in the class <care, treatment> we see an instrument link between the two terms of the class (a treatment is a means to take care of a patient) and in the class <care, physician>, the link is an agent one (the physician take care of his patients). Meanwhile, in the same structured domain, there were other classes containing the word *care*, some of them carrying the same meaning as care considered as a treatment. So classes do not partition the words of the domains, and they also do not partition the meanings of the words. In a further step, we will study if it is possible and suitable to merge the closest classes.

7. Query Expansion

We were interested in knowing which effects are produced by using different sorts of knowledge in query expansion. Thus, we did some preliminary experiments. Given a query made of words, we tried two kinds of expansion. One kind exploited the acquired classes and the other WordNet. WordNet is a lexical database made by lexicographers. It aims at representing the sense of the bigger part of the lexicon. It is composed of Synsets. A Synset is a set of words that are synonymous. These Synsets are linked by *IS-A* relations. We only did few experiments whose purpose was only to illustrate the interest of having contextual classes compared to a general database which often creates divergences when used as it is.

First, we selected the domain the closest to the query words. Different expansions where computed by adding the words that were belonging to the class of a word of the initial query, and this for each word of the query belonging to a class in the selected domain.

By this way, expansion is done relatively to the query domain of reference. It should be noted that another expansion might be done from a same word from another query, as soon as the other words of the query differ and refer to another context. On the contrary, when expanding with WordNet, the lack of domain knowledge does not allow to select only the right sense.

The queries were sent on Google, that only considers the first 10 words. We chose Google because it is a boolean engine, assuming that when the query contain a lot of words, the retrieved documents are more relevant, as they contain all the words of the query. It is also a way of showing the validity of the acquired classes. If there exists documents containing all the words of the expanded query, the class can be considered coherent. So, in this experiment, we tried to shorten the initial set of documents retrieved by Google.

Initial query : *prosecutor obstruction deliberation jury*
=> 477 documents
SVETLAN' query expansion 1: prosecutor obstruction deliberation jury charge case court trial attorney count
=> 141 answers
SVETLAN' query expansion 2: prosecutor obstruction deliberation jury charge case court trial attorney sentence
=> 222 answers

When using WordNet, we retrieved the different meanings of each word – first, all its synonyms and its hypernyms and second, only the synonyms – and add each of these sets to the initial query. Such a set was considered equivalent to an acquired class. Thus for the same initial query, we obtained the following query expansions.

1 sense of prosecutor (its synonymous and after "=>" its hypernyms)
Sense 1: prosecutor, prosecuting officer, prosecuting attorney
=> lawyer, attorney

Initial query : *prosecutor obstruction deliberation jury*
WordNet expansion 1: prosecutor obstruction deliberation jury, prosecuting officer, prosecuting attorney, lawyer, attorney
=> 65 answers

4 senses of obstruction
Sense 1 :obstruction, impediment, impedimenta
=> structure, construction
Sense 2: obstacle, obstruction
=> hindrance, deterrent, impediment, handicap
Sense 3: obstruction
=> hindrance, interference, interfering
Sense 4: obstruction
=> maneuver, manoeuvre, play

Initial query : *prosecutor obstruction deliberation jury*
WordNet expansion 2: prosecutor obstruction deliberation jury, impediment, impedimenta, structure, construction, obstacle, hindrance, deterrent, handicap, interference, interfering
=> No answer
WordNet expansion 2bis: prosecutor obstruction deliberation jury, impediment, impedimenta, obstacle
=> No answer

5 senses of deliberation
Sense 1: deliberation
=> discussion, give-and-take, word
Sense 2: deliberation, weighing, advisement
=> consideration
Sense 3: calculation, deliberation
=> planning, preparation, provision
Sense 4: slowness, deliberation, deliberateness, unhurriedness
=> pace, rate
Sense 5: deliberation, deliberateness
=> thoughtfulness

Initial query : *prosecutor obstruction deliberation jury*
WordNet expansion 3: prosecutor obstruction deliberation jury, discussion, give-and-take, word, weighing, advisement, consideration, calculation, planning, preparation, provision, slowness, deliberateness, unhurriedness, thoughtfulness
=> No answer
WordNet expansion 3bis: prosecutor obstruction deliberation jury, weighing, advisement, calculation, slowness, deliberateness, unhurriedness
=> No answer

2 senses of jury
Sense 1: jury
=> body
Sense 2: jury, panel
=> committee, commission

Initial query : *prosecutor obstruction deliberation jury*
WordNet expansion 4: prosecutor obstruction deliberation jury, discussion, body, committee, commission
=> 84 answers

We can see that expansions along the WordNet synonyms of polysemous words do not lead to a successful research, as for *deliberation* and *obstruction*. An explanation of this result comes from the fact that SVETLAN's added words are much more related to the query than those added via WordNet. It is due to the contextual construction of the classes and also to the fact that the context is explicitly represented by domains and so can be used to guide the choice of words, contrarily to what happen when using WordNet. WordNet coverage is large but this quality is, in a sense, its shortcoming. Indeed, the generality of its contents makes it difficult to use in real sized applications. It rarely can be used without a lot of manual adaptation.

We are now showing another example, in the sport domain. SVETLAN' added words that all belong to the baseball domain and also lead to reduce the number of retrieved documents.

Initial query: starter hitter batter : 14900 answers
Svetlan'A expansion: starter hitter batter run hit game inning pitch season home
=> 7660 answers

In WordNet, starter and batter are very polysemous words.

5 senses of starter
Sense 1: starter
=> electric motor
Sense 2: starter
=> contestant
Sense 3: starter, dispatcher
=> official
Sense 4: newcomer, fledgling, fledgeling, starter, neophyte, freshman, entrant
=> novice, beginner, tyro, tiro, initiate
Sense 5: crank, starter
=> hand tool

1 sense of hitter
Sense 1: batter, hitter, slugger, batsman
=> ballplayer, baseball player

2 senses of batter
Sense 1: batter, hitter, slugger, batsman
=> ballplayer, baseball player
Sense 2: batter
=> concoction, mixture, intermixture

In such a case, it is not possible to obtain a correct expansion by using only WordNet.

However, one can envisage using SVETLAN' knowledge to select a meaning in WordNet. By combining on one hand sets of semantic closed words, without explicit types of link, and on the other hand sets of words with typed semantic relations that often are no more semantically closed if they are all merged, we could

maybe use the first sets to select contextual meanings in the second sets.

8. Related Works

There is a lot of works dedicated to the formation of classes of words. These classes have very various statuses. They can contain words belonging to the same semantic field or near synonymous, for example.

Automatic systems apply different criteria to group words, but all make use of a context notion or a proximity measure. IMToolset, by Uri Zernik (Zernik, 1991), cluster local contexts of a studied word that is defined by the 10 words surrounding it in the texts. The proximity between words is evaluated by using the mutual information measure, as we do when segmenting the text. The result is groups of words that are similar to our domains but more focused on the sense of a word alone.

Faure and Nedellec (Faure & Nedellec, 1998) with Asium, or Lin (Lin, 1998) apply distributional approaches to learn classes. Asium was designed to build ontology of specialized domains, so there is no need for a context restriction. Its basic classes are clustered to create ontology by the mean of a cooperative learning algorithm. This manual cooperative part is a step analogous to our filtering step. Lin does not apply a contextual selection of the words before regrouping them; he defined a similarity measure between words of a same class to order them according to their similarity degree, This kind of method also lead to build large classes, analogous to our semantic domains.

9. Conclusion and Future Work

The system SVETLAN' we propose, in conjunction with SEGAPSITH and a syntactic parser, extracts classes of words from raw texts and structures domains initially made of bags of words. These classes are created by the gathering of nouns appearing with the same syntactic role after the same verb inside a context. This context is made by the aggregation of text segments referring to similar subjects. Our experiments on different corpus give good enough results, but they also confirm that a great volume of data is necessary in order to extract a large quantity of lexical knowledge by the analysis of syntactic distributions.

In order to show the interest of building small classes inside larger domains, we made some query expansions that comfort the feeling of real proximity between words in the classes and their interest for specializing a query. We are now studying how this expansion can be used in a question-answering system (Ferret et al., 2001) developed in the group that participated to the TREC evaluations. This task is open domain and when the answer is not expressed in the documents with the same words as the question, it requires finding exact synonyms in text sentences. A first step will consist of augmenting our base by applying our system on much more texts, then trying to use WordNet in conjunction with SVETLAN': a synonym in WordNet would be selected if it occurs in a class of SVETLAN' or in classes very close each others. As SVETLAN' classes do not only contain synonyms, the classes are not sufficient in this case, while used along with WordNet it would be a very sure criterion to obtain synonyms in a specified context. We have to verify that it will be applicable on a large scale.

10. References

R. Agarwal, Semantic feature extraction from technical texts with limited human intervention, PhD thesis, Mississippi State University, 1995.

Gaël de Chalendar and Brigitte Grau, SVETLAN' or how to Classify Words using their Context, Proceedings of 12th EKAW, Juan-les-Pins, France, October 2000, pages 203-216 Rose Dieng and Olivier Corby (Eds.), Springer, 2000, (Lectures notes in computer science; Vol. 1937 : Lectures notes in artificial intelligence).

David Faure and Claire Nedellec, ASIUM, Learning subcategorization frames and restrictions of selection. In Y. Kodratoff ed., proceedings of 10th ECML – Workshop on text mining, 1998.

Christiane Fellbaum, WordNet: an electronic lexical database, The MIT Press, 1998.

Olivier Ferret, How to thematically segment texts by using lexical cohesion? Proceedings of ACL-COLING'98 (student session), pp. 1481-1483, Montreal, Canada, 1998.

Olivier Ferret and Brigitte Grau, A Thematic Segmentation Procedure for Extracting Semantic Domains from Texts, Proceedings of ECAI'98, Brighton, 1998.

O. Ferret, B. Grau, M. Hurault-Plantet, G. Illouz, C. Jacquemin, Terminological variants for document selection and Question/Answer matching, Proceedings of the Open-Domain Question Answering Workshop, Conference of ACL/EACL, Toulouse, 2001

Dennis Grinberg, John Lafferty and Daniel Sleator, A robust parsing algorithm for link grammars, Carnegie Mellon University Computer Science technical report CMU-CS-95-125, and Proceedings of the Fourth International Workshop on Parsing Technologies, Prague, September, 1995.

Zellig Harris, Mathematical Structures of Language, Wiley, New York, 1968.

C.-Y. Lin, Robust Automated Topic Identification, Doctoral Dissertation, University of Southern California, 1997.

Douglas Lenat, Cyc: a large-scale investment in knowledge infrastructure. Communications of the ACM, 38(11), 1995.

Dekang Lin. Automatic retrieval and clustering of similar words. In Proceedings of COLINGACL '98, pages 768--774, Montreal, Canada, August 1998.

Fernando Pereira, Naftali Tishby and Lillian Lee, Distributional clustering of english words, Proceedings of ACL'93, 1993.

B. Roark and E. Charniak, Noun-phrase co-occurrence statistics for semi-automatic semantic lexicon construction. Proceedings of COLING-ACL'98, pp. 1110-1116, 1998.

Roger C. Schank, Dynamic memory. A theory of reminding and learning in computers and people. Cambridge University Press, 1982.

Uri Zernik, TRAIN1 vs. TRAIN2: Tagging Word Senses in Corpus, RIAO'91, 1991

Supported by ELSNET

The Workshop Programme

Sunday, June 2, 2002 (Provisional program)

Start	End	Action	Title	Actor(s)
14:30	14:45	Opening	Introduction to this workshop	Steven Krauwer
14:45	15:15	Talk	Summary of the MREMS Workshop	Mark Maybury
15:15	15:40	Talk	*Challenges and Important Aspects in Planning and Performing Evaluation Studies for Multimodal Dialogue Systems*	Susanne Höllerer
15:40	16:05	Talk	*XML and multimodal corpus design: experiences with multi-layered stand-off annotations in the GeM corpus*	John Bateman, Judy Delin, Renate Herschel
16:05	16:30	Talk	*Towards a roadmap for Human Language Technologies: Dutch-Flemish experience*	Diana Binnenpoorte, Catia Cucchiarini, Elisabeth D'Halleweyn, Janienke Sturm and Folkert de Vriend
16:30	17:00	**Break**		
17:00	17:30	Talk	About Roadmapping: Introduction to the plenary exercises	Steven Krauwer / Hans Uszkoreit
17:30	18:30	Exercise 1	Identifying priorities	All
18:30	19:30	Exercise 2	Putting them on a timeline	All
19:30	20:00	Discussion	Where to go from here	All & Steven Krauwer
20:00		**Closing**		

Workshop Organisers and Programme Committee

- Steven Krauwer (ELSNET / Utrecht University) (Chair)
- Hans Uszkoreit (DFKI Saarbruecken)
- Antonio Zampolli (Univ of Pisa)
- Joseph Mariani (LIMSI, Paris)
- Ulrich Heid (IMS Stuttgart)
- Khalid Choukri (ELDA Paris)
- Mark Maybury (MITRE)

Table of Contents

Author Index

Preface

Aim of the workshop:

The aim of the proposed workshop is to bring together key players in the field of resources and evaluation in order to make a first step towards the creation of a broadly supported Roadmap for Language Resources, i.e. a broadly supported view on the longer, medium and shorter term needs and priorities. This activity should be seen in the context of ELSNET's other roadmapping activities (see http://www.elsnet.org/roadmap.html), which aim at developing a technological roadmap for the whole field of Human Language Technologies.

The purpose of such roadmaps is to give the R&D community an instrument to identify opportunities for concertation of their activities and better exploitation of possible synergies between players all over the world.

Scope of this workshop:

there is no standard model for roadmaps for resources and evaluation available, we will narrow the scope of this roadmapping workshop to a specific sub-area: Multimodal Language Resources and Evaluation. This will make our discussions more focused and concrete, and it will also allow us to exploit the fact that this workshop will take place the As day after the workshop dedicated to Multimodal Resources and Evaluation of Multimodal Systems (MREMS) in general.

Recommended reading (preferably before the workshop):

- ELSNET's First Roadmap Report, edited by Ole Bernsen (http://utrecht.elsnet.org/roadmap/docs/rm-bernsen-v2.pdf),
- ELSNET's Second Roadmap Report, edited by Dorothee Ziegler-Eisele and Andreas Eisele (http://utrecht.elsnet.org/roadmap/docs/rm-eisele-v2.pdf),

Both reports can be found in the annex of the proceedings

Results of the workshop:

The results of the workshop will be published on the ELSNET website at http://www.elsnet.org

April 2002

Steven Krauwer

(ELSNET Co-ordinator)

Challenges and Important Aspects in Planning and Performing Evaluation Studies for Multimodal Dialogue Systems

Susanne Höllerer

ftw. Telecommunications Research Center Vienna
Tech Gate Vienna
Donau-City-Straße 1/ 2nd floor
1220 Vienna
Austria
hoellerer@ftw.at

Abstract

In this paper I want to discuss the problems researchers face in trying to plan and carry out an evaluation study for multimodal systems – particularly in qualifying the purpose of the testing, defining the intended user group for their application, arranging the testing setting and aligning the evaluation plan. It is my intention to show which aspects should be taken into account and which basic standards should be fulfilled. Furthermore, I provide two sections about points to consider in performing the study as well as in the analysis of the received data. Then I describe possible difficulties concerning the evaluation of (multimodal) systems and try to sketch longer term solutions. Finally, I list possible options on how to utilize the results of evaluations studies in further research.

1. Introduction

Multimodal dialog systems should be efficient, easy to handle and comprehensible for intended users – so how should the evaluation of such dialogue systems be designed and carried out in order to accomplish these goals and how can the outcome and the conclusions of the concerned studies be used for further research and development?

How can researchers and developers of dialogue systems answer the needs and preferences of the users, how can they accommodate their special interests and characteristics?

And how can problematic issues researchers and developers face today be tackled and solved? To which extent can experiences made until now help to find solutions for the future?

These questions need to be answered already in the very beginning of the whole development process – before the start of the planning and mental development of the system one has to designate the goals of the system and which functions it serves. At the same time the target group has to be defined – this can on the one hand be a small group of experts and for a special field of application or on the other hand the entire population, depending on the system or object. During the development process these facts must be taken into account in order to produce the most efficient system for the special target group. To this end, it is useful to perform an iterative mode of evaluation which means that for every important phase of development an evaluation study is provided so as to find out about the direction the development of the systems leads to and to make sure that the intended users are able to handle it. Especially as far as multimodal dialogue systems are concerned, evaluation studies are a relevant part of the development and – at the same time – a challenging task. The particular difficulty is to provide methods for logging and analysing two or more different modalities and to test each of them separately as well as combined with the other(s). This means that the developers and researchers receive much data, which require experience and reliable methods to be analysed.

Because of the innovative design and handling of those systems a careful evaluation planning has to be provided.

I wrote this paper from a social scientific point of view – as a different perspective concerning the preparation and the procedure of evaluating a system. Social and empiric science can provide information on methodological issues, questions concerning the analysis of the data, the selection of test subjects, the arrangement of the setting and the formulation of the specific tasks for the test persons.

2. Goal of the paper

The main intention of this paper is to show how important a careful planning and performance of an evaluation study – concerning especially multimodal dialogue systems – is. It should be made clear which features of the testing process are particularly relevant and which problems may appear and which challenges the evaluation of a system, possessing more than one modality for handling, may involve. Furthermore, in this paper important aspects which may appear negligible at the first sight should be mentioned, for example the range of persons who are going to use this system in the future, ergo the intended user group: what are their characteristics, their needs and how can the system serve them? For the designer and the researcher, this means also knowing exactly the functions of the system. Another aspect would be the setting in which the testing should take place: how should it be arranged and which role plays the tester?

A very important part of this paper is the one about how the results of the evaluation research in general and the experiences of each researcher can contribute to the further research done in these fields and consequently to establishing standards for the design and the performance of evaluation studies of multimodal systems.

I also like to state my point of view concerning the present as well as the longer term problems researcher might face in developing and evaluating multimodal dialogue systems, and also how they might be avoided.

3. What is the image of the intended (average) user and how does it affect the development of the evaluation of a system?

As I mentioned before, theoretically the entire population can be the target group of a certain system or object, for instance of information extracting systems like automatic telephone enquiry for train schedules. As far as IT systems are concerned, in the last years it was often assumed implicitly that the circle of intended users is a rather small one (compared to the one of objects of everyday life) and is composed of experts of fields like computer technology and science, managers or other academic job-holders in hierarchical higher positions.

But today such systems should serve everybody. It is the developers´ duty to design the system in such a way that it can also be conceived and used by non-experts. That concerns especially the presentation of the graphical user interface, which the user gets the first impression of before even having tried out how to handle the application.

So if one has in mind that the target group may be as heterogeneous as the general population there is no possibility to postulate any specific knowledge or experience concerning multimodal dialogue systems among all persons. This means that the researcher has to begin at the very start and make the use of the system as easy as possible. That is for sure a very challenging task – and an important one, because the design and the usability of the system are important factors for its acceptance among the intended users. One has to consider that persons of every age group, sex, society position and socialisation background may use this system. The sample the researcher assorts should be representative in so far that each of these parameters is taken into account. One possibility to find out about those features is to provide a user questionnaire.

One option is to search test persons of a certain age, sex and education level. The last parameter is useful to get some information about the position in society they bear. Another way would be to consider income or field of profession, or respectively the job they are working in. It is hard to find out much about the economic or social background of the test persons without violating their privacy. And one must not believe that one statistical feature gives information about a person's standard of living. So this parameter is a rather hard one to obtain. Nevertheless it should be included in the evaluation.

The aspect of age is also an important one, because one may find big differences between younger and older people concerning their competence as well as their experience with modern technological instruments and systems – a phenomenon which does not apply universally. But there may be the tendency that older persons are more sceptical and reserved if not afraid to serve as test persons for evaluations of such systems. They often argue that they need not be taken into account, because they are too old – which is of course a misbelief.

It is common to consider sex as a variable, too, because it is interesting to view possible differences between women and men in handling technological systems and to react to them in the further development of the system.

4. Recommended standards for multimodal dialogue systems

It seems to be of use to establish basic standards that need to be fulfilled in order to provide a system appropriate for a great range of users. This is particularly relevant for multimodal dialogue systems, which provide several ways of handling and therefore require extraordinary user-friendliness. The standards described in this chapter can be seen as provisional and extensible – they should serve as basic points of orientation.

These standards make clear which direction further research should take and on which aspects it should focus, but also which main issues any evaluations study should focus on.

4.1. Easy intelligibility of the functions and applicability of the system

In order to be able to use the system in an efficient way the users have to understand which actions one can perform and which goals one can accomplish with it. That is to say that the instruction manual must be clear and specific. But also the design of the user interface should give a clue to how to use the system.

4.2. Distinct visual design of the graphical user interface

The user interface, i.e. the part of the system the user sees and interacts with, should not be complex, but the various elements should be arranged clearly and distinguishably. Concerning this feature, knowledge from fields like psychology or the specific domain of advertisement could be advantageous, but the cooperation between these fields and the one of IT is not that strong yet.

4.3. Good intelligibility of the commands

The language in which the user communicates with the system is usually a set of commands – either given via speech or via GUI. And vice versa the systems gives commands or poses questions to the user – often in the form of spoken prompts. It is necessary to formulate these in a simple and intelligible way so that the user is able to catch it.

4.4. Good speech recognition

A dialogue system that provides the modality of handling via speech needs to have an excellent speech recognizer. That is a prerequisite for efficiency, which is an overall goal of such systems. This means that it should also work in noisy environment, as the system should be adaptable in awkward situations where the user cannot have regard of a clear articulation. Unfortunately – although there has been much research done in this area – it takes a long time to develop a good recognizer respectively it is hard to find a recognizer appropriate for the functions a system should fulfil

4.5. Efficiency as well as smooth performance of the actions

Multimodal dialogue systems have the special aim to work smoothly even in difficult or stressful situations, for instance if the user needs both hands for other actions. It would be very exhausting for the user to be forced to

repeat the commands or questions a several times because of the slow processing or the long upload time of the system.

4.6. Clear, intelligible output (speech-output as well as output via the GUI)

In order to provide a smooth process and a good information extraction respectively an optimum support the output the system delivers should be correct as well as intelligible.

5. Advantages of multimodal dialogue systems

The advantages listed in this chapter are supposed to supplement or – in part – condition each other. This list should – on the one hand – emphasize the differences between single- and multimodal dialogue systems and – on the other hand – show which anticipations researcher have concerning these kind of systems.

To reach the intended users as well as to make the system interesting for them, one has to emphasize its advantages in achieving a certain goal, possibly by comparing it to other kinds of systems or – in general – ways to reach this goal (for instance using a multimodal dialogue device to extract information about the surroundings instead of a simple map).

One big use of such a multimodal dialogue system is for sure the *flexibility*. The overall goal of the development of multimodal applications is for the users to interact with the system the way they like to – depending on the situation they are in. For example when driving in his car, the user cannot use his hands to operate the system – so there has to be one or more other ways to handle it in order to fulfil the claims of efficiency and usability. In this case, the modality of handling via speech input is an optimum alternative.

The optimum situation would be that every user was free to interact with the system the way the situation requires it – and to alternate the one modality with the other(s) in a spontaneous way. The system should therefore be designed to react and adapt to this user-specific behaviour. This demands – in case of a multimodal dialogue system – an excellent speech recognition as well as a synoptic user interface quick to apprehend.

Beside flexibility higher *efficiency* is another advantage of multimodal dialogue systems – provided that sufficient evaluation studies has been performed in order to find out about how a system needs to be designed to serve the users well. It´s clear that efficiency – at least in part – grows proportionally with flexibility (and the other way round), so these two aspects are connected tightly.

A third advantage which may be of great importance for "everyday users" is the *individuality and personality* a system gets when becoming multimodal, hence being able to be integrated smoothly in ones everyday life and supporting the performance of certain actions.

The great use of multimodal dialogue systems in comparison to other systems is the fact that they combine the advantages of the single modalities they include, this means that the user can profit from the advantages of handling via the GUI as well as via speech. In detail, this would be promptness as far as the modality of speech is concerned – action can be executed far more faster by speaking the commands that by typing them. The other advantage which is already known is the possibility to keep ones hands free for other actions which is, for instance, very important while driving the car. Regarding the modality of handling via the GUI the main advantage lies in the privacy of the commands the user is giving and of the actions the system is executing. While speech can be received by persons around the user, actions like typing are not audible.

Disadvantages of one modality might be eluded by using the other modality – for instance if the speech recognizer does not work properly.

6. Important items in planning and carrying out an evaluation

To evaluate a system one has to know exactly which functions it possesses and who the intended users are (cf. Nielsen 1993: 170). The evaluation study is performed to serve the purpose of finding out more about how the system should be designed in order to answer the needs and interests of the users. As I mentioned in the introduction the best way to carry out an exhaustive evaluation study is to perform several smaller "steps of evaluation". This means that – depending on the development stage of the system – the respective properties, the design and the effect on the users need to be measured.

And for each of these steps some important points must be considered. To receive sufficient and eligible data for the analysis afterwards, the evaluation study needs to be planned carefully, tasks for the test persons to perform must be formulated – which are supposed to accomplish the intentions the researchers have –, methods to log the process of evaluation need to be found as well as methods to capture the impressions and experiences of the test persons. The choice of these instruments depends on which aspects of the tests are important for the developers on the one hand, and – on the other hand – how easily the requested information can be extracted. A good way to find out which methods are appropriate for the evaluation study is to evaluate the logging methods themselves. That is also useful to assure that the methods one uses really measure what they are pretending to measure – hence if they are suitable for what the respective developer wants to find out. A good method for logging the evaluation process is to use instruments like audio recorder, video camera, mouse tracker, screen logger or eye tracker. But one must be aware that receiving too much data out of an evaluation study can be as well a problem as receiving too little.

Not only the methods and the technological equipment are to be considered – the whole setting of the testing process needs to be planned. The role of the tester who stays with the test persons must be defined – mostly he is the one who explains the aim of the testing as well as the specific tasks and who observes the test person during the performance. This raises some questions: How much information should the test person be given in order to not affect the authenticity of the situation and the (possible) impartiality of the user? Should the tester answer questions during the testing? Where should he place himself? To which extent should he adapt himself to the test person (concerning behaviour, speech, …) to provide a more informal setting or how can he prevent himself

from doing so? Are there differences in the performance of the test persons depending on the sex, the age or the credibility of the tester?

As far as the test persons are concerned, should they be given some time to get to know the system better (some minutes without logging or even observing) or should they be tested from the very beginning?

And how should the testing setting look like to provide as much authenticity as possible?

All these questions can become problematic when too little time and know-how is spent on the preparations of the evaluation studies – the difficult issues are explained in detail in chapter 7.

7. Analysis methods

It is important to find appropriate analysis methods as well, for instance annotating schemes to analyse spoken language and synchronize it with actions like mouse movements or clicks. That is a good and rather objective possibility of spotting the problems the users had performing the tasks, but also the points where the test persons apparently used the system in an efficient way, for example – concerning multimodal dialogue systems – combined speech and handling via the GUI. Especially for large numbers of test persons and hence a lot of data such standardized analysing methods are useful. However, the range of good and reliable annotating schemes is not that great. The few that are made use of in empirical studies fall short of easy applicability and efficient programming. Much remains to be done in this field of research. Also the methods themselves need to be tested to find out if they work the way the researcher wants them to. If the methods fail, the whole study needs to be repeated.

However, also the subjective impressions of the test persons are important for the analysis, so one should not surrender a questionnaire, an individual interview or informal talk with the test persons after the testing. These data need to be analysed either quantitatively – in the case of a questionnaire – and presented in statistics or analysed in a qualitative way, that is to collect the test persons' impressions and statements and to detect positive or negative tendencies.

But not in every case all the errors of a system can be detected: one cannot be sure that all the problems could actually be recovered – "one troubling aspect of testing is the uncertainty that remains even after exhaustive testing by multiple methods". [Shneiderman 1998: 125]

8. Problems to be solved concerning the process of developing and testing a new system

There is a range of problems researchers of multimodal dialogue systems have to face during the process of developing and optimizing the system. In some ways, preparing the evaluation study for multimodal dialogue systems does not differ from preparing one for "singlemodal" systems. Just a few aspects are more challenging as far as multimodal systems are concerned.

8.1. Defining the user group and test subjects

First of all, it is difficult to find out about the intended user group: how should the researchers know which persons the system will be used by? And how can they be sure that the users they design the system for are really the ones who will use the system in the end? A step towards finding a solution to this problem would be to carry out a survey among the supposed target group or among the whole population to get a clue about who is interested in the product and may benefit from it.

The range of test persons should be representative for the group of intended users, that is to say that the test subjects should represent the properties of the target group. If the system was designed primarily for elder persons, it is recommended to choose such persons for the evaluation study. In this regard the question must be raised where one should find appropriate test subjects. There are several possibilities:

One may look for persons in public institutions or buildings like schools, universities or on the street. An alternative would be the search for people by an advertisement. Or one may get access to a range of test persons by buying (or exchanging) subjects databases.

8.2. The discrepancy between researcher and user

Another difficulty in the process of developing a (multimodal dialogue) system is the discrepancy between researcher/developer and "normal" user or test subject. The researcher who designs the system is an expert in this field, he/she possesses knowledge and experiences concerning this specific system and knows how to handle it – so one can assume that he/she is the person appropriate for testing the system. That is true – to some extent. The persons, who understand the functions and operations of the system best, may also know how to measure and optimize them. The problem, which may occur, is that the researcher knows the system to well. This means that he/she is not able to put him/herself in the situation of the non-expert user and, therefore, blind out all his/her knowledge. One may argue that just because of these problems evaluation studies are carried out. That is correct. But it is not enough to perform one or more evaluation studies, it has to be guaranteed that the study is performed in a correct way, this means to really find out about the user group and its needs and expectations. The researchers are – in some way – preoccupied. So they do not seem to suit for planning such a study. A possibility to avoid this problem would be to separate the role of the researcher and the one of the evaluation designer strictly. But here another difficulty appears: how can the evaluation designer know enough about the system to understand its functions and features and at the same time know not too much about it in order to stay as objective as possible?

8.3. The evaluation setting

In order to get valid testing results, not only the tasks to fulfill need to be chosen carefully, also the setting where the testing should take place has to be planned regardfully.

The easiest possibility is to perform the tests within an isolated laboratory or at least in the rooms of the company which developed the system. This would mean that the testing situation could be controlled rather easily and that no unexpected disturbances would happen. These apparent advantages entail one negative aspect. Choosing such a testing setting would mean that the authenticity of the situation would be in peril. Especially as far as

applications are concerned which are not designed to be used at home or in a quiet and private place, testing within the circumstances mentioned above would not represent the conditions which the user has to face when using the application in reality. The researcher cannot foresee all the different situations in which the system may be applied but he knows the intended user group and the functions of the application and therefore can assume how it is going to be used.

Portable devices for example are supposed to be applied on the way, for instance on the streets, in public buildings and institutions, while walking or traveling by car, at different events or in likewise noisy environment. The noise must not be underestimated – as well as other factors, for instance when information is required as quick as possible (train departure times for example). A system, which works perfectly within the laboratory setting, might turn out to fail when being used in real surroundings. How should these settings be imitated in the laboratory to gain valid results?

As a matter of course, one must in this case consider the development phase of the system. If there is not an application to be carried around yet it can hardly be tested like if there was. An iterative kind of evaluation study requires several different testing settings.

8.4. Methods

Finding appropriate methods for logging the testing process might be a problem as well. The choice depends on which modalities the system has, as there are several options for each of them to be logged and measured. Most multimodal dialogue systems offer at least the two following modalities: the speech-modality and the handling via the GUI.

In order to log spoken user-output, one could for instance use a simple recorder with a microphone or a camera which could also tape visual impressions like the gesture and the face of the test subject as well as the monitor of the computer or the display of the application device (if it is not too small) – depending on which kind of system is tested. At the same time, the output of the system should also be taped for to liaise the both kinds of output in order to get information about the quality of the speech recognition and the smoothness of the whole process.

It is not as simple to find methods – beside the camera – to trace the operations on the monitor or the display, ergo the handling via the GUI. There exist some software tools like screen logger or key tracker which log the mouse movements or clicks as well as the input via the keyboard or the selection via the menu. Unfortunately, the existing software is either very expensive or only available for companies of specific fields.

In addition, there are other tools to log the process in order to gain more information about the handling of the system – for instance a so-called eye-tracker that logs the eye movements of the user. Through its analysis one may find out about which elements of the GUI are bold and how easy or difficult it is for the user to understand how to operate the system.

The challenging aspect concerning multimodal systems is to connect the methods used for logging the handling of different modalities. One kind of information needs to be related with another. The speech signals must be synchronized with the manual actions, for instance. This intention requires another software or program like an annotation scheme.

8.5. Possible solutions and recommendations for the future

As I said before, it is necessary to involve several persons in the developing and the testing process of the system, as more perspectives are required for an effective evaluation study. Concretely, this means that persons from several fields of research should work together, the tasks should be distributed and the roles the persons occupy within this process should be defined well. The researcher, the developer, the designer, the market research institute, the tester, several university institutes like psychology, sociology and computer science – all of these persons and institutions have competences in their specific fields and can contribute to producing a good working system. Through exchanging experience and know-how, as many difficulties as possible might be avoided.

In my view, this strategy will play an important role in the future, for aspects like user friendliness and acceptance of the system by the users are more and more coming into prominence. It is not any longer the group of IT experts and business people only who need applications of new technologies, but “average persons” from every part of the society.

Nowadays the number of those companies increases which specialize on evaluation studies and tests on usability – a fact that indicates the prominence of these aspects.

While IT companies spent most time on producing new systems and optimising new technologies, the aspect of user friendliness was rather neglected. The big chance to catch up on these experiences is the cooperation with persons of other fields or companies; to get support at finding the right test subjects, equipment and methods.

9. Evaluation outcomes as resources for further research

First of all, the outcome of the evaluation studies serves the improvement and the development of the evaluated system. But the received data are not useless after completing the evaluation process. The lessons one draws out of this testing can be used for other – similar – studies. On the one hand developers get to know the logging and analysing methods better, on the other hand they learn more about this kind of systems in general and how the intended users manage them – this knowledge can be made use of in further research.

To mention the economical aspect, the results of an evaluation study can of course also be used in cooperation with other technological enterprises or research centers with commercial as well as scientific interests; they can be exchanged or sold.

This procedure does not need to be restricted to similar, ergo technological, fields of research – instead the knowledge can also be connected to different fields like psychological or sociological research or the particular field of advertisement, where methods of usability and analyses of effects on consumers and users have long tradition. Knowledge from these disciplines can be used for evaluation studies and – vice versa – the results of these kinds of studies can be made use of in other fields.

10. Conclusions

The challenges in designing evaluation studies for multimodal dialogue systems are plain to perceive: in contrast to dialogue systems using one modality, evaluating multimodal systems demands more than one perspective of testing and hence just as many methods of logging. To use the received data for the improvement of the system and for further research it has to be processed with the support of suitable analysis methods – the particular challenge at this is to find an appropriate method for each modality and each kind of data.

Another aspect is the definition of the purpose as well as the intended users, and as a main task the designing of the user interface and the systems functions in order to meet the interests and needs of the target group.

The field of research of multimodal dialogue systems and applications is relatively new and few standards concerning the design of the user interface or the ways of testing and analysing have been established. But today usability studies are attached more importance than ever – for every kind of system or object, not only in fields of technology. There are – on the contrary – branches that deal frequently with aspects of usability and already gained precious information, for instance (cognitive) psychology. This knowledge can be useful for enterprises or persons who develop such multimodal systems. In my opinion, the cooperation with other companies or even other fields of research and hence the exchange of experiences and know-how is one big chance to improve usability testing, even for very specific applications.

Acknowledgements:
This work was supported within the Austrian competence center program ***K****plu*s, and by the companies Kapsch, Mobilkom Austria and Siemens.

11. References

Nielsen, Jakob (1993): Usability Engineering. Academic Press. San Diego.

Shneiderman, Ben (1998): Designing the User Interface. Strategies for Effective Human-Computer Interaction. Addison-Wesley. Reading, Massachusetts.

XML and multimodal corpus design: experiences with multi-layered stand-off annotations in the GeM corpus

John Bateman*, Judy Delin†, Renate Henschel‡

*University of Bremen, Bremen, Germany
bateman@uni-bremen.de

†University of Stirling, Stirling, Scotland
and Enterprise Information Design Unit, Newport Pagnell, Bucks, England
j.l.delin@stir.ac.uk *and* judy.delin@enterpriseidu.com

‡University of Stirling, Stirling, Scotland
rhenschel@uni-bremen.de

Abstract

Current views of multimodal language resources have not yet sufficiently captured the complex interrelationships within page-based information delivery. This is critical for development of multimodal corpora and language resources suitable for large-scale empirical investigation. Serious attempts to interrogate the nature of multimodal meaning-making in professionally-produced documents, both paper and electronic, require a clear understanding of the organisation of the layers into which meaning is organised. In this paper, we present the first multi-layered XML annotation scheme that meets these requirements, developed using a combination of expertise from computational linguists and designers from various sectors of the publishing industry.

1. Introduction

With current developments and goals involving multimodal documents in the widest sense—i.e., including highly interactive artifacts capable of responding to, and producing information in, input/output modes ranging across verbal, gesture, touch and so on, animated/video content, traditional texts, graphics, and so on—it is perhaps tempting to believe that the organization of 'simpler', more traditional document forms, such as two-dimensional presentations involving textual, graphical and diagrammatic information, has been 'solved'. Attention is then drawn away from the complexities of these document types, such as they are, and are to be picked up as a by-product of dealings with more complex artifacts. In our ongoing work on two-dimensional, non-animated information presentations—e.g., books, information leaflets, traditional websites, newspapers (in both print and online forms), and so on—we have found a wealth of complexity that raises serious doubts about such an approach. One aspect of the problem, and the challenge, can be seen in the large gap that exists between previous corpus encoding initiatives (e.g., TEI and the derived CES) which are text based and more recent proposals for capturing mixed media/mode presentations: Somewhere between these two extremes, much of the highly flexible and meaningful resources of two-dimensional information presentation traditionally and non-technically subsumed under 'layout' and graphic design go missing.

As a consequence of this, we have found it necessary to develop a new annotation scheme for describing the informational relationships employed in the area. Two-dimensional information presentation—whether on the page, screen, or whatever—still represents the overwhelming majority of users' contact with information, and so a revealing and empirically based understanding of the meaning-making resources of this area remains of crucial importance. Previous attempts to provide annotation schemes for setting up corpora for documents of this kind have not succeeded in covering very much of the range of phenomena encountered in natural documents however (Corio and Lapalme, 1998; Bouayad-Agha, 1999; Bouayad-Agha, 2000). In this paper, we describe the goals of our own annotation work, set out the basic levels of annotation we believe are required, describe the technical approach taken, and indicate what we see as the next immediate stages, problems and challenges of follow-up development.

2. Goals

We take the view that language, layout, image, and typography are all purposive forms of communication. Accordingly, in our research project GeM ("Genre and Multimodality", http://www.purl.org/net/gem), we aim to describe and analyse all these elements within a common framework, thereby providing a more complete understanding of meaning-making in visual artefacts. By analysing resources across visual and verbal modes, we can see the purpose of each in contributing to the message and structure of the communicative artefact as a whole.

One particular goal of the research is to formalise and model the role of *genre* in layout and typographical decisions. Through the analysis of sample types of multimodal document, the project aims to develop a theory of visual and textual page layout in electronic and paper documents that includes adequate attention to local and expert knowledge in information design. The model is being implemented in the form of a computer program that allows exploration of both existing and potential layout genres, generating alternative and novel layouts for evaluation by design profes-

sionals.

Our use of the term genre here is similar to Biber's (1989, pp5–6), who in his study of linguistic variation states that 'text categorizations readily distinguished by mature speakers of a language; for example—novels, newspaper articles, editorials, academic articles, public speeches, radio broadcasts, and everyday conversations—categories defined primarily on the basis of external format'. We adhere, too, to Biber's view that these categories of text also reflect distinctions in the author's purpose: the documents look different, and contain different language forms, because they are intended to do different things.

Although there are many attempts to categorise the kinds of language that occur in different genres of texts in linguistics, there are few attempts to extend genre analysis into other aspects of visual meaning: Twyman (1982) and Bernhardt (1985), for example, provide preliminary schemes for categorising documents according to the interrelationships between images and text, while Kress and Van Leeuwen (2001) have now also explicitly begun to relate multimodality and genre. Waller (1987), however, is the only attempt extant, to our knowledge, that has attempted to describe the role of language, document content, practical production context and visual appearance in the formation of document genre within the same framework. Our work draws upon and extends Waller's in several ways, as we shall make clear below.

For this, or any project addressing the communicative strategies involved in two-dimensional visual artefacts, the provision of suitable corpus materials is fundamental. Furthermore, since such materials are not currently available, the development of such a corpus has been adopted as an additional explicit goal of the GeM project. The purpose of the corpus development within GeM is to investigate systematic connections between a rich characterisation of the context of use of multimodal documents and their linguistic, graphical, and layout realisations. Within the GeM project itself, four broad document genres have been selected for initial treatment: traditional paper-based newspapers, online web-based newspaper sites, instructional documents, and wildlife books; in each area we have secured a collection of documents and have established contact with designers either expert in these respective fields or, in several cases, actually responsible for the documents gathered. We focus here on the annotation scheme that we have found necessary for structuring the corpus developed.

3. Basic levels of annotation

Waller (1987, pp178ff) represents the constraints on the typographer in producing a graphical document as emerging from three sources:

- Topic structure: 'typographic effects whose purpose is to display information about the author's argument—the purpose of the discourse';
- Artefact structure: 'those features of a typographic display that result from the physical nature of the document or display and its production technology';
- Access structure: 'those features that serve to make the document usable by readers and the status of its components clear'.

Waller did not produce detailed text analyses based on his model but, grounded as it is in the very practical concerns of document design, his view that document appearance results from satisfying goals at different levels is persuasive. We have particularly taken the force of his point that the physical nature of the document and its method of production play a major role in its appearance. In this way, the 'ideal' layout of information on a page may never occur: it must be 'folded in' to the structures afforded by the artefact, and labelled and arranged according to the structures required for access. Document design is therefore never 'free', in the sense that it is never motivated solely by the dictates of the subject matter. We therefore have required a place for these kinds of constraints in our annotation.

In our revision of Waller's model, we suggest that there is an advantage to be gained in uncollapsing his 'topic structure' into a separation between content and rhetorical presentation. We view content to be the 'raw' data out of which documents are constructed. What Waller describes as 'the author's argument' is not solely or completely dictated by content: many rhetorical presentations are compatible with the same content. In terms more familiar from natural language generation, we separate out the 'what-to-say' from rhetorically structured text plans for expressing that content. Secondly, we take what Waller terms 'artefact structure' to be not a structure in the sense of some set of ideas that are to be incorporated in the document, but rather as a constraint on the combination of all the other elements into a finished form.

The levels we propose as minimally necessary for revealing accounts of the operation of the kinds of visual artifacts being gathered in our corpus are, then, as follows:

- Content structure: the structure of the information to be communicated;
- Rhetorical structure: the rhetorical relationships between content elements; how the content is 'argued';
- Layout structure: the nature, appearance and position of communicative elements on the page;
- Navigation structure: the ways in which the intended mode(s) of consumption of the document is/are supported; and
- Linguistic structure: the structure of the language used to realise the layout elements.

We suggest that document genre is constituted both in terms of levels of description, and in terms of the constraints that operate on the information at each level in the generation of a document. Document design, then, arises out of the necessity to satisfy communicative goals at the five levels presented above, while also addressing a number of potentially competing and/or overlapping constraints:

- Canvas constraints: Constraints arising out of the physical nature of the object being produced: paper or

screen size; fold geometry such as for a leaflet; number of pages available for a particular topic, for example;

- Production constraints: Constraints arising out of the production technology: limit on page numbers, colours, size of included graphics, availability of photographs; for example, and constraints arising from the micro-and macro-economy of time or materials: e.g. deadlines; expense of using colour; necessity of incorporating advertising;

- Consumption constraints: Constraints arising out of the time, place, and manner of acquiring and consuming the document, such as method of selection at purchase point, or web browser sophistication and the changes it will make on downloading; also constraints arising out of the degree to which the document must be easy to read, understand, or otherwise use; fitness in relation to task (read straight through? Quick reference?); assumptions of expertise of reader, for example.

Following Waller (1987), then, we claim that not only is it possible to find systematic correspondences between these layers, but also that those correspondences themselves will depend on specifiable aspects of their context of use. In particular, they will depend on 'canvas constraints' set by the nature of the realizational medium (paper, screen-based browser, palmtop, screen resolution) and 'production constraints' imposed by available technology and design choices (allowable cost, number of pages, available printing or rendering techniques, etc.). A model of multimodal genre must begin by expressing adequately the above five levels of description as well as finding the most appropriate way of satisfying the three sets of constraints.

Our provision of a corpus of multimodal documents serves as the empirical basis for more thorough investigations of this claim. So far our work has identified widespread mismatches between rhetorical purposes and layout structures even among professionally produced documents; this offers a useful basis for constructive critique. We see the collection of extensive corpora of multimodal documents of this kind, annotated according to the levels of description that we have here briefly motivated, as an essential research and direction for the next five years.

4. Technical implementation

As we have seen, the two communication modes of visual and verbal information presentation are the main perspectives to be captured in the GeM annotation scheme. The scheme accordingly identifies textual elements (verbal mode) and layout elements (visual mode) in a multi-layered annotation, and specifies how these elements are grouped into hierarchical structures (primarily: the rhetorical structure for textual elements, the layout structure formed by the layout elements, and a page model formed by an 'area model': see below). The alignment between these intersecting hierarchies is achieved by specification of the 'GeM base'—a list of the basic units out of which the document is constructed. In accordance with the goal of the GeM project, the granularity of the linguistic basic units employed in the annotation is approximately the sentence level—this does not preclude providing correspondences with other levels of granularity that might be required for other purposes of course.

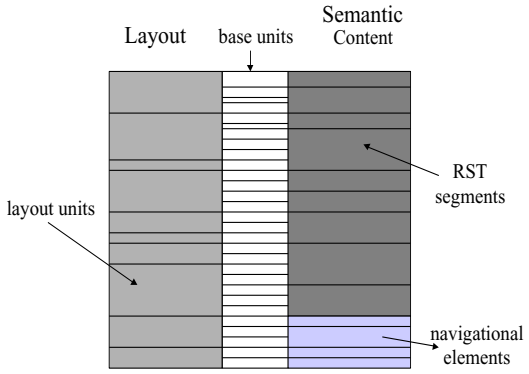


Figure 1: The distribution of base elements to layout, rhetorical and navigational elements

Each layer in the GeM model is represented formally as a structured XML specification, whose precise informational content and form is in turn defined by an appropriate Document Type Description (DTD).[1] The markup for one document then consists generally of the following four inter-related layers:

Name	content
GeM base	base units
RST base	rhetorical structure
Layout base	layout properties and structure
Navigation base	navigation elements and structure

All information apart from that of the base level is expressed in terms of pointers to the relevant units of the base level. This stand-off approach to annotation readily supports the necessary range of non-isomorphic, overlapping hierarchical structures commonly found even in the simplest documents. The relationships of the differing annotation levels to the base level units is depicted graphically in Figure 1. This shows that base units (the central column) provide the basic vocabulary for all other kinds of units and can, further, be cross-classified.

This annotation scheme is being developed further in response to the needs of concrete annotation tasks. Its current state is described in the technical manual available on the GeM website (Henschel, 2002). We describe it further here only in sufficient detail to give an impression of the kinds of annotation information and work involved.

4.1. Basic constituents

The purpose of the base level annotation is to identify the minimal elements which can serve as the common denominator for textual elements as well as for layout elements. Where speech-oriented corpora use the time line as basic reference method, and syntactically oriented corpora use the sequence of characters or words, the GeM annota-

[1] For the DTDs themselves, as well as further information and examples, see the GeM corpus webpages.

tion operates at a less delicate level and uses bigger chunks (mostly sentences and graphical page elements) as the bases of the markup. Everything which can be seen on each page of the document has to be included. How the material on each page is broken up into basic units is given by the following list, each is marked as a base unit:, orthographic sentences, sentence fragments initiating a list, headings, titles, headlines, photos, drawings, diagrams, figures (without caption), captions of photos, drawings, diagrams, tables, text in photos, drawings, diagrams, icons, tables cells, list headers, list items, list labels (itemizers), items in a menu, page numbers, footnotes (without footnote label), footnote labels, running heads, emphasized text, horizontal or vertical lines which function as delimiters between columns or rows, lines, arrows, and polylines which connect other base units.

Everything on a page should belong to one base unit. The base annotation has a flat structure, i.e. it consists of a list of base units.[2] Generally any text portion which is differentiated from its environment by its layout (e.g. typographically, background, border) should be marked as a base unit. The list of base units needs to comprise everything which can be seen on the page/pages of the document. The tag used to mark base units is the <**unit**>. Each base unit has the attribute **id**, which carries an identifying symbol. If the base unit consists of text, the start and end of this text is marked by the <**unit**> tag. Illustrations, however, are not copied into the GeM base. Thus, base units which represent an illustration or another graphical page element are empty XML-elements but can optionally be equipped with an **scr** and/or an **alt** attribute to show, indicate or access the source of an illustration.

4.2. Layout base

The layout base consists of three main parts: (a) layout segmentation – identification of the minimal layout units, (b) realization information – typographical and other layout properties of the basic layout units, and (c) the layout structure information – the grouping of the layout units into more complex layout entities. We explain these three components in more detail below.

In typography, the minimal layout element (in text) is the glyph. In GeM, however, we are primarily concerned with typographical and formatting effects at a more global level for a page; therefore we do not go into such detail, instead considering the paragraph as minimal layout element. That means, a sequence of sentences with the same typographical characteristica which makes up one paragraph is marked as one layout unit. In addition to that we mark all graphically realized elements from the GeM base as layout units. Also highlighted text pieces in sentences, or text pieces within illustrations are marked as layout units. Hence the same list which has been given for the markup of the base units applies here, but with paragraphs instead of orthographic sentences. The tag for a layout unit is <**layout-unit**>. Each layout-unit has the attribute **id**, which carries an identifying symbol, and the attribute **xref** which points to the base units which belong to this layout unit.

The second part of the layout base is the realization. Each layout unit specified in the layout segmentation has a visual realization. The most apparent difference is which mode has been used – the verbal or the visual mode. Following this distinction, the layout base differentiates between two kinds of elements: textual elements and graphical elements marked with the tags <**text**> and <**graphics**> respectively. These two elements have a differing sets of attributes describing their layout properties. The attributes are generally consistent with the layout attributes defined for XSL formatting object and CSS layout models.

Some of the layout units identified in the segmentation part of the layout base can be grouped into larger layout chunks. For instance, the heading and its belonging text form together a larger layout unit, or the cells of a table form the larger layout unit "table". The criterion for grouping layout elements into chunks is that the chunk should consist of elements of the same visual realization (font-family, font-size, ...), or the chunk is differentiated as a whole from its environment *visually* (e.g. by background colour or a surrounding box). In Reichenberger et al. (1995), the authors propose identifying layout chunks by applying a decreasing resolution to the document. The grouping into chunks usually can be applied in several steps, thus forming larger and larger layout chunks out of the basic layout units up to the entire document. Note that one chunk can consist of layout elements of different realizations (text and graphics). The third part of the layout base then serves to represent this hierarchical layout structure. Generally we assume that the layout structure of a document is tree-like with the entire document being the root. Each layout chunk is a node in the tree, and the basic layout units, which have been identified in the segmentation part of the layout base, are the terminal nodes of that tree.

Area model. Each page usually partitions its space into sub-areas. For instance, a page is often designed in three rows – the area for the running head (row-1), the area for the page body (row-2), and the area for the page number (row-3) – which are arranged vertically. The page body space can itself consist of two columns arranged horizontally. These rows/columns need not to be of equal size. For the present, we restrict ourselves to rectangular areas and sub-areas, and allow recursive area subdivision. The partitioning of the space of the entire document is defined in the **area-root**, which structures the document (page) into rectangular subareas in a table-like fashion.[3]

The tag to represent the area root is <**area-root**> The tag to represent the division of a sub-area into smaller rectangles is <**sub-area**>, this shares the attributes of the root but adds a **location** attribute so that subareas are positioned relative to their parent. Locations are indicated with respect to a logical grid defining rows and columns. If, for example, we were considering a page made up of a running

[2] In certain cases, we diverge from the flat structure of the base file. See the technical documentation for further details.

[3] Note that the area-root need not to be a page; if the document to be annotated is a book or brochure, then it can also be the entire book or brochure.

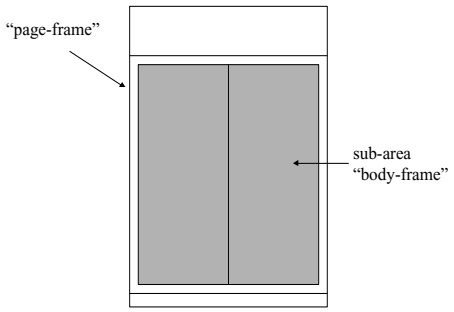


Figure 2: Visualized area model

head, a page body, and a footer for the page number, and in which the page body itself is divided into two columns, then the following annotation would define a corresponding area model. Here, the example's area model consists of a specification of the area-root (called "page-frame"), and the specification of one particular sub-area located in row-2 (called "body-frame"):

```
<area-root id="page-frame" cols="1" rows="3"
      hspacing="100" vspacing="10 85 5"
      height="16cm" width="14cm">
  <sub-area id="body-frame" location="row-2"
      cols="2" rows="1" hspacing="50 50"
      vspacing="100"/>
</area-root>
```

The attribute `vspacing=''10 85 5''` means that the running head takes 10% of the entire page height, the page body 85% and the page number 5%. The page body consisting of two columns is indicated by the hspacing attribute value "50 50", i.e., that both columns are equal in width and take half of the parent unit's width.[4] This area model is visualized in Figure 2.

The area model then provides logical names for the precise positioning of the layout units identified in the layout structure proper.

4.3. RST base

The RST base presents the rhetorical structure of the document. The rhetorical structure is annotated following the Rhetorical Structure Theory (RST) of Mann and Thompson (1988). In RST, a **span** is a continuous text fragment consisting either of a nucleus and one or more satellites (mononuclear relation), or of a number of nuclei which stand in a multinuclear relation (joint, sequence, ...) Some characteristics of RST vary between different research traditions, especially the granularity of the segmentation, the assumed set of rhetorical relations and the branching style of the rhetorical structure tree. We have also needed to make some extensions for the particularities of dealing with mixed verbal and visual information; clearly, when one wants to apply RST to modern, often multimodal, documents, new issues arise. Previous generalizations of RST to multimodal documents have either added new relations to model the relations between graphics and text (Schriver, 1996; Barthes, 1977) or parameterize the existing relation set by a mode parameter (André, 1995). We favour the second approach. However, there are other problems when generalizing RST to multimodal documents, which have not been addressed previously:

- The prominence of graphics in multimodal documents makes it often difficult to decide upon nuclearity in multimodal relations.
- The linear order of the constituents of the document is lost.
- The minimal unit for RST segmentation cannot be restricted to a clause or clause-like phrase.

We address these concerns briefly in turn.

Nuclearity in multimodal relations. Although graphical illustrations are often used to *rephrase* a text passage, it is often difficult to decide which of the two segments – the illustration or the text passage – is in fact nulear and which is the satellite. This seems to be a particular problem of graphics-text relations. To model this problem, we use the multinuclear **restatement** relation. A similar relation can also be found in Barthes under the name **redundant**.

Linear order. Conventional RST builds on the sequentiality of text segments. Relations are only possible (with some minor exceptions) between subsequent segments/spans (sequentiality assumption). With multimodal documents, the mutual spatial relations between the segments changes (from relations in a string-like object to relations in a graph). Segments can have not only a left and a right, but also an upper and a lower neighbour segment. In general one can imagine neighbouring segments in any direction, not only the four which presuppose a rectangular-based page layout. In addition to this, there can be more than one neighbour in each direction. The simpliest solution to apply RST (with its sequentiality assumption) to such a document would be to introduce a reading order on the segments of the document, which is then used as the sequence behind the RST structure. However, this can easily fail to reflect the actual reading behavior. A better, more straightforward generalization of the sequentiality assumption, which we will adopt here, is to restrict RST relations to pairs (sets) of document parts (segments/spans) which are adjacent in any direction. But again, in real documents, one can sometimes find a layout where the rhetorical structure obviously is in conflict with this adjacency condition. Our hypothesis here is that this is generally possible, but that in such a case an explicit navigational element is required so as to indicate the intimate relation between two separated layout units.

Clause as segment. The clause usually serves as minimal unit in RST. There are also approaches, which allow prepositional phrases to be a segment on their own. This is straightforward because both approaches assume something which denotes an action, an event or a state – also called eventualities – as the basic unit. However, if we move to modern documents, particularly multimodal documents, it is questionable whether the clause/PP basis should be kept. Typical examples in multimodal documents are:

[4] For the time being, we ignore space for margins, at least as long as they do not contain footnotes or other text.

- a diagram picturing a certain object and a text label which identifies (puts a name to) this object
- a list with an initiating sentence fragment, as in:

In the box are: ◇ three cordless handsets ◇ the base unit ◇ a mains power lead with adapter ◇ a telephone line cable ◇ two charger pods

- an attribute-value table, as in:

Juvenile	Grey-brown, flecked becoming whiter, adult plumage after three years.
Nest	Mound of seaweed on bare rocky ledge.
Voice	Harsh honks and grating calls at colony.

The cited examples are all expressions of states, or of static relationships between two objects or between an object and a property such as: identification, location, possession, and predication relations. In a traditional linear text, such relations would have been expressed as *is*- and/or *has*-clauses. Each such clause would constitute **one** basic RST segment. In our examples above, however, the two constituents of such a static relation clause are broken out and printed as separate layout units—in the first example, they are even given in differing modes. It is their mutual arrangement on the page plus possible extra graphical devices that expresses the relation between them. This raises the question as to what counts as a minimal unit for an RST analysis in such documents. We solve this issue by introducing a new component for annotation distinct from RST: we analyse the object-object/property relations, if they are clearly separate layout units, according to a small set of relations based on Halliday (1985), which we term 'intraclausal-relations'.

The tag used to mark the basic RST units is <**segment**>. In order to find out which base units form segments, one has to filter out those base units which are in the document for navigational reasons only. These are, for example, page numbers, running heads, footnote labels, document deictic expressions. We also consider headings as navigational elements, and do not include them in the RST analysis. In addition to these segments, we compose other complex segments consisting of more than one base unit for the cases where a intraclausal-relation is expressed on the page by two (or more) separate layout units. Typical examples are diagram + label, table cell$_{i,1}$ + table cell$_{i,2}$ in a two-column table, list initiating sentence fragment + list items. And, finally, sentences disrupted into two base units by page/column breaks only form one segment in the RST base.

The GeM XML annotation for RST aims to overcome some drawbacks found in existing RST annotation approaches. The two standards common in the RST community are those provided by the annotation tools of Daniel Marcu and Mick O'Donnell (see, e.g., www.sil.org/~mannb/rst/toolnote.htm). In both these tools, the annotated output is primarily seen as the program-internal representation of RST structures to be visualized as graphical trees with the help of the tool, but not as output to be used for further XML processing; we describe the pros and cons of the alternatives more in the technical documentation.

4.4. Navigation base

Navigation in a document is performed with the help of pointers, text pieces which tell the reader where the current text, or 'document thread', is continued or which point to an alternative continuation or continuations. The addresses used by such pointers are either names of RST spans or names of layout chunks. For long-distance navigation, typical nodes in the RST structure and in the layout structure have been established for use in pointers; in particular, chapter/section headings are names for RST spans and page numbers are names for page-sized layout-chunks, which tend to be used for navigation. However, there can also be other name-carrying layout-chunks or RST spans such as, for example, figures, tables, enumerated formulas, and so on. The navigation base of a document lists all these "names" which have been defined in this document to be actually or potentially used in pointers. We call the names of RST spans **entries** because they are usually placed immediately before the text of this span. We call the name of a layout-chunk an **index**.

The tag for an entry definition is <**entry**>. We allow entries simultaneously to be segments. We annotate the definition of an index at the page where it is defined, and refer with xref to the base unit which serves as the identifier.

Beside the list of entries and indices, which just defines addresses, the most important part of the navigation base consists of all pointers occuring in the document. The surface realization of pointers are "document deictic expressions", a term coined by Paraboni and van Deemter (2002). Document deictic expressions occur either within sentences or as separate layout units. We have marked the first type as embedded base units and the second as main level base units in the GeM base. In the navigation base, we specify the semantic meaning of such a document deictic expression as **pointer**. We distinguish pointers which operate on the layout structure, and pointers which operate on the RST structure. A pointer (or link) operating on the RST structure points from the current segment (which entails the document deictic expression) to an RST span – the goal RST span – which is layouted at a different place and is not adjacent. A pointer operating on the layout structure points from the layout chunk (which entails the document deictic expression) to another layout chunk which is not adjacent. Another distinction is the pointer type, which indicates different pointing situations. A **continuation** pointer is used in the situation where the layout of an article is broken into two non-adjacent parts. The second part is often printed several pages later than the first part. Continuation pointers are typically layout-operating pointers. **Branching** pointers are used in the situation where a certain piece of information is with respect to its content appropriate at two (or

more) places in the same document. The designer has decided to put it at one of the possible places. In order to indicate the other possible place, a pointer is given at the other location. A third type of pointer is the **expansion pointer**. It is used when more information is available, but not central to the writer's goal. An expansion pointer points to this extra information. Coming along a branching or an expansion pointer, the reader has the choice between two alternatives to continue reading the document. With a continuation pointer there is only the choice between reading continuation or stopping.

4.5. Uses of the corpus

The main results found so far in use of the corpus have been local, in that we are uncovering the rather wide variation that exists between selected layout structures on the one hand and rhetorical organization on the other *within single documents*. In surprisingly many cases, this variation goes beyond what might be considered 'good' design: in fact, we would argue that such designs are flawed and would be improved by a more explicit attention to the rhetorical force communicated by particular layout decisions. This represents the use of the corpus for document critique and improvement (cf. Delin and Bateman (2002)); here further corpus collection is nevertheless essential in order to map further the limits of acceptable functional variation.

We are also exploring the formulation of constraints over collections of corpus entries—e.g., over the pages of a book, or over collections of books in a series, etc.—by means of further annotation levels in which values from the primary annotation levels are partially specified. These need to be hierarchically related. It is at these 'meta' levels that the role of Waller's production and canvas constraints become particularly clear. We are employing this information as an important source of input in a prototype automatic document generation system capable of producing the kinds of variation and layout forms seen in our corpus, thus extending the early generation work in this spirit presented in Bateman et al. (2001).

Finally, we are still searching for more effective means of interogating the corpus maintained in the GeM style. Queries expressed in the XML Xpath language allow simple retrieval of information maintained in the corpus, but are cumbersome for more complex queries. Whether further developments such as XQL or XQuery will bring benefits is not yet clear. Somewhat disappointing was the unsuitability of the previous generation of linguistic-oriented corpus tools, which, despite considerable investment, seem to have been outstripped by the very rapid developments seen in the mainstream XML community. Most of our current work is done directly with XMLSpy and XLST tools such as Xalan. We have found the non-linearity and the non-consecutive nature of the units grouped within our annotation scheme as presenting a major problem for annotation models that have been developed in the speech processing tradition where contiguity of units is the expected case.

5. Follow-up goals, challenges and requirements

We expect that the details of annotation will be refined further as we approach a wider range of documents. It is now a major challenge to produce workable annotation schemes and corresponding corpus collections that include the kind of information we have argued to be necessary in this paper. This information represents a crucial bridging between technicalities of document production and the real issues of design faced in the publishing industry. Corpora built in this way will face two-ways: both to further linguistic and computational plinguistic research and development and to practical issues of design and evaluation. We believe that this needs a firm place in any roadmap now envisaged for language resource construction.

With this in mind we are also exploring a second round of corpus collection and annotation; it is our conviction that only a thorough corpus-oriented study of documents will allow further motivated theoretical and practical statements to be made about the meaning resources that such documents offer. If language resources are to be constructed that include documents of the kind targetted within GeM, then information such as that captured in the GeM annotation scheme will be crucial.

Here there are several issues that require concerted effort. Theoretically, the acceptance of the value and role of rhetorical analyses as giving a fine-grained description of communicative intentions is not uncontroversial. There are attempts in progress to produce corpora of texts annotated rhetorically. We believe this is also essential for multimodal documents. However, as we have detailed above, there are also significant issues that need now to be faced when we move away from linear presentations even to two-dimensional page-based presentations.

More practically, there are issues concerning how much information can be obtained from existing annotation and industry-standard markups: for example, the information maintained in professional document preparations tools such as QuarkXpress or Adobe Framemaker, InDesign, etc. Providing conversion tools to the kinds of linguistically motivated corpus annotations described here would open up a huge area of data. The genre and design knowledge encoded implicitly in style sheets and templates needs also to be made available so that it may be subjected to the kinds of study described above.

Of particular interest to us at present are further extensions across languages so as to compare cultural variation in visual/verbal presentations and further, more detailed comparison of documents variants created by repurposing (e.g., print-to-web, web-to-palmtop, etc.). In both cases, we are concerned that quite ordinary, everyday documents be considered equally, such as bills, consumer letters, instruction manuals, newspapers—these are the documents which users encounter in their everyday lives and understanding how they can be best structured could have significant practical benefits. The acquisition of annotated data across genre and cultures should also therefore be a high priority task.

Finally, we also require that the GeM annotation should

be able to fit into broader annotation schemes. Thus any kind of artifact that includes two-dimensional presentations (for example, a video embedded in a webpage) may also receive a GeM annotation for that component of the information offering. Our claims concerning coherence and consistency of information presentation decisions across text, visuals and layout can then be investigated here also. In such cases, the GeM annotation offers an annotation slice consisting of several annotation levels contributing to more comprehensive annotations that take in other important aspects of the artifact's design beyond that considered within the GeM model. In this respect, we consider it a crucial design feature that such annotation slices be additive and open rather than excluding and closed.

6. References

Elisabeth André. 1995. *Ein planbasierter Ansatz zur Generierung multimedialer Präsentationen*, volume 108. Infix, St. Augustin.

Roland Barthes. 1977. *Image – Music – Text*. Hill and Wang, New York.

John A. Bateman, Thomas Kamps, Jörg Kleinz, and Klaus Reichenberger. 2001. Constructive text, diagram and layout generation for information presentation: the DArt$_{bio}$ system. *Computational Linguistics*, 27(3):409–449.

Stephen Bernhardt. 1985. Text structure and graphic design: the visible design. In James D. Benson and William S. Greaves, editors, *Systemic Perspectives on Discourse, Volume 1*, pages 18–38. Ablex, Norwood, New Jersey.

Douglas Biber. 1989. A typology of english texts. *Linguistics*, 27:3–43.

Nadjet Bouayad-Agha. 1999. Annotating a corpus with layout. In Richard Power and Donia Scott, editors, *Proceedings of the AAAI Fall Symposium on Using Layout for the Generation, Understanding or Retrieval of Documents*, pages 58–61, Cape Cod, Massachusetts, November. American Association for Artificial Intelligence.

Nadjet Bouayad-Agha. 2000. Layout annotation in a corpus of patient information leaflets. In M. Gavrilidou, G. Carayannis, S. Markantonatou, S. Piperidis, and G. Stainhaouer, editors, *Proceedings of the Second International Conference on Language Resources and Evaluation (LREC'2000*, Athens, Greece. European Language Resources Association (ELRA).

Marc Corio and Guy Lapalme. 1998. Integrated generation of graphics and text: a corpus study. In M. T. Maybury and J. Pustejovsky, editors, *Proceedings of the COLING-ACL Workshop on Content Visualization and Intermedia Representations (CVIR'98)*, pages 63–68, Montréal, August.

Judy L. Delin and John A. Bateman. 2002. Describing and critiquing multimodal documents. *Document Design*, 3(2). Amsterdam: John Benjamins.

Michael A. K. Halliday. 1985. *An Introduction to Functional Grammar*. Edward Arnold, London.

Renate Henschel. 2002. GeM annotation manual. Gem project report, University of Bremen and University of Stirling, Bremen and Stirling. Available at http://purl.org/net/gem.

Gunther Kress and Theo Van Leeuwen. 2001. *Multimodal discourse: the modes and media of contemporary communication*. Arnold, London.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text*, 8(3):243–281.

Ivandré Paraboni and Kees van Deemter. 2002. Towards the generation of document deixis reference. In Kees van Deemter and Rodger Kibble, editors, *Information sharing: reference and presupposition in language generation and interpretation*, pages 333–358. CSLI.

Klaus Reichenberger, Klaas Jan Rondhuis, Jörg Kleinz, and John A. Bateman. 1995. Effective presentation of information through page layout: a linguistically-based approach. In *Proceedings of ACM Workshop on Effective Abstractions in Multimedia, Layout and Interaction*, San Francisco, California. ACM.

Karen A. Schriver. 1996. *Dynamics in document design: creating texts for readers*. John Wiley and Sons, New York.

Michael Twyman. 1982. The graphic presentation of language. *Information Design Journal*, 3:2–22.

Robert Waller. 1987. *The typographical contribution to language: towards a model of typographic genres and their underlying structures*. Ph.D. thesis, Department of Typography and Graphic Communication, University of Reading, Reading, U.K.

Towards a roadmap for Human Language Technologies: Dutch-Flemish experience

Diana Binnenpoorte[1,2], Catia Cucchiarini[2,3], Elisabeth D'Halleweyn[3], Janienke Sturm[2] and Folkert de Vriend[2]

[1]Speech Processing Expertise Centre (SPEX), Nijmegen, the Netherlands
[2]Department of Language and Speech, University of Nijmegen
Erasmusplein 1, Nijmegen, The Netherlands
{D.Binnenpoorte, C.Cucchiarini, F.deVriend, Janienke.Sturm}@let.kun.nl
[3]Nederlandse Taalunie, The Hague, The Netherlands
EdHalleweyn@ntu.nl

Abstract

In this paper we describe how the project "Dutch Human Language Technologies Platform" has contributed to creating the preconditions for establishing a roadmap for Human Language Technologies in the Dutch speaking area. Our overview of the results obtained so far reveals that the goals of all four action lines have been achieved and that there are clear directions for how to proceed in the near future. We hope that our experiences will be useful to other countries that intend to start similar initiatives.

1. Introduction

Establishing a roadmap for Human Language Technologies for a given language requires that first a number of important basic elements be defined, such as:

1. what is minimally required to guarantee an adequate digital language infrastructure for that language?
2. what is the current situation of HLT in that language?
3. what needs to be done to guarantee that at least what is required be available?
4. how can 3 best be achieved ?
5. how can we guarantee that once an adequate HLT infrastructure is available, it also remains so?

It is exactly these questions that were at the core of the activities that in the last two years were carried out within the framework of the Dutch-Flemish project "Dutch Human Language Technologies Platform". The ultimate aim of this project is to further the development and secure the usability of an adequate digital language infrastructure for Dutch, which is required to maximise the outcome of future efforts and to guarantee progress in the field of HLT.

In this paper we will report on our approach and our experiences in carrying out the activities envisaged in this project, because we think that this information can contribute to the aim of this workshop: establishing a roadmap for Human Language Technologies for the next decade.

2. The Dutch HLT Platform: action plan

The plan to set up a Dutch HLT platform was launched by the Dutch Language Union (Nederlandse Taalunie, NTU) which is an intergovernmental organisation established in 1980 on the basis of the Language Union Treaty between Belgium and the Netherlands. The NTU has the mission of dealing with all issues related to strengthening the position of the Dutch language (see also www.taalunie.org). In addition to the NTU, the relevant Flemish and Dutch ministries and organisations are involved in the HLT Platform. The various organisations have their own aims and responsibilities and approach HLT accordingly. Together they provide a good coverage of the various perspectives from which HLT policy can be approached.

The rationale behind the Dutch HLT platform was not to create a new structure, but rather to co-ordinate the activities of existing structures. The platform is a flexible framework within which the various partners adjust their respective HLT agendas to each other's and decide whether to place new subjects on a common agenda. Initially, the Dutch HLT platform was set up for a period of five years (1999-2004).

To achieve the objectives mentioned above, an *Action plan for Dutch in language and speech technology* was defined, which encompasses various activities organised in four action lines:

2.1. Action line A: performing a 'market place' function

The main goals of this action line are to encourage co-operation between the parties involved (industry, academia and policy institutions), to raise awareness and give publicity to the results of HLT research so as to stimulate market take-up of these results.

2.2. Action line B: strengthening the digital language infrastructure

The aims of action line B are to define what the so-called BLARK (Basic LAnguage Resources Kit) for Dutch should contain and to carry out a survey to determine what is needed to complete this BLARK and what costs are associated with the development of the material needed. These efforts should result in a priority list with cost estimates which can serve as a policy guideline.

2.3. Action line C: working out standards and evaluation criteria

This action line is aimed at drawing up a set of standards and criteria for the evaluation of the basic materials contained in the BLARK and for the assessment of project results.

2.4. Action line D: developing a management, maintenance and distribution plan

The purpose of this action line is to define a blueprint for management (including intellectual property rights), maintenance, and distribution of HLT resources.

Soon after the HLT Platform was set up it was decided that survey (action line B) and evaluation (action line C) be carried out in an integrated way because the actual availability of a product is not determined merely by its existence, but depends heavily on the quality of the product itself.

In the remainder of this paper we analyse the results of each action line in detail and in the final section we consider how this work has paved the way to a roadmap for Dutch HLT.

3. Action line A: results

In setting up HLT projects such as the *Spoken Dutch Corpus* and *NL-Translex*, much time was invested in the search for the appropriate responsible (funding) bodies in the Netherlands and Flanders. Moreover, various studies had indicated that the fragmentation of responsibilities made it difficult to conduct a coherent policy and meant that the field lacked transparency for interested parties. For these reasons the NTU, as the coordinator of the HLT Platform, stimulated the creation of a network aimed at:

disseminating the results of research in the field of HLT;

bringing together demand and supply of knowledge, products and services;

stimulating co-operation between academia and industry in the field of HLT.

After only two years of activity the HLT Platform has already produced important results. The success of Action line A is also partly due to the fact that the NTU acts as the National Focal Point (NFP) in the HOPE (Human Language Technology Opportunity Promotion in Europe) project. HOPE is a multi-country, shared-cost accompanying measure project of the IST-Programme of the European Commission that aims at providing awareness, bridge-building and market-enabling services to boost opportunities for market take-up of the results of national and European HLT RTD. The key focus is on helping to accelerate the volume of HLT transfer from the research base to the market by creating communities of interest between the critical players in the development and value chain. The aims of HOPE clearly coincide with the aims of Action line A.

At the beginning of the HOPE project an extensive informational website on the HLT sector in The Netherlands and Flanders was established by the NTU. This website provides up-to-date information on all relevant actors in the field of HLT (i.e. researchers, developers, integrators, users and policy makers) on how the HLT sector evolves on a cross-border Dutch/Flemish level, and on HLT related events throughout Europe. All this information is presented in Dutch and English.

The site also includes a calender of HLT events and a form for people who want to be included in the contacts database, as well as links to the HLTCentral website. All information on HLT related programmes and actions of the European Commission is provided on a separate website, established and maintained by subcontractor Senter/EG-Liaison, which is the most knowledgeable party on this subject. These two sites have one entry point from the HOPE point-of-view, via an intermediate site that was developed to provide clarity on where to find which information. This intermediate site (also in Dutch and English) has been placed on http://www.hltcentral.org/euromap/ and should be considered as the common homepage for the two websites. Visitors who do not find answers to their questions on the website can contact the NTU or Senter/EG-Liaison directly (preferably by e-mail) and may expect to receive quick and accurate replies.

Part of the infodesk task is also to conduct mailings to national contacts. These mailings are done on an ad-hoc basis, either at a third party's request (e.g. if an organizing committee wants to announce an event) or on the NFP's own initiative (e.g. if there is important news about an EC programme). From the beginning of the HOPE project, an extensive contacts database has been compiled by the NFP. At present, this database contains almost a thousand persons from over six hundred organisations in The Netherlands and Flanders. It is a valuable backbone for all information activities of the NFP.

The Dutch/Flemish NFP also visits companies with HLT related needs to demonstrate the benefits of HLT, to solicit a clear picture of the company's knowledge state and future plans, and to provide information of cross-linking services where appropriate. The NFP, in collaboration with its partners in The Netherlands and Flanders, has organised various seminars and workshops, which were attended by people from industry, academia, and policy institutions. The aim of such events is to further enhance awareness of recent developments in the HLT sector at the national and international level, such as the dissemination of information on European Commission HLT actions and their relevance to the national situation. Note that the cross-border Flemish/Dutch level should be considered here as the "national" level. The first national seminar took place in March 2001, and was a major event with over 150 participants. The second seminar was held in November 2001 and was directly related to the general survey carried out under action line B and C. Two other events are being organised for 2002. To conclude, we can safely state that in two years time the activities carried out within Action line A have certainly contributed to creating transparency and structure in the HLT field in The Netherlands and Flanders.

4. Results of Action lines B and C

The field survey comprised the following three stages: defining the BLARK for Dutch, making an inventory of available HLT resources, establishing a priority list. These three stages are described in more detail below.

4.1. Defining the BLARK

In defining the BLARK a distinction was made between applications, modules, and data:

Applications: refers to classes of applications that make use of HLT. The following classes were defined: CALL (Computer Assisted Language Learning), access control, speech input, speech output, dialogue systems, document production, information access, and multilingual applications or translation modules.

Modules: refers to the basic software components that are essential for developing HLT applications.
Data: refers to data sets and electronic descriptions that are used to build, improve, or evaluate modules.

In order to guarantee that the survey is complete, unbiased and uniform, a matrix was drawn up by the steering committee describing (1) which modules are required for which applications, (2) which data are required for which modules, and (3) what the relative importance is of the modules and data. This matrix (subdivided in language technology and speech technology) is depicted in Table 1, where "+" means important and "++" means very important.

This matrix serves as the basis for defining the BLARK. Table 1 shows for instance that monolingual lexicons and annotated corpora are required for the development of a wide range of modules; these should therefore be included in the BLARK. Furthermore, semantic analysis, syntactic analysis, and text pre-processing (for language technology) and speech recognition, speech synthesis, and prosody prediction (for speech technology) serve a large number of applications and should therefore be part of the BLARK, as well. Note that only language specific modules and data were considered in this survey.

Based on the data in the matrix the BLARK for Dutch should consist of the following components:

4.1.1. Language technology BLARK

Modules:
- Robust modular text pre-processing (tokenisation and named entity recognition),
- Morphological analysis and morpho-syntactic disambiguation,
- Syntactic analysis,
- Semantic analysis.

Data:
- Monolingual lexicon,
- Annotated corpus written Dutch (a treebank with syntactic, morphological, and semantic structures),
- Benchmarks for evaluation.

4.1.2. Speech technology BLARK

Modules:
- Automatic speech recognition (including tools for robust speech recognition, recognition of non-natives, adaptation, and prosody recognition),
- Speech synthesis (including tools for unit selection),
- Tools for calculating confidence measures,
- Tools for identification (speaker identification as well as language and dialect identification),
- Tools for (semi-) automatic annotation of speech corpora.

Data:
- Speech corpora for specific applications, such as CALL, directory assistance, etc.,
- Multi-modal speech corpora,
- Multi-media speech corpora,
- Multi-lingual speech corpora,
- Benchmarks for evaluation.

4.2. Inventory and evaluation

In the second stage, an inventory was made to establish which of the components - modules and data - that make up the BLARK are already available; i.e. which modules and data can be bought or are freely obtainable for example by open source. Besides being available, the components should also be (re-)usable. Obviously, components can only be considered usable if they are of sufficient quality; therefore, a formal evaluation of the quality of all modules and data is indispensable. Given the limited amount of time, only a formal evaluation was carried out by using a checklist with the following items: availability, programming code, platform, documentation compatibility with standard packages, reusability, adaptability and extendibility.
The information on availability, the matrix in Table 1 and the preliminary inventory were submitted to a group of HLT experts from both industry and academia, so that a balanced picture could be obtained.

Based on this information a second matrix was filled in which the availability of the components in the BLARK (cf. Table 2) was described. Availability in this matrix is expressed in numbers from 1 (‘module or data set is unavailable’) to 10 (‘module or data set is easily obtainable’).

At the end of the second stage, all information gathered was incorporated in a report containing the BLARK, the availability figures together with a detailed overview of available HLT resources for Dutch, a priority list of components that need to be developed, and a number of recommendations. This report was considered as being provisional as feedback on this version from a lot of actors in the field was considered desirable.

4.3. Feedback

One of the aims of Action lines B and C was that the majority of the actors in the HLT field would subscribe to the priorities and recommendations for the future. To this end, the provisional report containing the inventory, the priority lists and the recommendations was sent to a total of about 2000 people active in the HLT field who were asked to send their comments by email. After the relevant comments had been incorporated in the report, the same group of people was invited to participate in a workshop in which the results (overview, BLARK, priority lists and recommendations) were officially presented to the public.

On this occasion some people were given the opportunity to publicly present their views on the results of the survey. The workshop was concluded with a general discussion between the audience and a panel of five experts that were responsible for the survey.

The workshop provided useful information that could be used to complete the final report. A number of important points that emerged form this workshop are listed below:
- Cooperation between universities, research institutes and companies should be stimulated.
- For all components in the BLARK it should be clear how they can be integrated with off-the-shelf software packages. Furthermore, documentation and information about performance should be readily available.
- Control and maintenance of all modules and data sets in the BLARK should be guaranteed.
- Feedback from users on the quality and the performance of the various components should be processed in a structured way.

Special attention should be paid to the issue of open source policy and its possible effects for companies.

Modules	Data									Applications							
	mono lex	multi lex	thes	ann corp	unann corp	speech corp	multi ling	multi mod	multi media	CALL	access control	speech input	speech output	dialog system	doc prod	info access	transla-tion
Language Technology																	
Grapheme-phoneme conv.	++			++						+			++	++	+	+	
Token detection	++			+	++					+		+		+	+	+	+
Sent boundary detection	+			++	++					+		++	++	+	++	++	++
Name recognition	+	+	+	++	++	++				+		++	++	+	++	++	++
Spelling correction										+							
Lemmatizing	++			++	+					+		+	+	+	+	+	+
Morphological analysis	++			++	+					+		+	++	+	++	++	++
Morphological synthesis	++			++	+					+			++	+	++		++
Word sort disambig.	++			++	+					+		++	+	++	++	++	++
Parsers and grammars	++			++						+		++	++	++	++	++	++
Shallow parsing	++			++	++					+		++	++	++	++	++	++
Constituent recognition	++			++	+					+		++	++	++	++	++	++
Semantic analysis	++		++	++				++	++	+		++	++	++		++	++
Referent resolution	+		++	++	+					+		++		++	++	++	++
Word meaning disambig.	+		++	++	+					+		++	+	+	+	++	++
Pragmatic analysis	+		+	++				++	++	+		++	++	++		+	++
Text generation	++		++	++				++	++	+			++	++	++		++
Lang. dep. translation		++	++	++			++			+						++	++
Speech Technology																	
Complete speech recog.	++	+		++	+	++	+	++	++	++	++	++		++	++	++	++
Acoustic models	++	+		++	+	++	+	+	+	++	+	++		++	+	+	+
Language models	+			++	+	+	+	+	+	++	+	++		++	++	++	++
Pronunciation lexicon	++	+		+		++	+	+	+	++	+	++	+	++	+	++	++
Robust speech recognition	+			+	+	+	+	+	++	+	+	++		++	+	+	+
Non-native speech recog.	+	++		+		++	++	+	+	++	+	+		+		+	+
Speaker adaptation	+			+	+	++	+	+	++	+	+	++		+	+	++	+
Lexicon adaptation	++	+		+		++	+	+	+	++	+	++	+	++	+	++	++
Prosody recognition	+	+		++	+	++	+	+	+	++	+	++		++	++	++	++
Complete speech synth.	++	+		+		+		+		+			++	++	+	+	++
Allophone synthesis	+	+		+		+		+		+			+		+	+	+
Di-phone synthesis	++	+		+		+		+		+			++	++	+	+	+
Unit selection	++	+		+		+		+		+			++	++	+	+	+
Prosody prediction for Text-to-Speech	++	+		+		+		+	+	++			++	++		+	++
Aut. phon. transcription	++	++		+	+	++	+	+	+	++	+	+	+	+	+	+	+
Aut. phon. segmentation	++	++		+	+	++	+	+	+	++	+	+	+	+	+	+	+
Phoneme alignment	+	+		+		++	+	+	+	++	+	+		+			+
Distance calc. phonemes	+	+		+		++	+	+	+	++	+	+		+			+
Speaker identification	+			++	++	++	+	++	+	+	++	+		+		+	+
Speaker verification	+			++	++	++	+	++		+	++	+		+		+	+
Speaker tracking	+			++		++			++	+	++	+		+	+	+	+
Language identification	+	++		+	+	++	++	+	+	+	+	+		+		+	+
Dialect identification	+	++		+	+	++	++	+	+	+	+	+		+		+	+
Confidence measures	+			+	+	++	+	++	+	++	++	++		++	+	+	+
Utterance verification	+			+	+	++	+	+	+	+	+	++		++	+	+	+

Table 1 *Overview of the importance of data for modules and the importance of modules for applications.*

Modules	Availability
Grapheme-phoneme conversion	8
Token detection	9
Sentence boundary detection	3
Name recognition	4
Spelling correction	3
Lemmatizing	9
Morphological analysis	
Morphological synthesis	
Word sort disambiguation	7
Parsers and grammars	3
Shallow parsing	2
Constituent recognition	5
Semantic analysis	3
Referent resolution	2
Word meaning disambiguation	2
Pragmatic analysis	1
Text generation	3
Language dependent translation	3
Complete speech recognition	4
Acoustic models	8
Language models	3
Pronunciation lexicon	5
Robust speech recognition	2
Non-native speech recognition	2
Speaker adaptation	2
Lexicon adaptation	2
Prosody recognition	2
Complete speech synthesis	6
Allophone synthesis	7
Di-phone synthesis	6
Unit selection	1
Prosody prediction for Text-to-Speech	3
Autom. phonetic transcription	3
Autom. phonetic segmentation	5
Phoneme alignment	8
Distance calculation of phonemes	8
Speaker identification	2
Speaker verification	2
Speaker tracking	2
Language identification	2
Dialect identification	2
Confidence measures	2
Utterance verification	2
Data	
Unannotated corpora	9
Annotated corpora	5
Speech corpora	4
Multi lingual corpora	3
Multi modal corpora	1
Multi media corpora	1
Test corpora	1
Monolingual lexicons	8
Multilingual lexicons	6
Thesaurus	4

Table 2 Availability of modules and data

4.4. Inventory, priority list and recommendations

The survey of Dutch and Flemish HLT resources resulted in an extensive overview of the present state of HLT for the Dutch language. By combining the BLARK with the inventory of components that are available and of sufficient quality, the following priority for language and speech technology lists were drawn up.

4.4.1. Priority list for language technology:

1. Annotated corpus written Dutch: a treebank with syntactic and morphological structures,
2. Syntactic analysis: robust recognition of sentence structure in texts,
3. Robust text-preprocessing: tokenisation and named entity recognition,
4. Semantic annotations for the treebank mentioned above,
5. Translation equivalents,
6. Benchmarks for evaluation.

4.4.2. Priority list for speech technology:

1. Automatic speech recognition (including modules for non-native speech recognition, robust speech recognition, adaptation, and prosody recognition),
2. Speech corpora for specific applications (e.g. directory assistance, CALL),
3. Multi-media speech corpora (speech corpora that also contain information from other media such as newspapers, WWW, etc.),
4. Tools for (semi-) automatic transcription of speech data,
5. Speech synthesis (including tools for unit selection),
6. Benchmarks for evaluation.

On the basis of the inventory and the reactions from the field the following recommendations were made:

- existing parts of the BLARK should be collected, documented and maintained by a central institution;
- the BLARK should be completed by financing the development of the resources prioritised;
- the BLARK should be made available to industry and academia through open source development;
- benchmarks, test corpora, and methods for evaluation and validation should be developed.
- the training of qualified HLT researchers should be encouraged.

5. Results of Action line D: the HLT Blueprint

In many cases official bodies such as ministries and research organisations are prepared to finance the development of language resources and no longer feel responsible for what should happen to these materials once the project has finished. However, materials that are not maintained quickly lose value. Moreover, unclear intellectual property right arrangements can create difficulties for exploitation. The purpose of action line D was to draw up a blueprint for management, maintenance and distribution of basic language materials that have been developed with government money. This includes, among other things, dealing with intellectual property rights issues, with the acquisition of resources, the adaptation of data and modules to other systems and applications,

making documentation available, providing a help desk function, maintaining and updating the material. Finally, this blueprint should provide guidelines for organizing a structural form of co-operation in this respect and should serve as an instrument for field organisations as well as for funding bodies.

The *Blueprint for management, maintenance and distribution of digital materials developed with public money (Blueprint)*, P. van der Kamp, T. Kruyt en P.G.J. van Sterkenburg) was prepared in the period 2000 -2001 by a team of language technology experts of the Institute for Dutch Lexicology, INL. In addition to the general aim of providing guidelines for the acquisition, management, maintenance and distribution of HLT materials, the *Blueprint* aims at providing information to be used by policy organisations when assessing research projects aimd at developing HLT materials, for preparing policy plans concerning the acquisition, management, maintenance and distribution of HLT materials and practical information on how to acquire, manage, maintain and distribute HLT materials, answers to questions concerning the (re)usability of HLT materials after the consortia that were set up for their development cease to exist. All this information is presented in the *Blueprint* in nine chapters that, apart from the introductory chapter 1, deal with the following topics:

- Acquisition of HLT resources (Chapter 2)
- Processing of acquired data (Chapter 3)
- Linguistic processing of HLT resources (Chapter 4)
- Management of HLT resources (Chapter 5)
- Maintenance of HLT resources (Chapter 6)
- Distribution of HLT resources (Chapter 7)
- Support to users (Chapter 8)
- Recommendations for future policy (Chapter 9)

The following eight recommendations for future policy are made in the final chapter:

1. An HLT agency is necessary
   In order to prevent that HLT materials developed with government money outside a permanent infrastructure become obsolete and therefore useless, a legal body such as an HLT agency is required.
2. Organisation form of HLT agency and role of NTU
   This HLT agency could be a Dutch-Flemish consortium of institutions and should not be related to one existing institution in particular, because not all expertise is available in one single institution. A co-ordinator could be appointed by NTU to ensure that the interests of the whole HLT field are represented.
3. Tasks of the HLT agency.
   Primary tasks of an HLT agency:
   Task 1. Management
   Task 2. Guarantee accessibility of data and software
   Task 3. Maintenance
   Secondary tasks of an HLT agency:
   Task 4. User support
   Task 5. Acquisition
   Distribution should be entrusted ELDA and LDC.
4. Costs to be met by the government.
   Since extra costs for personnel and hardware will be incurred, additional government funding is required.
5. Costs to be met by the users of the HLT agency
   Depending on the specific use and user, general conditions must be agreed on that guarantee fair tariffs.
6. Acceptance of HLT data and software by the HLT agency.
   The HLT agency can refuse HLT resources that do not meet certain quality standards or that are not essential for a wide range of applications.
7. International participation.
   The HLT agency should be given the possibility, through government funding, to participate in European and/or global projects that are related to its tasks.
8. Development and maintenance of HLT expertise.
   Given the considerable shortage of language and speech technologists, the government should stimulate policies that are aimed at developing and maintaining expertise in the field of HLT.

6. Future prospects

In the previous sections we have provided an overview of the results obtained within Action lines A and D. This has revealed that the aims identified in the *Action plan for Dutch in language and speech technology* have been achieved, at least for these two action lines. Now it remains to be seen how these results will be used in the future in order to achieve the ultimate aim of the "Dutch Human Language Technologies Platform" project: to further the development and secure the usability of an adequate digital language infrastructure for Dutch. To this end in the following sections we consider our future plans with respect to Action lines A (5.1) and D. (5.2).

6.1. Action line A

Since Action line A has already contributed to creating a co-operative framework in the HLT field in The Netherlands and Flanders, our future activities will be directed to maintaining and enlarging it. This entails among, other things, keeping our databases and websites up to date, ensuring communication between interested partners, gradually enlarging the initial network, identifying and promoting the inclusion of new representatives; increasing the visibility and the strategic impact of relevant results and new initiatives; fostering cooperation; providing a forum for discussing, exchanging and sharing experiences, best practices, information data and tools.

6.2. Action lines B and C: HLT priorities

The future activities of these two action lines will be directed to ensuring that the priorities identified in the survey are realized so that an adequate HLT infrastructure for Dutch is obtained.

6.3. Action line D: implementation of the recommendations in the HLT Blueprint

In the near future a number of Dutch-Flemish digital HLT resources will become available. These development projects, in many cases, do not provide a permanent infrastructure. As projects aimed at the development of digital basic resources mostly result in intermediary products, extra efforts and investments are needed in order to implement them in applications that find their way to the end users. Furthermore, when planning such large scale projects a lot of time is invested in building the

necessary structures (often at a supra-institutional level) and finding the right experts. The completion of a project often means that the managerial and operational structures cease to exist. Therefore it is of vital importance that the right measures are timely taken in order to ensure that the resources are stored in such a way that they will be expertly managed and maintained. When establishing an adequate infrastructure for maintenance of digital basic resources, proper attention should be given to a) intellectual rights, overall responsibility and co-ordination, b) actual physical management and maintenance of the resources and c) maintenance of expertise. In the following sections we will describe the facilities that we envisage to implement in the Dutch speaking area in the near future.

6.3.1. Necessary facilities

A. Intellectual rights, responsibility, co-ordination: ***NTU***

A careful transfer of intellectual rights is of crucial importance to the exploitation of resources. Furthermore, after completion of projects a visible policy responsibility is needed, even if the actual management and maintenance is carried out by an HLT agency (see B).

Organisational structure: The NTU (Nederlandse Taalunie/Dutch Language Union), representing a permanent Dutch-Flemish infrastructure, can act as the appropriate legal body handling all legal affairs. A member of the NTU will be appointed as co-ordinator and supervise from a policy point of view management, maintenance and exploitation of HLT basic resources that are contributed to the HLT agency (see B)..

The NTU will look after the interests of the entire HLT field and will function as a kind of 'broker' by:

- supervising the activities of the HLT agency (see B) and the various HLT committees (see C);
- looking after legal issues;
- stimulating the application of international standards;
- stimulating funding bodies to stipulate that in proposals proper attention is paid to allocating funding for management and maintenance and that resources financed with public funding be made available through the HLT agency;
- playing an intermediate role in the acquisition of digital data, e.g. from the industry.

B. Management and maintenance of digital resources: HLT agency

The *Blueprint* recommends the co-operation of the institutes in a consortium, an **HLT agency**, as this makes it possible to use dispersed expertise and infrastructure. This construction clearly has a number of advantages:

- efficient use of persons and means can be cost-reducing;
- combining resources and bringing together different kinds of expertise can create surplus value (e.g. extra applications);
- offering resources through one window (one-stop-shop) will create optimal visibility and accessibility;
- in international projects the Dutch language area can act as a strong partner;

Organisational structure: The HLT agency can take the form of a Dutch-Flemish consortium of organisations contributing their resources and expertise in a virtual resource centre. These organisations should strike binding agreements for a determined period of time. One Dutch-Flemish organisation (e.g. the Dutch Institute of Lexicology in Leiden) should be appointed as responsible co-ordinator.

- management: taking the appropriate (mostly technical) measures so as to make sure that data and software remain operational and usable;
- accessibility data and software: facilitating reusability of HLT resources: e.g. technical, legal and administrative settlements so as to optimise the route from developer via HLT agency to the distributor;
- maintenance: taking the appropriate measures to ensure long-term usability of data and software: technical maintenance of formats of HLT data, HLT software, system and application software, equipment; maintenance of legal contracts; content management of the HLT data and annotations;
- service: help desk, service to the users of the HLT data and HLT software (e.g. advising, maintenance of website and mailing lists, supplying tailor made data or software on demand);
- acquisition: active acquisition of HLT data and HLT software developed by the industry or research institutes;
- evaluation and validation: contributing to establishing international standards and methods for evaluating and validating HLT resources.

For the actual, physical distribution of the resources appeal will be made on the expertise of organisations s.a ELRA and LDC as they have the proper expertise and marketing tools.

C. Expertise: Dutch-Flemish steering committees and HLT management committee

In dissolving the managerial and operational infrastructure after the completion of a project, valuable specific knowledge concerning the project may be lost causing difficulties in the exploitation of the results. All the same it would not be realistic to maintain these structures. A solution would be to install a number of Dutch-Flemish **steering committees** and one co-ordinating Dutch-Flemish **HLT management committee**. The tasks of these committees should not be too heavy, but to ensure continuity and effectiveness a strong secretarial support should be provided

Organisational structure: For each completed large scale project the results of which are contributed to the HLT agency, a steering committee should be installed. Each steering committee delegates one representative to a co-ordinating HLT management committee. For small scale projects it has to be examined whether the necessary expertise is already present in the HLT management committee. Probably one expert, responsible for the combined 'small' projects, will be added to this committee. The various committees should receive the appropriate secretarial support.

Tasks: The steering committees will be responsible for specific resources and specific domains. They will

- act as a knowledge base for questions concerning the resources contributed to the HLT agency;
- act as intrinsic supervisors on management, maintenance and exploitation of specific resources;
- act as advisors in specific domains s.a. language and speech technology, terminology, lexicology;
- be instrumental in the organisation of 'major repairs' of the resources that are put in their custody;
- be instrumental in developing the appropriate infrastructure for new projects or updating of existing results in their domain.

The HLT management committee will be responsible for the co-ordination, overall management, maintenance and distribution of HLT resources. It will

- act as general knowledge base and give advise in the broad field of language and speech technology, terminology, lexicology etc..
- act as general intrinsic supervisor on management, maintenance and exploitation of finished resources;
- be instrumental in developing the appropriate personnel infrastructure for new projects or updating of existing results.

6.3.2. Financing

Since the exploitation of basic resources will not result in considerable revenues, the authorities have expressed their explicit wish to make these resources available as broadly as possible. This results in keen prices: cost price for non-commercial research, a higher but not prohibitive price for commercial organisations. Consequently, the implementation of the above mentioned structures requires extra funding. Since a considerable percentage of the development costs should be allocated to management and maintenance, by combining the infrastructures required for different projects the percentage the costs would decrease. This applies as much to the material infrastructure (equipment, data, software, licences, etc…) as to the immaterial infrastructure (experts, personnel etc.). As is stressed in the recommendations of the *Blueprint*, the activities of the HLT agency cannot be carried out by the consortium partners in addition to their daily work, but require extra staff. Based on the data in the *Blueprint* and on experiences in other projects, a number of persons will be appointed at one or more of the organisations forming the HLT agency (e.g. experts on language and speech technology, IT-specialist, administrative personnel etc.). One overall co-ordinator and at least one secretary of the committees will be appointed at the NTU.

It is to be expected that the costs will increase with the increase of project results contributed to the HLT agency. These costs should be covered with funds allocated to management, maintenance and accessibility at the start of the development of new projects.

6.3.3. Conclusions

After the completion of projects aimed at developing HLT resources, efforts are needed to ensure long-term usability of the results. Timely attention to intellectual property rights, management, maintenance and distribution can guarantee that investments pay off in the future. In this respect, it is recommended, to make optimal use of existing expertise and infrastructure. In concrete this would mean that in the Dutch speaking area:

- the co-ordinating policy responsibility and as much intellectual property rights as possible should be placed in the hands of the NTU;
- the actual exploitation (management, maintenance and distribution) should be entrusted to a Dutch-Flemish HLT agency, that will take the shape of a consortium of institutions but acts as a one-stop-shop of digital HLT resources for the Dutch language
- the existing expertise should be combined as much as possible in a number of Dutch-Flemish steering committees consisting of representatives of projects, the results of which are contributed to the HLT agency and a co-ordinating Dutch-Flemish HLT management committee.

The NTU envisages to implement the above mentioned structures in its new long-term policy plan (2003-2007).

7. General conclusions

In this paper we have reported on the activities that in the last two years have been carried out within the framework of the project "Dutch Human Language Technologies Platform". In particular, we have focussed on two of the four action lines within this project: Action line A, which was aimed at raising awareness of the results of HLT research and promoting communication among interested partners, and Action line D which was concerned with management, maintenance and distribution of HLT resources.

Our overview of the results obtained so far has revealed that a cooperative framework has been created and that there are clear plans to set up a structure that will take care of all HLT resources developed with public funding, so that they will remain available for all interested parties: an HLT agency. In other words, the goals of action lines A and D have been achieved (for the results of B and C, the reader is referred to Binnenpoorte et al. (2002)) and clear directions for how to proceed in the near future have also been outlined. To conclude, it seems that in the Dutch speaking area pioneering work has been carried out from which other countries can probably profit in their attempts to start similar initiatives.

8. Acknowledgements

We are indebted to the steering committees of Action lines B, C, and D and to the authors of the Blueprint (P. Van der Kamp, T. Kruyt, and P. Van Sterkenburg) and of the Report B and C (G. Bouma, W. Daelemans, A. Dirksen, D. Heijlen, F. de Jong, J.-P. Martens, A. Nijholt, H. Strik, D. van Compernolle, F. van Eynde, and R. Veldhuis) for their invaluable contribution to the work presented in this paper.

9. References

Binnenpoorte, D., de Vriend, F., Sturm, J., Daelemans, W., Strik, H., and Cucchiarini, C. (2002). A Field Survey for Establishing Priorities in the Development of HLT Resources for Dutch. In *Proceedings of LREC2002*.

ANNEXES

SPEECH-RELATED TECHNOLOGIES

Where will the field go in 10 years?

Niels Ole Bernsen, NISLab, Denmark (editor)

Abstract

This paper is a draft position paper for discussion at the ELSNET Brainstorming Workshop 2000-2010 in Katwijk aan Zee, The Netherlands, on 23-24 November, 2000. The paper first describes some general emerging trends which are expected to deeply affect, or even transform, the field of speech technology research in the future, including trends towards advanced systems research, natural interactivity, multimodality, and medium-scale science. A timeline survey of future speech-related technologies is then presented followed by analysis of some of the implications of the proposed timelines. Timeline projections may turn out to have been false, of course, but even their turning out to be true is subject to future actions which are (not) taken to make them true. Accordingly, the final part of the paper discusses some actions which would seem desirable from the point of view of strengthening the position of European speech-related research.

1. Introduction

The term *speech-related research* has been chosen to designate the topic of the present paper for lack of ability to invent a more appropriate term, if there is one. At least, the term partly manages to convey the author's expectation that the field of speech research will change rather dramatically in the coming ten years as speech technologies become merged with other technologies into a field which, so far, lacks a name.

According to many observers, the coming decade will be the decade of speech technologies. Computer systems, whether stationary or mobile, wired or wireless, will increasingly offer users the opportunity to interact with information and people through speech. This has been made possible by the arrival of relatively robust, speaker-independent, spontaneous (or continuous) spoken dialogue systems in the late 1990s as well as through the constantly falling costs of computer speed, bandwidth, storage, and component miniaturisation. The presence of a speech recogniser in most appliances combined with distributed speech processing technologies will enable users to speak their native tongue when interacting with computer systems for a very large number of purposes. Although no doubt exaggerated as just presented, there probably is some truth to this vision of a breakthrough in the application of speech technologies in the coming years. If this is the case, it would seem worthwhile that we lift our sights and take a long-term view of the issues ahead. This may help setting a reasonable research agenda for the coming years of advanced speech systems research and development, one which does not succumb to the usual hype associated with fashionable technologies. Today, some believe that "the speech problem" has been solved already. Some believe that speech, because of its naturalness, is the solution to every conceivable problem of user-system interaction. On the other hand, surprising as it may seem, some human factors and interactive systems experts believe that we have just arrived at the touch-tone telephony stage and share no notion of the actual state-of-the-art in the field with its practitioners. Since

all of those beliefs are far from the truth, it is important to provide a more balanced picture of the state-of-the-art in speech technologies in order to set the stage for solid progress.

In what follows, Section 2 presents some trends in the speech-related research field. Section 3 excels in guesswork by estimating the times of appearance of a range of novel speech-related technologies. Section 4 discusses implications of the timelines presented in Section 3. Section 5 proposes a series of actions which would appear appropriate given the preceding discussion.

2. Some Trends

The speech field is making progress on a broad scale as demonstrated by the 900 or so papers and posters presented at the recent International Conference on Spoken Language Processing (ICSLP) in Beijing, October 2000. [To be illustrated by listing topics.] Three points may be made on the preceding list of current topics in speech research. Firstly, the wealth of topics that are being addressed in current fundamental and applied research obviously demonstrates that "the speech problem" has *not* been solved but continues to pose a series of major research challenges. [Mention some of them.] Secondly, the breadth of the speech topics that are being addressed could be taken as evidence that the speech field is simply doing *business as usual,* albeit on a larger and more ambitious scale than ever before. Thirdly, however, it is clear from the topics list that *the speech field is no longer separate from many other fields* of research but is in a process of merging into something which might perhaps be called the general field of interactive technologies. This latter trend, it may be argued, is the single most important factor which will influence the speech field in the future and which already suggests that the field is in a state of profound transformation.

Interactive technologies

It is relatively straightforward to explain why the speech field is gradually merging into the general field of interactive technologies. Since speech now works for a broad range of application purposes, a rapidly growing fraction of the speech research community are becoming involved in advanced interactive *systems* research rather than continuing to work on improving the speech *components* which form part of those systems. In advanced interactive systems research, speech is increasingly being used not as a stand-alone interactive modality as in, e.g., spoken language dialogue systems over the telephone, speech dictation systems, or text-to-speech systems, but as a modality for exchanging information with computer systems in combination with other modalities of information representation and exchange. Moreover, speech is not just an interactive technology among many others. Spontaneous speech is an extremely powerful input/output modality for interacting with computer systems, a modality which, furthermore, is available and natural to the large majority of users without any need for training in using it for interactive purposes.

The ongoing shift from speech components research to research on integrating speech in complex interactive systems has a number of important implications for the speech field. Speech researchers are becoming systems researchers and engineers. Far more than components research, systems research and engineering is exposed to the full complexity of today's world of information and telecommunications technologies. Few, if any, groups can build full systems on their own from scratch. To stay competitive, they have to *follow closely* the global developments in relevant systems architectures, platforms, toolkits, available components of many different kinds, de facto standards, work in standards committees, market trends etc. They need larger and much more *interdisciplinary teams* in order to keep up with competitive developments. They need *access* to platforms and component technologies in order to avoid having to do everything by themselves. And they need expertise in *software systems engineering best practice as specialised to the kind of systems*

they are building, including expertise in systems and usability evaluation. As we shall see in Section 4, they need even more than this, such as *hardware* access or expertise, development *resources, behavioural research* in new domains, and skills in *form and contents design.*

Compared to traditional research on improving a particular speech component technology, the world of advanced interactive systems research would appear to be orders of magnitude more complex. Moreover, that world is quite diffuse for the time being. It does not have a single associated *research community,* being inhabited instead by researchers from most traditional ITC (Information Technologies and Telecommunications) research communities. The world of advanced interactive systems research does not have any clear *evolutionary direction,* being characterised rather through ever-changing terms of fashion, such as 'ubiquitous computing', 'things that think', 'wearable computing', 'the disappearing computer' or 'ambient intelligence'. Significantly, all or most of those terms tend to refer to combined hardware and software systems rather than to components, and none of them refer to the traditional communities in the ITC field, such as speech processing, natural language (text) processing, machine vision, robotics, computer graphics, neural networks, machine learning, or telecommunication networks. Indeed, most of our current stock of inspired and visionary terms for describing the future of interactive technologies tends to be rather vague with regard to the technologies which they include or, if any, exclude.

Rather than trying to clarify what might be meant by the terms of fashion mentioned above, it may be useful to look at two other developments in conceptualising the field of advanced interactive systems research of which speech research has begun to form a part. To be sure, the concepts to be discussed are expressed by fashion terms as well, but at least it would seem that those concepts are of a more systematic and theoretically stable nature at this point.

Natural interactivity

When being together, most humans interact through speech when they exchange information. The telephone allows them to use spoken interaction at a distance as well, and the function of the telephone will soon be shared, or even taken over, by computing systems. When humans interact through speech, it does not matter if they are just a twosome or if they are more than two together. Moreover, except when speaking over the telephone, speech is not their only modality for information exchange. Gesture, lip movements, facial expression, gaze, bodily posture, and object manipulation all contribute to adding information, however redundant, to the spoken message. Together with speech, those modalities constitute *full natural human-human communication.* Moving beyond current technologies, we envision not just a single human speaking on the telephone or to a (desktop) computer in order to get a particular task done. Rather, the vision is one in which multiple humans speak together whether or not they are in the same physical location whilst using the system as an increasingly equal partner in communication. The system mediates their communication when needed, understands full natural communication, and produces full natural communication itself, increasingly acting as its human counterparts in communication. In order to take this vision into account, it would seem timely to abandon the traditional model of interaction which is called 'human-computer interaction', and replace it with the more general model of *natural human-human-system interaction* (HHSI). Natural HHSI, it appears, it a necessary end-point of current research in speech technologies. Thus, natural interactivity may serve as an important, even if distant, guidepost for the role of speech research in the complex world of interactive systems research.

The received picture of the role of theory in engineering goes something like this. It is hardly ever possible to deduce from theory a complete specification of the artefact that would constitute an optimal solution to some engineering problem. The reason is that the complexity

of the problem space involved always exceeds the power of theory. On the other hand, without theory (of physics, chemistry, computation etc.), it would not have been possible to build many of the artefacts we use in our daily lives. Thus, theory has a necessary supporting function in engineering. This is clear in the case of natural interactivity. To achieve the ultimate goal of natural HHSI, we need far better theory than is available at present: about how humans behave during natural interaction, about the behavioural phenomena which are relevant to the development of fully natural interactive systems, about how these phenomena are interrelated, about how they should be encoded etc. We also need a novel theory of natural communication which can replace speech acts theory and discourse theory by taking the notion of a complete communicative act as its basic notion.

Multimodality

The trend towards multimodal interactive systems reflects the trend towards blending of traditional research communities noted above as well as the increasing role of speech in future interactive systems. Multimodal systems are systems which offer the user combinations of input/output modalities for (or ways of) exchanging information with computer systems. Given the naturalness and expressive power of speech, speech input and speech output have the potential for becoming key modalities in future interactive systems. However, compared to natural interactivity, our current understanding of multimodality is much less capable of providing guideposts for future advanced interactive systems research in general and research on multimodal systems which include speech modalities in particular. Much too little is known about how to create good modality combinations which include speech for a variety of interactive purposes. This topic has become an active field of research, however (Bernsen 1997a, Benoit et al. 2000, Bernsen 2001). Further progress in this field is likely to complement research on natural interactivity in providing guideposts for speech-related research in the complex world of advanced interactive systems. In fact, these two research directions are intertwined in so far as it remains an open issue for which application purposes technologies, such as, e.g., animated speaking characters might provide useful solutions.

Medium-scale science

The final trend to be mentioned is the trend towards medium-scale science in advanced interactive systems research. Increasingly, it is becoming evident that the standard 3/4/5-team, low-budget, 3-year isolated advanced systems research project is often an inefficient means of achieving significant research progress. In many projects, the participants share discouraging experiences, such as the following: even if small, the project is only able to start almost one year after its conception because of the administrative processing needed to release the funding for the project; when the project begins, the participants discover that their objectives have already been achieved elsewhere; the participants spend the first half of the project trying to identify the best platform to work from only to discover that they cannot get access to it; the participants spend half of the project building and putting together a low-quality version of the contextual technologies they need before they can start addressing their core research objectives; at the start of the project, the participants realise that it will take too long to produce the data resources they need, such as tagged corpora, and decide instead to work with sub-optimal resources which they can get for free; etc. One way to avoid, or reduce the number of, such experiences is to launch larger-scale concerted research efforts which have a better chance of moving beyond the state of the art. World-wide, experiments are currently underway on how to carry out such medium-scale science. In the US DARPA Communicator project which addresses spoken language and multimodal dialogue systems, for instance, all participants start from shared core technologies without having to build these themselves (http://fofoca.mitre. org/). In the German SmartKom project which addresses multimodal

communication systems, the budget is large enough for the participants to build and integrate the technologies needed (http://smartkom.dfki.de/start.html). In the European Intelligent Information Interfaces (i3, http://www.i3net.org/) and CLASS (http://www.class-tech.org/) initiatives, whilst the traditional 3-year small-scale project topology has been preserved, major efforts are being made to promote cross-project collaboration, synergy, and critical mass.

For reasons too obvious to mention, relatively small-scale research should continue to exist, of course. Still, the complexity of the world of advanced interactive systems research is not likely to go away. This raises the question of whether we need more medium-scale science and less small-scale science in order to make efficient use of the funds available for advanced interactive systems research. If this question is answered in the affirmative, the important issue becomes how best to do medium-scale science, i.e. which model(s) to adopt for the larger-scale research efforts to come.

3. Estimated Technology Timelines

This section attempts to estimate the time of first appearance of a broad selection of generic and/or landmark speech technologies including natural interactivity technologies and multimodal technologies involving speech. Some qualifications are necessary to the proper interpretation of the proposed predictions. Despite the numerous uncertainties involved in estimating technology progress, timelines, when properly estimated, qualified, and peer reviewed, do seem a useful means of conveying a field's expectations to the outside world and serving as a basis for actions to be undertaken to support research in the field.

Qualifications

(a) As in all timeline forecasts, there is some uncertainty in the forecasts below with respect to whether the technology is deployable or will in fact have been deployed in products at the suggested time. The claim for the figures below rather tend towards the *deployable* interpretation which is the one closest to the point of view of research. The *actual deployment* of a deployable technology is subject to an additional number of factors some of which are unpredictable, such as company technology exploitation strategies, pricing strategies, and the market forecasts at deployability time. Thus, several years may pass before some of the technologies below go from deployability to actually being used in mass products. This implies that one cannot from the estimations below construct scenarios for the Information Society in which people in general will be using the described technologies at the times indicated. In other words, the years below refer to "earliest opportunity" for actual deployment in what may be sometimes rather costly systems to be embraced by relatively few customers. Similarly, given the fact that there are thousands of languages in the world, it goes without saying that a technology has been established when it works in at least one of the top languages, a "top language" being defined as a language used by developers in the more affluent parts of the world.

(b) Another point related to (a) above is to do with underlying "production platforms". For many advanced, and still somewhat futuristic, speech and language -related systems, it is one thing to have produced a one-of-a-kind demonstrator system but quite another to have produced the system in a way which enables oneself or others to relatively quickly produce more-of-the-same systems in different application domains. An example is the so-called intelligent multimedia presentation systems which will be discussed in more detail in Section 4. Several examples exist, such as the German WIP system and corresponding systems from the USA. However, as long as we haven't solved the problem of how to produce this kind of system in a relatively quick and standardised way, intelligent multimedia presentation

systems are not going to be produced in numbers but will remain research landmarks. The timeline list below mostly avoids mentioning systems of this kind, assuming for the kinds of systems mentioned that the "production platform" issue has been solved to some reasonable extent at the time indicated.

(c) There is some, inevitable because of the brevity of the timeline entries, vagueness in what the described technologies can actually do.

(d) It is assumed that, after a certain point in time which could be, say, 2006, the distinction between technology use for the web and technology use for other purposes will have vanished.

(e) There is no assumption about *who* (which country, continent, etc.) will produce the described landmark results. However, given the virtually unlimited market opportunities for the technologies listed as a whole, it is expected that a consolidated technology timeline list will command keen interest among decision makers from industry and funding agencies.

(f) There is nothing about (software) agent technologies below. It is simply assumed that what is currently called software agent technologies will be needed to achieve the results described and will be available as needed.

(g) In principle, of course, any technology timeline list is subject to basic uncertainty due to the "if anything is done about it" –factor. If nothing will be done, nothing will happen, of course. However, most of the technologies listed below are being researched already and the rest will no doubt be investigated in due course. The uncertainty only attaches to who will get there first with respect to any given technology, who will produce the product winners, and how much effort will be invested in order to achieve those results before anybody else.

Technology timelines

Basic technologies

Hypotheses lattices, island parsing, spotting in all shapes and sizes for spoken dialogue	2001
Continuous speech recognisers in OSs for workstations in top languages	2002
Continuous speech recognisers in mobile devices (10000 words vocabulary) in top languages	2003
High quality competitive (with concatenated speech) formant speech synthesis in top languages	2003
Task-oriented spoken dialogue interpretation by plausibility in context and situation	2003
Generally usable cross-language text retrieval	2003
Multilingual authoring in limited domains by constructing conceptual representations	2003
Usable ontological lexicons for limited domains	2003
Usable translation systems for written dialogues (multilingual chatting)	2003
Useful speaker verification technology	2004
Seamless integration of spoken human/machine and human/human communication	2004
First on-line prosodic formant speech synthesis in top languages	2004
Simple task-oriented animated character spoken dialogue for the web	2004
Concept-to-speech synthesis	2004
Stylistically correct presentation of database content	2004
Superficial semantic processing based on ontological lexicons	2004

Max. 2000 words vocabulary task-oriented animated character dialogue for the web	2005
Prosodic formant speech synthesis replaces concatenated speech in top languages	2005
Full free linguistic generation (from concepts)	2005
Robust, general meta-communication for spoken dialogue systems	2005
Writer-independent handwriting recognition	2005
Learning at the semantic and dialogue levels in spoken dialogue systems	2006
Useful multiple-speaker meeting transcription systems	2006
Task-oriented fully natural animated characters (speech, lips, facial expression, gesture) output (only)	2007
Context sensitive summarization (responsive to user's specific needs)	2007
Answering questions by making logical inferences from database content	2007
Speech synthesis with several styles and emotions in top languages	2008
Continuous speech understanding in workstations with standard dictionaries (50000 words) in top languages	2008
Controlled languages with syntactic and semantic verification for specific domains	2008
Large coverage grammars with automatic acquisition for syntactic and semantic processing for limited applications	2008
Task-oriented fully natural speech, lips, facial expression, gesture input understanding and output generation	2010

Systems

First personalised spoken dialogue applications (book a personal service over the phone)	2002
Useful speech recognition-based language tutor	2003
Useful portable spoken sentence translation systems	2003
Useful broadcast transcription systems for information extraction	2003
First pro-active spoken dialogue with situation awareness	2003
Current spoken dialogue systems technology for the web (office, home)	2004
Satisfactory spoken car navigation systems	2004
Current spoken dialogue systems technology for the web (in cars)	2005
Useful special-purpose spoken sentence translation systems (portable, web etc.)	2005
High quality translation systems for limited domains with automatic acquisition	2005
Small-vocabulary (>1000 words) spoken conversational systems	2005
Medium-complexity (wrt. semantic items and their allowed combinations) task-oriented spoken dialogue systems	2005
Multiple-purpose personal assistants (spoken dialogue, animated characters)	2006
Task-oriented spoken translation systems for the web	2006
Useful speech summarisation systems in top languages	2006
Useful meeting summarisation systems	2008
Usable medium-vocabulary speech/text translation systems for all non-critical situations	2010
Medium-size vocabulary conversational systems	2010

Tools, platforms, infrastructure

Standard tool for cross-level, cross-modality coding of natural interactivity data	2002
Infrastructure for rapid porting of spoken dialogue systems to new domains	2003
Platform for generating intelligent multimedia presentation systems with spoken interaction	2005
Science-based general portability of spoken dialogue systems across domains and tasks	2006

Other problems which were strongly felt when producing the list above include: (i) the fact that there is plenty of continuity in technology development. "Continuity" may not be the right term because what happens is that what is later perceived as a new technological step forward is constituted by a large number of smaller steps none of which could be mentioned in a coarse-grained timeline exercise such as the one above. General speaker identification, robust speech recognition in hard-to-model noise conditions, "real" speaker-independent recognition (almost) no matter how badly people speak, or pronounce, some language, are all examples of minute-step progress. (ii) Another problem is to do with speech in fancy-termed circumstances, such as 'ambient intelligence' applications. It may be that there is a hard-core step of technological progress which is needed to achieve speech-related ambient intelligence but then again, may be there isn't. Maybe this is all a matter of using the timelined speech technologies above for a wide range of systems and purposes. Similarly, it is tempting to ask, for instance: "When will I have a speech-driven personal assistant?". But everything depends on what the personal assistant is supposed to be able to do. Some personal assistant technologies exist already. Thus, it does not seem possible to timeline the appearance of speech-driven personal assistants even if this might be attractive for the purpose of advertising the potential of speech technologies.

How well is Europe doing?

No attempt has been made, so far, to annotate the technology timelines with indications of how well, or how badly, European research is doing and hence how likely it is that a particular technology will be made deployable in Europe before anywhere else. In most of the timelined cases above, this would seem to depend primarily on the financial resources and research support mechanism which will be available to European research in the coming decade. In some cases, the US is presently ahead of Europe, such as with respect to continuous speech recognisers in workstations or broadcast transcription systems. In other cases, Europe has the lead, such as in building a standard tool for cross-level, cross-modality coding of natural interactivity data, continuous speech recognisers in mobile devices, advanced spoken dialogue systems, and spoken car navigation systems.

Beyond 2010

Beyond 2010 lie the dreams, such as unlimited-vocabulary spoken conversational systems, unlimited-vocabulary spoken translation systems, unlimited on-line generation of integrated natural speech, lips, facial expression and gesture communication, unlimited on-line understanding of natural speech, lips, facial expression and gesture communication by humans, summarisation-to-specification of any kind of communication, multimodal systems solutions on demand, and, of course, full natural interactive communication.

4. Implications of the Timelines

When analysing the implications of the timelines in Section 3, a number of uncertainties come up with respect to how the market for speech products will develop. At present, most speech

products are being marketed by some 5-10 major companies world-wide. These companies are growing fast as are hundreds of small start-up companies many of which use basic technologies from the larger technology providers. It may be assumed that this market structure will not continue in the future. Rather, speech recognition and synthesis technologies would seem likely to become cheap, or even free and open source, components which will come with all manner of software and hardware systems. The implication is that all ITC providers who want to, will provide value-added speech products and that the basic speech technologies will not be dominated by a small number of large suppliers. Some important share of the speech market, including de facto standards in various areas, will probably be picked up by large custom software and mobile phone technology suppliers, such as Microsoft and Nokia, but that is likely to happen in any realistic scenario for the coming decade. The conclusion is that, during the coming decade, speech will be everywhere, in all sorts of products made by all sorts of companies. But will speech be everywhere in bulk? This raises a second uncertainty.

In one scenario, speech will be present in all or most ITC products by 2010, and speech will be popular and will be used as much as input keys, input buttons, and output graphics displays are being used today. In another scenario, however, speech uptake will be slow and arduous. Several reasons could be given for the latter scenario. Thus, (a) it may take quite some time before speech recognition is being perceived by users to be sufficiently robust to make users switch to speech where speech is better ideally. (b) It may take quite some time before the field and the market has sorted out when to use speech as a stand-alone modality and when to use speech in combination with other input/output modalities. If these two (a + b) take-up curves do not grow in any steep manner, speech may still be widespread by 2010, but speech will still not be as important an input/output modality as it is likely to become later on. For the time being, we would appear to have too little information to be able to decide between the two scenarios just discussed. There is simply not enough data available on user uptake of speech technologies to enable a rational decision to be made.

Exploitation today

Already today, there is a great exploitation potential for speech technologies because of the simple facts that (i) the technologies which already exist in a few top languages could be ported to hundreds of other languages, and (ii) the types of applications which already exist can be instantiated into numerous other applications of similar complexity. At this end of the speech technology spectrum, the emphasis is on flexible and versatile production platforms, quality products, and low-cost production rather than on research. This is particularly true of low-complexity over-the-phone spoken language dialogue information systems using continuous speech input. Users would seem to have adopted these systems to a reasonable extent already. The same degree of user acceptance does not appear to characterise the uptake of, e.g., spoken language dictation systems or simple spoken command systems for operating screen menus. Even if purchased by widely different groups of users, the former would appear to be used primarily by professionals, such as lawyers and medical doctors, and the latter hardly seems to be used at all. Also, text-to-speech systems for the disabled and increasingly for all users, do appear to have a significant exploitation potential already.

Key technologies: speech-only

The timelines in Section 3 highlight a series of key speech-only technologies which are still at the research stage, including:

- prosody in on-line speech synthesis;
- multi-speaker broadcast and meeting transcription;

- speech summarisation;
- speech translation; and
- conversational spoken dialogue.

Prosody in on-line speech synthesis

Prosody in on-line speech synthesis is probably important to the speed of take-up of speech technologies because users would appear likely to prefer prosodic speech output to non-prosodic speech output. However, there do not seem to exist firm estimates as to how much prosody matters. Reasonably clear and intelligible non-prosodic text-to-speech already exists for some top languages and might turn out to be satisfactory for most applications in the short-to-medium term.

Multi-speaker broadcast and meeting transcription

Multi-speaker broadcast transcription forms the topic of massive US-initiated research at the moment and appears likely to start becoming widely used in practice relatively soon. Like *meeting transcription* technology, multi-speaker broadcast transcription technology has a large potential for practical application as well as for acting as a driving force in speech and natural language (text) processing research. Once multi-speaker broadcast speech audio and meeting speech audio can be useably transcribed so that first application paradigms for these technologies have been achieved, the transcriptions can be further processed by other technologies, such as speech summarisation and speech translation technologies. It would be very valuable for European speech research if Europe could launch a meeting transcription technology evaluation campaign before the US (evaluation campaigns will be discussed below).

Speech summarisation

Speech summarisation is being experimented with already, often by using text or transcribed speech instead of raw speech data. Speech and text summarisation technology including intelligent speech and text search would seem to hold enormous potential by enabling users to obtain at-a-glance information on the contents of large repositories of information. The same applies to related technologies, such as question-answer systems which enable the user to obtain answers to specific questions from large repositories of information. Progress in these fields is difficult because of the difficulty of the research which remains to be done. However, the difficulties ahead are counter-balanced by expectations that far-less-than-perfect solutions could help to establish first application paradigms which, in their turn, might help accelerate progress.

Speech translation

Despite the embattled 40-year history of language (text) translation systems, speech translation is now being researched across the world because of the realisation that far-less-than-perfect paragraph-by-paragraph translation could yield useful applications in the shorter term. In their turn, those first application paradigms could serve as drivers of further progress. The German Verbmobil project (http://verbmobil.dfki.de/), for instance, demonstrated just how difficult human-human spoken dialogue translation is. Once application paradigms have been achieved, however, speech translation technology would appear set to gain an enormous market. Still, it may take quite some time before there is a massive growth in the market for speech translation products, due to the difficulty of the research which remains to be done.

Conversational spoken dialogue

For some time, the term ‘conversational spoken dialogue’ has been a catch-all for next-step spoken language dialogue systems, such as those explored in the DARPA Communicator

project. However, the DARPA Communicator agenda remains focused on task-oriented dialogue, such as flight ticket reservation. Even if conducted through mixed initiative spoken dialogue in which the human and the machine exchange dialogue initiative in the course of their dialogue about the task, task-oriented spoken dialogue might not qualify as conversational spoken dialogue. Conversational spoken dialogue is mixed-initiative, to be sure, but in conversational spoken dialogue there is no single task and no limited number of distinct tasks which have to be accomplished. Rather, spoken conversation systems may be characterised as *topic-oriented.* It is the breadth and complexity of the topic(s) on which the system is able to conduct conversation which determine its strength. Research on spoken conversation systems is still limited. Obviously, however, spoken conversation systems hold an enormous application potential because they represent the ultimate generalisation of the qualities which everybody seem to appreciate in task-oriented mixed initiative spoken language dialogue systems.

Key technologies: multimodal systems

In addition to speech-only technologies, the timelines in Section 3 highlight a series of multimodal speech systems technologies which are still at the research stage in most cases, including:

- intelligent multimodal information presentation including speech;
- natural interactivity;
- immersive virtual reality and augmented reality.

Intelligent multimodal information presentation including speech

Intelligent multimodal information presentation including speech is a mixed bag of complex technologies which do not seem to have any clear research direction at the present time. The reason is that the term *multimodality,* as pointed out in Section 2 above, refers to a virtually unlimited space of combinations of (unimodal) modalities. Thus, Modality Theory (Bernsen 1997b, 2001) has identified an exhaustive developers' toolbox of unimodal input/output modalities in the media of graphics (or vision), acoustics (or hearing), and haptics (or touch) consisting of more than a hundred unimodal modalities. The number of possible combinations of these unimodal input/output modalities is evidently staggering and, so far, at least, no way has been found to systematically generate a subset of good and useful modality combinations which could be recommended to system developers. The best current approach is to list modality combinations which have been found useful already in experimental or development practice. Obviously, given the limited exploration of the space of possible modality combinations which has taken place so far, those combinations constitute but a tiny fraction of the modality combinations which eventually will be used in HHSI. The same lack of systematicity applies to the subset of useful modality combinations which include speech output and/or speech input. Thus, for instance, it is known that speech and static graphics image output is a useful modality combination for some purposes and that the same holds for combined speech and pen input into various output domains as well as for speech and pointing gesture input into, e.g., a static graphics map output domain. The qualifying term *intelligent* is being used to distinguish intelligent multimodal information presentation systems from traditional multimedia presentations. In traditional multimedia presentations, the user uses keyboard and mouse (or similar devices) to navigate among a fixed set of output options all of which have been incorporated into the system at design-time. In intelligent multimodal information presentation systems, the system itself generates intelligent multimodal output at run-time. This may happen through run-time language and/or speech generation coordinated with run-time graphics image generation and in many other ways as

well. Some years ago, a reference model for intelligent multimodal information presentation systems was proposed by an international consortium of developers (Computer Standards and Interfaces 18, 6-7, 1997). Since then, little systematic development has happened, it appears, which is probably due to the fact that the field is as open-ended at it is. Still, it would appear that (i) the field of intelligent multimodal information presentation systems is an extremely promising approach to complex interactive information presentation, such as in interactive systems for instruction tasks for which several output modalities are needed, including speech. In order to advance research in this field, research is needed on Modality Theory in order to identify potentially useful modality combinations as well as on next-step architectures and platforms for intelligent multimodal information presentation.

Natural interactivity

As argued in Section 2, fully natural interactive systems represent a necessary vision for a large part of the field of interactive systems. Furthermore, spontaneous speech input/output is fundamental to natural interactive systems. Given this (latter) fact, it would seem that speech research is set to take the leading role in the development of increasingly natural interactive systems. Already today, this research and development process can be broken down into a comprehensive, semi-ordered agenda of research steps. The steps include, at least, (i) *fundamental research on human communicative behaviour,* including identification of the relevant phenomena which are being coordinated in human behaviour across abstraction levels and modalities, such as speech prosody and facial expression; validated coding schemes for these phenomena; and standard tools for coding the phenomena in order to create research and training resources in an efficient and re-usable fashion; (ii) *speech and graphics integration* in order to achieve full run-time coordination of spoken output with lip movement, facial expression, gaze, gesture and hand manipulation, and bodily posture; (iii) *speech and machine vision integration* in order to enable the system to carry out run-time understanding of spoken input in combination with lip movement, facial expression, gaze, gesture and hand manipulation, and bodily posture; and (iv) *conversational spoken dialogue* as discussed above. Other relevant technologies include, i.a., machine learning and 3D graphics modelling of human behaviour. Although research in underway on (i) through (iv), there is no doubt that the field might benefit strongly from a focused effort which could connect the disparate research communities involved and set a stepwise agenda for achieving rapid progress. The application prospects are virtually unlimited, as witnessed by the consensus in the field that increased natural interaction tends to generate increased trust in HHSI.

Immersive virtual reality and augmented reality

It is perhaps less clear what are the speech technology application prospects of immersive virtual reality. Today, immersive virtual reality requires that users are wired up with 3D goggles, force feedback data gloves, data suits, and/or wired surfaces and other wired equipment, such as flight cockpits or bicycles. At the present time, it seems uncertain to which extent and for which purposes immersive virtual reality technologies will be found useful in the future. The primary purposes for which these technologies are being used to day are advanced technology exhibition and demonstration, and the building of rather expensive simulation setups, such as flight simulators. Furthermore, it is far from clear which role(s) speech will come to play in immersive virtual environments. These remarks also apply to *augmented reality* technology.

Other research and supporting measures needed

In order to promote efficient research progress on advanced interactive systems which include speech as a modality, technology research is far from sufficient. As pointed out in Section 2,

present and future advanced systems research takes place in an extremely complex context in which leading research efforts must incorporate global state-of-the-art developments in many different fields. World-leading speech-related systems research should be accompanied by the following kinds of research, at least:

- state-of-the-art generic platforms;
- generic architectures;
- hardware;
- specialised best practice in development and evaluation;
- standard re-usable resources;
- behavioural research;
- neural basis for human natural communicative behaviour;
- design of form and contents;
- porting technologies to languages, cultures and the web;
- the disabled;
- maintenance for uptake.

State-of-the-art generic platforms

In order to effectively aim at exploitable results from early on, speech-related systems research needs to build upon existing state-of-the-art generic platforms including APIs. If a state-of-the-art generic platform is not available to the researchers, either because it does not yet exist or because it is inaccessible for proprietary reasons, researchers have to build it themselves. This is not possible in small-scale research projects which have an additional research agenda which presupposes a working platform. The consequence is that the research project will either build upon some sub-optimal platform in order to complete the research agenda, or build a better platform but not complete the research agenda. Both consequences are unacceptable, of course, but the former may work temporarily if the research aims are very advanced ones. However, when the research aims have been achieved or, at least, somehow explored, there will typically be no practical way of continuing the research in order to produce a state-of-the-art generic platform which could bring the research results towards the market. Two implications seem to follow: (i) it would be highly desirable if companies could be encouraged to make their most advanced platforms accessible to researchers. (ii) If a state-of-the-art generic platform is missing altogether, it should either be produced in a separate project or projects should be made so large as to include platform development. Both implications would seem to require a transformation of existing European research funding mechanisms.

Generic architectures

It would seem likely that overall research speed and efficiency in Europe could be accelerated by research on *generic architectures* for future systems, such as conversational spoken dialogue systems, intelligent multimodal information presentation systems which include speech, or natural interactive systems. In the absence of research initiatives on generic architectures for future systems, research projects are likely to specify idiosyncratic architectures which may satisfy their present needs but which do not sufficiently take into account global developments nor prepare for the next steps in advanced systems development. For the time being, there does not appear to be any European speech-related initiative in this field apart from the CLASS project which was launched in the autumn of 2000 (http://www.class-tech.org/). For efficiency, work on generic architectures should be done as

a collaborative effort between many small-scale research projects and industry as in CLASS, or between a medium-scale research project and industry.

Hardware

Increasingly, advanced systems demonstrators require *hardware* design and development. For many research laboratories, this is a new challenge which they are ill-prepared to meet. Moreover, there is no strong tradition for involving hardware producers in the field of speech technologies, primarily because the need for involving them is a rather recent one. Ways must be found to forge links with leading hardware producers in order to make emerging hardware available to researchers. This problem has much in common with the platform issue discussed above.

Specialised best practice in development and evaluation

Advanced speech systems research is conducted in a software engineering space bounded by, on the one hand, general software engineering best development and evaluation practice and, on the other, emerging ISO standards and de facto standards imposed by global industrial competition. Between these boundaries lies software engineering best practice in development and evaluation specialised for various speech-related systems and component technologies. This field remains ill-described in the literature. Apart from the DISC project on best practice in the development and evaluation of spoken language dialogue systems (www.disc2.dk), some work on evaluation in EAGLES Working Groups during the 1990s (http://www.ilc.pi.cnr.it/ EAGLES96/home.html), various national evaluation campaigns, and planned work in CLASS, little work has been done in Europe. By contrast, massive work has been done on component evaluation in the US over the last fifteen years. The result is that the speech-related technology field is replete with trial and error, repetitions of mistakes, and generally sub-state-of-the-art approaches. These negative effects are multiplied by the presence in the field of a large number of developers who are new to the field.

Admittedly, the field of software engineering best practice in development and evaluation specialised for various speech systems and component technologies is difficult and costly to do something about under present conditions. Technology *evaluation* campaigns are costly to do and require serious logistics. Yet the US experience would seem to indicate that technology evaluation campaigns are worth the effort if carried out for key emerging technologies including some of the technologies described in this paper. When a technology has gone to the market, industry does not want to participate any more and rather wants, e.g., evaluation toolkits for internal use. For emerging technologies, however, technology evaluation campaigns are an efficient means of producing focused progress. In fact, all participants tend to become winners in the campaigns irrespective of their comparative scorings according to the metrics employed, because everybody involved learns how to improve, or when to discard, their technologies and approaches. For Europe, technology evaluation campaigns for key emerging technologies could be a means of creating lasting advances on its global competitors. In order to take care of the complex logistics needed for the campaigns, it is worth considering to establish a European agency similar to the US NIST (National Institute for Standards in Technology) whose comprehensive experience with technology evaluation campaigns makes it comparatively easy to plan and launch campaigns in novel emerging technologies. Alternatively, NIST might be asked to undertake to run technology development and evaluation campaigns in Europe, provided that this does not offend political and industrial sensibilities too much.

Effective *development* best practice work specialised for speech technologies is difficult to do under the current European funding mechanisms. The reason is that development best practice work requires access to many different components, systems and approaches in order to

create an effective environment for the discussion and identification of best practice. This environment can only be established across many different small-scale projects or within medium-scale projects. CLASS is the first example of such an environment.

Standard re-usable resources

The term *resources* covers raw data resources, annotated data resources, annotation schemes for data annotation, and annotation tools for efficient automatic, semi-automatic or manual annotation of data. Resources are crucial for many different purposes, such as research into coding schemes or the training of components. Also, resources tend to be costly to produce. This means that, if the relevant resources are not available, research projects often take the easy way out which is to use less relevant but existing and accessible resources for their research. The results are sub-optimal research results and slowed-down progress. Common to resources of any kind is the need for standardisation. If some resource is not up to the required standards, its production is often a waste of effort because the created resource cannot be used for anything useful. In its strategy paper from 1991, ELSNET (http://www.elsnet.org/) proposed the establishment of a European resources agency. This recommendation was adopted through the creation of ELRA (European Language Resources Agency http://www.icp.inpg.fr/ELRA/ home.html) in 1995. ELRA is now a world-recognised counterpart to the US LDC (Linguistic Data Consortium, http://www.ldc.upenn.edu/). Still, ELRA is far from having the capacity to produce on its own all the resources and standards needed for efficient research progress. By contrast with technology evaluation campaigns, Europe has been active in the resources area during the 1990s. Today, there is a strong need to continue activities in producing publicly available resources and standards for advanced natural language processing, natural interactive systems development, evaluation campaigns as described above, etc. Recently, the ISLE (International Standards for Language Engineering) Working Group on Natural Interactivity and Multimodality (http://www.isle.nis.sdu.dk) has launched cross-Atlantic collaboration in the field of resources for natural interactivity and multimodality.

Behavioural research

Humans are still far superior to current systems in all aspects of natural interactive communication. Furthermore, far too little is known about the natural interactive behaviour which future systems need to be able to reproduce as output or understand as input. There is a strong need for basic research into human natural communicative behaviour in order to chart the phenomena which future systems need to reproduce or understand. This research will immediately feed into the production of natural interactive resources for future systems and components development, as described above.

Neural basis for human natural communicative behaviour

Related to, but distinct from, basic research into human natural communicative behaviour is basic research into the neural basis for human natural communicative behaviour. In the heydays of cognitive science in the 1980s, many researchers anticipated steady progress in the collaboration between research on speech and language processing, on the one hand, and research into the neural machinery which produces human speech and language on the other. However, massive difficulties of access to how human natural communicative behaviour is being produced by the brain turned out to prevent rapid progress in linking neuroscience with speech and language processing research. Today, however, due to the availability of technologies such as MR imaging and PET scanning, as well as the increasing sophistication of the research agenda for the speech technology field, the question arises if it might be timely to re-open the cognitive science agenda just described. Potential results include, among others, input to generic architecture development (cf. above), identification of biologically

motivated units of processing, such as speech and lip movement coordination, and identification of biologically motivated modalities for information representation and exchange. Relevant research is already going on in the field of neuroscience but, so far, few links have been established to the fields of speech technologies and natural interactive systems more generally.

Design of form and contents

Yet another consequence of the increasing emphasis on systems as opposed to system components is the growing importance of form and contents design. It is a well-established fact that design and development for the web requires skills in contents design and contents expression which are significantly different from those which have been developed through centuries for text on paper. In order to develop good demonstrator systems for the web or otherwise, there is a need for strongly upgraded skills in the design and expression of multimodal digital contents. For instance, it is far from sufficient to have somehow gleaned that speech might be an appropriate modality for some intelligent multimodal information presentation instruction system and to have available a state-of-the-art development platform for building the system. To actually develop the system, professional expertise in form and contents design is required. At the present time, few groups or projects in the speech field are adequately staffed to meet this challenge.

Porting technologies to languages, cultures and the web

Right now, the gap between the "have" countries whose researchers have access to advanced speech and natural interactivity components and platforms, and the "have-not" countries whose researchers cannot use those technologies for their own purposes because they speak different languages and behave differently in natural interactive communication, seems to be increasing. There is therefore a need to *port advanced technologies to different languages and cultures* both in Europe and across the world. The market will close the gap eventually in its own way, of course. However, in order to rally the full European research potential in the field in a timely fashion, it would appear necessary to actively stimulate the porting of technologies to new languages and cultures. From a research point of view, the best way to make this happen might be to include in medium-to-large-scale projects the best researchers from "have-not" countries even if, by definition, those researchers have to spend significant time catching up on basic technologies and resources before being able to actively contributing to the research agenda.

There is another sense of the 'porting technologies' -phrase in which Europe as a whole risks falling behind global developments. It is that of *porting speech, multimodal and natural interactivity technologies to the web.* The claim here is not that this is not happening already. The claim is that this cannot happen fast enough. In order to increase the speed of porting technology to the web, it would seem necessary to strongly promote advanced components and systems development for the web. It is far from sufficient to wait until some non-speech technology has been marketed for the web, such as electronic commerce applications, and then try to "add speech" to the technology. A much more pro-active stance would appear advisable, including a strongly increased emphasis on form and contents design as argued above.

The disabled

Advanced technologies for the disabled have a tendency to lag behind technology development more generally for the simple reason that the potential markets for technologies for the disabled are less profitable. Correspondingly, advanced technologies development for the disabled tends to be supported by small separate funding programmes rather than being integrated into mainstream programme research. In many cases, however, it would appear that

systems and components technologies could be developed for any particular group of users before being transferred into applications for many other user groups. To the extent that this is the case, there may be less of a reason to confine the development of technologies for the disabled to any particular research sub-programme.

Maintenance for uptake

Finally, the small-scale science paradigm of small and isolated research projects does not at all cater for the fact that, in the complex world of advanced systems research, a wealth of prototype systems, proto-standard resources, web-based specialised best practice guides, etc., are being produced which have nowhere to go at the end of the projects in which they were developed. Their chances of industrial uptake, re-use by industry and research, impact on their intended users, etc., might become very substantially increased if it were possible to maintain them and make them publicly accessible for, say, two years after the end of projects. For this to happen, there is a need for (i) a stable web portal which can host the results, such as the present HLT (Human Language Technologies) portal under development (http://www.HLTCentral.org); (ii) open source clauses in research contracts for technologies which have nowhere to go at the end of a project; and (iii) financial support for maintenance. These requirement are likely to impose considerable strain of current European research support mechanisms. However, with some legal effort and a modest amount of financial support, the many research results produced in the speech-related field in Europe which are not being taken up immediately and which are not within the remit of ELRA, could gain much more impact than is presently the case.

5. Proposed Actions

Early preparations for the European Commission's 6th Framework Programme (FP6) including IST (Information Society Technologies) research are now in progress. It is premature to make predictions with any degree of certainty as to how the IST part of FP6 will shape up. Current information suggests an increased emphasis on basic research compared to the present FP5. In addition, it is possible that FP6 will include opportunities for the medium-scale research initiatives which were called for on several occasions above, i.e. large-scale "clusters" of projects all addressing the same research topic in a coordinated fashion. Finally, the current covering title for FP6 IST research is "ambient intelligence" which is one of the terms of fashion quoted in the present paper. Given the timelines and their analysis above, it does not seem to matter much which covering term is being chosen for FP6. "Ambient intelligence" is as apt as several others for FP6 and future advanced interactive systems research but, as argued in Section 3, it is far from clear if ambient intelligence requires us to focus on any particular segment of future speech-related technologies. However, the possible, increased emphasis on basic research as well as the possibility of carrying out medium-scale science in speech-related technologies are to be welcomed in the light of the argument above.

5.1 Research priorities for speech-related technologies 2000-2010

Taking into our stride the transformations of the field of speech-related research from speech-only to interactive systems in general, and from components research to interactive systems research, the top priorities in speech-related technologies research are:

- multi-speaker meeting transcription development and evaluation campaigns;
- speech summarisation development and evaluation campaigns;
- speech translation prototypes, generic platforms, and generic architectures. Development and evaluation campaigns are highly desirable;

- conversational spoken dialogue prototypes, generic platforms, and generic architectures. Development and evaluation campaigns are highly desirable;
- next-step prototypes, generic platforms, and generic architectures for intelligent multimodal information presentation;
- next-step prototypes, generic platforms, and generic architectures for natural interactive systems.

As soon as theoretically and practically feasible, all of the above advanced speech, multimodal and natural interactivity technologies should be developed for the web including hardware, form and contents design. The fact that some top research priorities have been mentioned above emphatically does not preclude the desirability of continuing "business as usual" in the field of speech-related research, including continued research into *all* of the technologies which have been mentioned earlier in the present paper. On the contrary, business as usual is actually assumed by the above top priorities list which focuses on technologies over and above business as usual. This also applies to next-step research into already deployed speech-related technologies, such as mixed initiative, task-oriented spoken dialogue systems.

For basic research leading to novel concepts, theories and formalisations, the top priorities are:

- basic research into human natural communicative behaviour;
- a novel theory of natural communication which can replace speech acts theory and discourse theory by taking the notion of a complete communicative act as its basic notion;
- research on Modality Theory in order to identify potentially useful modality combinations;
- establishment of collaborative links to research into the neural basis for human natural communicative behaviour.

5.2 Research organisation needed

Medium-scale science is needed for, at least, the coordinated development of natural interactive systems prototypes, generic platforms, generic architectures, best practice in development and evaluation, and standard resources. A large, medium-scale science project with these objectives should include the porting of technologies to new languages and cultures.

It is quite possible that the medium-scale science model could be applied to research into other speech-related technologies, such as speech translation technologies, conversational spoken dialogue systems, or speech technologies for ambient intelligence.

For researchers in small-scale speech-related projects, in particular, the creation of a generic platforms and hardware "bourse" through contributions from European industry would be of great importance.

Finally, we should stop having research programme ghettos for technologies for the disabled.

5.3 Infrastructural actions needed

In order to promote maximum uptake of the research results produced, it would be highly desirable to have funding for low-cost ways of maintaining research results for later uptake.

Given the emphasis on technology development and evaluation campaigns above, Europe needs to establish an evaluation and standards agency. It is not evident to the present author

that current political and industrial sensibilities would allow the US NIST to undertake to run technology development and evaluation campaigns in Europe.

This having been said, there is much to be said for increasing global collaboration on many aspects of speech-related research, such as creating a coordinated global infrastructure for resources distribution.

References

Benoit, C., Martin, J. C., Pelachaud, C., Schomaker, L., and Suhm, B.: Audio-Visual and Multimodal Speech-Based Systems. In D. Gibbon, I. Mertens and R. Moore (Eds.): *Handbook of Multimodal and Spoken Dialogue Systems*. Dordrecht: Kluwer Academic Publishers 2000, 102-203.

Bernsen, N. O.(1997a): Towards a tool for predicting speech functionality. *Speech Communication* 23, 1997, 181-210.

Bernsen, N. O. (1997b): Defining a Taxonomy of Output Modalities from an HCI Perspective. *Computer Standards and Interfaces,* Special Double Issue, 18, 6-7, 1997, 537-553.

Bernsen, N. O.: Multimodality in language and speech systems - from theory to design support tool. In Granström, B. (Ed.): *Multimodality in Language and Speech Systems.* Dordrecht: Kluwer Academic Publishers 2001 (to appear).

CLASS: http://www.class-tech.org/

Computer Standards and Interfaces, Special Double Issue, 18, 6-7, 1997.

DARPA Communicator: http://fofoca.mitre.org/

DISC www.disc2.dk

EAGLES: http://www.ilc.pi.cnr.it/EAGLES96/home.html

ELRA: http://www.icp.inpg.fr/ELRA/home.html

ELSNET http://www.elsnet.org/

i3: http://www.i3net.org/

ISLE: http://www.ilc.pi.cnr.it/EAGLES96/isle/ISLE_Home_Page.htm

ISLE Working Group on Natural Interactivity and Multimodality: http://www.isle.nis.sdu.dk

HLT portal: http://www.HLTCentral.org

LDC http://www.ldc.upenn.edu/

SmartKom: http://smartkom.dfki.de/start.html

Verbmobil: http://verbmobil.dfki.de/

Towards a Road Map on Human Language Technology: Natural Language Processing

Editors: Andreas Eisele, Dorothea Ziegler-Eisele

Version 2 (March 2002)

Abstract

This document summarizes contributions and discussions from two workshops that took place in November 2000 and July 2001. It presents some visions of NLP-related applications that may become reality within ten years from now. It investigates the technological requirements that must be met in order to make these visions realistic and sketches milestones that may help to measure our progress towards these goals.

1. Introduction

Scope of this Document

One of the items on ELSNET's agenda for the period 2000-2002 is to develop views on and visions of the longer-term future of the field of language and speech technologies and neighboring areas, also called ELSNET's Road Map for Human Language Technologies. As a first step in this process, ELSNET's Research Task group is organizing a series of brainstorming workshop with a number of prominent researchers and developers from our community. The first one of these workshops took place in November 2000 under the general motto “How will language and speech technology be used in the information world of 2010? Research challenges and infrastructure needs for the next ten years". The second one was co-organized in July 2001 by ELSNET and MITRE as part of ACL-2001 and had the somewhat more specific orientation on “Human Language Technology and Knowledge Management (HLT-KM)”. This workshop brought together more than 40 researchers from industry and academia and covered a considerable range of topics related to KM and HLT in general.

This paper aims at summarizing and organizing material from both workshops, but concentrates on applications and technologies that involve NLP, i.e. the processing of written natural language, as speech-related technologies and new models of interactivity have already been covered in documents presented around the first workshop. In the discussion of question answering and summarization, vision papers and roadmaps compiled by researchers in the US and published by NIST have been taken as an additional source of inspiration.

The Growing Need for Human Language Technology

Natural language is the prime vehicle in which information is encoded, by which it is accessed and through which it is disseminated. With the explosion in the quantity of on-line

text and multimedia information in recent years there is a pressing demand for technologies that facilitate the access to and exploitation of the knowledge contained in these documents.

Advances in human language technology will offer nearly universal access to on-line information and services for more and more people, with or without skills to use computers. These technologies will play a key role in the age of information and are cited as key capabilities for competitive advantage in global enterprises.

Extraction of knowledge from multiple sources and languages (books, periodicals, newscasts, satellite images, etc.) and the fusion into a single, coherent textual representation requires not only an understanding of the informational content of each of these documents, the removal of redundancies and resolution of contradictions. Also, models of the user are required, the prior knowledge that can be assumed, the level of abstraction and the style that is appropriate to produce output that is suitable for a given purpose.

More advanced knowledge management (KM) applications will be able to draw inferences and to present the conclusions to the user in condensed form, but let the user ask for explanations of the internal reasoning. In order to find solutions for problems beyond a static pool of knowledge, we need systems that are able to identify experts, who have solved similar problems. Again, advanced NLP capabilities will be required to appraise the aptitude of candidates from documents authored by them or describing prior performance.

But also outside of KM, sophisticated applications of NLP will emerge over the next years and decades and find their way into our daily lives. The range of possibilities is almost unlimited. An important group of applications is related to electronic commerce, i.e. new methods to establish and maintain contact between companies and their customers. Via mobile phones, e-mail, animated web-based interfaces, or innovative multi-channel interfaces, people will want to make use of all kinds of services related to buying and selling goods, home-banking, booking of journeys, and the like. Also in the area of electronic learning a considerable growth is expected within the coming years.

Multilinguality

Whereas English is still the predominant language on the WWW, the fraction of non-English Web pages and sites is steadily increasing. Contrasting earlier apprehensions, the future will probably present ample opportunities for giving value to different languages and cultures. However, the possibility to collect information from disparate, multilingual sources also provides considerable challenges for the human user of these sources and for any kind of NLP technology that will be employed.

One of the major challenges is lexical complexity. There will be about 200 different languages on the web and thus about 40.000 potential language pairs for translation. Clearly, it will not be possible to build bilingual dictionaries that are comprehensive both in the number of language pairs and in the coverage of application domains. Instead, multilingual vocabularies need to provide mappings into language independent knowledge organization structures, i.e. common systems of concepts linked by semantic relations. However, the definition of such an “interlingua” will be difficult in cases in which languages make distinctions of different granularity.

Research Trends and Challenges

The field of human language technology covers a broad range of activities with the goal of enabling people to communicate with machines using natural communication skills.

Although NLP can help to facilitate knowledge management, it requires a large amount of specialized knowledge by itself. This knowledge may be encoded in complex systems of linguistic rules and descriptions, such as grammars and lexicons, which are written in dedicated grammar formalisms and typically require many person-years of development effort. The rules and entries in such descriptions interact in complex ways, and adaptation of such a sophisticated system to a new text style or application domain is a task that requires a considerable amount of specialized manpower.

One way to cope with the difficulties in the acquisition of linguistic knowledge was to restrict attention to shallower tasks, such as looking for syntactic "chunks" instead of a full syntactic analysis. Whereas this has proven rather successful for some applications, it obviously severely limits the depth to which the meaning of a document or utterance is taken into account.

Another approach was to shift attention towards models of linguistic performance (what occurs in practice, instead of what is principally possible) and to use statistical or machine learning methods to acquire the necessary parameters from corpora of annotated examples. These data-driven approaches offer the possibility to express and exploit gradual distinctions, which is quite important in practice. They are not only easier to scale and adapt to new domains, their algorithms are also inherently robust, i.e. they can deal, to a certain extent, gracefully with errors in the input.

Statistical parsers, trained on suitable tree banks, now achieve more than 90% precision and recall in the recognition of syntactic constituents in unseen sentences from English financial newspaper text.

However, a lot of work remains to be done, and it is not obvious how the success of corpus-driven approaches can be enlarged along many dimensions simultaneously. One challenge is that analysis methods need to work for many languages, application domains and text types, whereas the manual annotation of large corpora of all relevant types will not be economically feasible. Another challenge is that, other than syntax, many additional levels of analysis will be required, such as the identification of word sense, the reference of expressions, structure of argumentation and of documents, and the pragmatic role of utterances. Often, the theoretical foundation that is required before the annotation of corpora can begin is still lacking.

One could say that for corpus-driven approaches the issue of scalability of the required resources shows up again, albeit in a somewhat different disguise. Hence, research in NLP will have to address this issue seriously, and find answers to the question how better tools and learning methods can reduce the effort of manual annotation, how annotated corpora of a slightly different type could best be re-used, how data-driven acquisition processes can exploit and extend existing lexicons and grammars, and finally how analysis levels for which the theoretical basis is still under development could be advanced in a data-driven way.

Structure of this Document

The remainder of this document is structured as follows. In Chapter 2 we describe a number of prototypical applications and scenarios in which NLP will play a crucial role. Whereas each of these scenarios is discussed mainly from a user's perspective, we also give indications, which technological requirements must be met to make various levels of sophistication of these applications possible. In Chapter 3, the technologies that have been mentioned earlier are discussed in more detail, and we try to indicate which levels of functionality may be expected within the timeframe of this study. These building blocks are

then put into a tentative chronological order, which is displayed in Chapter 4. Finally, Chapter 5 gives some general recommendations about beneficial measures concerning the infrastructure for the relevant research.

2. Applications of NLP

Recent developments in natural language processing have made it clear that formerly independent technologies can be harnessed together to an increasing degree in order to form sophisticated and powerful information delivery vehicles. Information retrieval engines, text summarizers, question answering and other dialog systems, and language translators provide complementary functionalities which can be combined to serve a variety of users, ranging from the casual user asking questions of the web to a sophisticated, professional knowledge worker.

Though one cannot strictly separate the following applications from each other, because one can act as a part of another, we try to dissect the large field of existing and future applications in the hope of making the field as a whole more transparent.

Information Retrieval (IR)

What is called information retrieval today is actually but a foretaste of what it should be. Current systems neither understand the information need of the user, nor the content of the documents in their repositories. Instead of meaningful replies, they just return a ranked, and often very long list of documents that are somehow related to the given query, which is typically very short. A better name for this restricted functionality would be text retrieval.

Information retrieval systems must understand a query, retrieve relevant information, and present the results. Retrieved information may consist of a long document, multiple documents of the same topic, etc and good systems should present the most important material in a clear and coherent manner.

Current information retrieval techniques either rely on an encoding process using a certain perspective or classification scheme to describe a given item, or perform a superficial full-text analysis, searching for user-specific words. Neither case guarantees content matching.

The ability to leverage advances in input processing (especially natural language query processing) together with advances in content-based access to multimedia artifacts (e.g., text, audio, imagery, video) promises to enhance the richness and breadth of accessible material while at the same time improving retrieval precision and recall and thus reducing the search time. Dealing with noisy, large scale, and multimedia data from sources as diverse as radio, television, documents, web pages, and human conversations (e.g., chat sessions and speech transcriptions) will offer challenges.

One important part of IR would be multi-document summarization that can turn a large set of input documents into several different short summaries, which can then be sorted by topics or otherwise put into a coherent order.

Summarization

Summarization will enable knowledge workers access to larger amounts of material with less required reading time. The goal of automatic text summarization is to take a partially structured source text, extract information content from it and present the most important content in a condensed form in a manner sensitive to the needs of the user and task. Scalability to large collections and the generation of user-tailored or purpose-tailored summaries are active areas of research.

The summarization can either be an extract consisting entirely of material copied from the input, or an abstract containing material not present in the input, such as subject categories, paraphrases of content, etc.

For extraction shallower approaches are possible, as frequently the sentences may be extracted out of context. The transformation here involves selecting salient units and synthesizing them with the necessary smoothing (adjusting references, rearranging the text…). Training by using large corpora is possible.

Abstracts need a deeper level of analysis, the synthesis involves natural language generation and some coding for a domain is required.

Depending on their function, three types of abstracts can be distinguished: An indicative abstract provides a reference function for selecting documents for more in-depth reading. An informative abstract covers all the salient information in the source at some level of detail and evaluative abstracts express the abstractor's views on the quality of the work of the author.

Characteristics for the summarization are the reduction of the information content (compression rate), the fidelity to the source, the relevance to the user's interest, and the well-formedness regarding both to syntactic and discourse level. Extracts need to avoid gaps, dangling anaphora, ravaged tables and lists, abstracts need to produce grammatical, plausible output.

Some current applications of summarization are:

1. Multimedia news summaries: watch the news and tell what happened while I was away
2. Physicians' aids: summarize and compare the recommended treatments for this patient
3. Meeting summarization: find out what happened at that teleconference I missed
4. Search engine hits: summarize the information in hit lists retrieved by search engines
5. Intelligence gathering: create a 500-word biography of Osama bin Laden
6. Hand-held devices: create a screen-sized summary of a book
7. Aids for the Handicapped: compact the text and read it out for a blind person

Though there are already promising approaches towards mastering all types of summaries, there are still obstacles to overcome such as the need for robust methods for the recognition of semantic relations, speech acts, and rhetorical structure.

Question Answering (QA)

The straightest way to get access to the gigantic volume of knowledge around us is probably asking questions by communicating with other persons, computers or machines.

An important new class of systems will move us from our current form of search on the web (type in keywords to retrieve documents) to a more direct form of asking questions in natural language, which are then directly responded to with an extracted or generated answer. Currently it is rather straightforward to get an answer to "what questions" (what is the capital of China, what are the opening hours of the hermitage etc.), whereas "why questions" (why did the new market fail) are normally not answered by an information retrieval query, unless the answer happens to be present in the information database, or can be inferred afterwards by the user from the answers she gets.

In the next decade time has come to find answers to why questions from information systems by letting the systems make the appropriate inferences. This requires very sophisticated automatic reasoning methods, based on systematic extraction of information from texts, storing the information in a systematized way, which lends itself to reasoning and inference rules that will be able to draw the proper conclusions from the knowledge stored in the information database.

We can subdivide the long-term goal of building powerful, multipurpose information management systems for QA in simpler subtasks that can be attacked in parallel at varying levels of sophistication, over shorter time frames.

Clearly there is not a single, archetypical user of a Q&A system. In fact there is a full spectrum of questions, starting with simple factual questions, which could be answered in a single short phrase found in a single document (e.g. "Where is the Taj Mahal?"). Next, questions like "What do we know about Company xyz?", where the answer cannot be found in a single document but will require retrieving multiple documents, locating portions of answers in them and combining them into a single response. This kind of question might be addressed by decomposing it into a series of single focus questions.

Finally there are very complex questions, with broad scope, using judgment terms and needing deep knowledge of the user's context to be answered. Imagine someone is watching a television newscast, becomes interested in a person, who appears to be acting as an advisor to the country's Prime Minister. And now the person wants to know things like: "Who is this individual. What is his background? What do we know about the political relationship of this person and the Prime Minister and/or the ruling party?". The future systems that can deal with this type of questions must manage the search in multiple sources in multiple media/languages, the fusion of information, resolution of conflicting data, multiple alternatives, adding interpretation, drawing conclusions.

In order to realize this goal, research must deal with question analysis, response discovery and generation from heterogeneous sources, which may include structured and unstructured language data of all media types, multiple languages, multiple styles, formats and also image data i.e. document images, photography and video.

To the extent to which NLP research will learn to master the challenges of source selection, source segmentation, extraction, and semantic integration across heterogeneous sources of unstructured and semi-structured data, NLP technology will help us to reduce the time,

memory, and attention required to sift through many returned web pages from a traditional search by providing direct answers to questions.

Semantic Web

The standardization committee for the WWW (called W3C) expects around a billion web users by 2002 and an even higher number of available documents. However, this success and exponential grow makes it increasingly difficult to find, to access, to present, and to maintain the information of use to a wide variety of users.

The semantic web will bring structure to the meaningful content of Web pages, creating an environment where software agents roaming from page to page can readily carry out sophisticated tasks for users.

The semantic web is not a separate web but an extension of the current one, in which information is given well-defined meaning better enabling computers and people to work in cooperation. With the help of ontologies large amounts of text can be semantically annotated and classified.

Currently pages on the web use representations rooted in format languages such as HTML or SGML. The information content, however, is mainly presented by natural language. Thus, there is a wide gap between the information available for tools that try to address the problems above and the information kept in human readable form.

The semantic web will provide intelligent access to heterogeneous and distributed information enabling software agents to mediate between the user needs and the available information sources.

The first steps in weaving the semantic web into the structure of the existing web are already under way. In the near future, these developments will usher in significant new functionality as machines become much better able to process and "understand" the data that they merely display at present.

What is required: creation of a machine understandable semantics for some or all of the information presented in the WWW i.e.

- Developing languages for expressing machine understandable meta-information for documents, in the line of RDF, DAML, and similar proposals.
- Developing terminologies (i.e., name spaces or ontologies) using these languages and making them available on the web.
- Integrating and translating different terminologies
- Developing tools that use such languages and terminologies to provide support in finding, accessing, presenting and maintaining information sources.

Developing such languages, ontologies and tools is a wide-ranging problem that touches on the research areas of a broad variety of research communities.

Creation of the relevant tools will require a better knowledge of what the users want to know from websites, i.e. these developments need to be based on a user-centered process view.

Another crucial issue will be: “Who is going to populate the semantic web?” The semantic markup that is required by automated software agents needs to be very easy to create and supporting tools need to be provided, otherwise this wonderful idea will not have significant impact for a long time. Advanced NLP technology that can “guess” the correct semantic annotation and propose suitable markup semi-automatically will enable conformance to the needs of software agents with minimal manual effort.

Dialogue Systems

No matter if people want to buy something, find or use a service or just need information, dialog systems promise user-friendly and effective ways to achieve these goals, even for first time users.

Despite the apparent resemblance to QA systems, there are several specific problems to be solved concerning dialogue modality and structure. Input to a dialog system might be via keypad, voice, pointing device, combinations thereof, or other channels, so all errors and incompleteness of spontaneous natural language will show up. In contrast to QA systems, there will be mixed initiatives of speaker and system and the scope is much wider if we take into account that the focus during natural dialogue may often change. Also, the utterance made during a dialog can only be correctly interpreted based on the dialog context and the mutual knowledge that has been accumulated before it was made.

In future we require systems that can support natural, mixed initiative human computer interaction that deals robustly with context shift, interruptions, feedback and shift of locus or control.

Open research challenges include the ability to tailor flow and control of interactions and facilitate interactions including error detection and correction tailored to individual physical, perceptual and cognitive differences.

Motivational and engaging life-like agents offer promising opportunities for innovation.

Agent/user modeling: Computers can construct models of user beliefs, goals and plans as well as models of users’ individual and collective skills by processing materials such as documents or user interactions/conversations. While raising important privacy issues, modeling users or groups of users unobtrusively from public materials or conversations can enable a range of important knowledge management capabilities

tracking of user characteristic skills and goals enhances interaction as well as discovery of experts by other users or agents

A central problem for the development of dialogue systems is the fact that contemporary linguistics is still struggling to achieve a genuine integration of semantics and pragmatics. A satisfactory analysis of dialogue requires in general both semantic representation i.e. representation of the content of what the different participants are saying and pragmatic information, i.e. what kinds of speech acts they are performing (are they asking a question, making a proposal…)

Analysis of a dialog needs to explain the purpose behind the utterances it consists of. Determining the semantic representation of an utterance and its pragmatic features must in general proceed in tandem. A dialogue system identifying the relevant semantic and pragmatic information will thus have to be based on a theory in which semantics and

pragmatics are both developed with the formal precision that is a prerequisite for implementation and suitably attuned to each other and intertwined.

Applications in Electronic Commerce

New technological possibilities can quickly impact the interaction between companies and their customers. One example are dialog systems that allow customers to obtain personal advises or services. For reasons indicated above, these systems are difficult to build, but once this investment has been done, they can be operated at low cost for the company.

Another example, which may be even sooner to come, is the creation of systems that support processing of emails sent by customers. According to business analyses, e-mail has already now become one of the most common forms of customer communication. For numerous businesses that are not well-prepared, this has transformed e-mail into a severe pain point, giving rise to the pressing need to adopt e-mail response management systems.

Obviously, NLP technologies that are able to extract the salient facts from email messages can constitute a central part of these systems. Due to the potential complexity of the queries and additional problems like ungrammatical input and spelling errors, the correct interpretation of arbitrary messages is far from easy. However, there are several factors that alleviate the situation: Messages that are too difficult for automatic processing can be routed to human agents. In cases in which doubts about the correctness of generated responses persist, these responses can always be checked by manual inspection. Historical data about email exchange with customers can be used to bootstrap the models that are required for the system. Depending on the business, a significant fraction of the emails may be amenable to NLP, including requests for information material, business reports, certificates, statements of account, scheduling requests, conference registrations etc.

e-Learning

Using modern technology to facilitate learning is one of the most promising application domains of NLP. Good QA systems that are able to give answers to the point, or summarization systems that can adapt to the user's prior knowledge and present important additions in a way that is easy to understand could immediately take the place of a good teacher, which an unlimited supply of time and patience. One technology is ripe to build these tools, using them for e-learning will one of the biggest opportunities to our knowledge society.

However, as the European society evolves more and more into multilingualism, it is natural to ask how NLP can help to make language learning easier and more effective. We can imagine systems to help train children to write and to speak a foreign language. There will be combinations of multi-modal aids for the handicapped. A child will write a sentence and the system will correct it and tutor him about the problems. A child will read a text aloud and the system will monitor which words are not right and why and will analyze where the pronunciation problems are. Later the system would suggest some pronunciation exercises in the particular problem.

Systems that are able to guess the intention of a speaker from the speaker's utterances in a flexible and intelligent way will offer a plethora of possibilities for e-learning. As similar capabilities are required for dialog systems in general, there will be significant synergy effects between these fields of research.

Translation

The idea of machine translation (MT) has been one of the driving forces in the early days of NLP. However, even after more than 50 years of effort, current systems still produce output of limited quality, which is suitable for assimilation of foreign-language documents, but not for the production of publishable material. But even if the old dreams did not come true, MT will play an increasing role in the multilingual world.

Last year, for the first time, English constituted less than half the material on the web. Some predict that Chinese will be the primary language of the web by 2007. Given that information on the web will increasingly appear in foreign languages and not all users will be fluent in those languages, there will be a need to gist or skim content for relevance assessment and/or provide high quality translation for deeper understanding. Some forms of translation for information access is already today available in the web at no cost. The increasing demand for these services will give a push to improve their quality and the providers will find ways to increase vocabularies and translation quality semi-automatically from terminological resources, bilingual corpora and similar sources. Also the need for interactive systems that can give rough translations of chat sessions in real time will create interesting challenges.

Clearly, any systematic collection of lexical and terminological information in the form of domain-specific ontologies will help to build better MT systems for these domains. Conversely, the construction of ontologies can be facilitated by automatic alignment of existing translations, as this will naturally lead to a clustering of the vocabulary along the relevant semantic distinctions.

These developments will also have an impact on improved systems for high-quality translation for the dissemination of documents. Chances are that hybrid combinations of symbolic and stochastic translation engines, able to learn relevant terminology from translation memories will eventually achieve a level of performance that will make them useful for the professional translator. Combined with multi-modal workbenches where voice input, keyboard and mouse interaction will make the composition of the target text as convenient as possible, these new technologies may help at least in some easier domains, where so far the effort of the human translator is dominated by low-level activities such as entering the text, adjusting the formatting, copying names and numbers, which are clearly amenable to partial automation.

3. Technologies for NLP

This chapter contains a more detailed discussion of some of the technologies that are required for the applications mentioned in the last chapter. Most of the material is organized along traditional fields of research in NLP, describing technologies that already exist, but must be further developed to achieve the ambitious goals. Some technologies cannot be assigned to one specific level, because they serve a more generic purpose, such as the extraction of relevant knowledge from text corpora.

Low-Level Processing

Most systems that analyse natural language text typically start by segmenting the text into meaningful tokens. Sometimes, the exact spelling of these tokens needs to be brought into a

canonical form, so that it can match with a lexical entry. Both processes can be based on matching the input against regular expressions, for which efficient algorithms exist. Whereas this task looks straightforward from the distance, there are actually some subtle details that need to be considered. Quite often, a decision whether a word should be split at a special character or whether a dot ends a sentence or is part of the preceding word depends on the vocabulary of the domain and on layout conventions used in this document, so that general rules cannot be defined. Documents that need to be analyzed may contain markup from text processors, which needs to be stripped or interpreted in a suitable way. The knowledge required in these preliminary stages of processing can already be quite specific, so that a manual creation of suitable rule systems is not economically feasible.

Current research on the automatic tokenization and normalization of texts therefore concentrates on the question how the knowledge required by these methods can automatically be derived from examples, using techniques statistical or machine learning approaches.

Another difficulty is the treatment of noise in the input. Output of speech recognition systems often contains recognition errors at rather high rates. Utterances entered interactively or printed documents that have undergone OCR have similar problems. Unfortunately, the distortion of even a single character can mess up the linguistic analysis of the complete input. But of course, we expect NLP systems to deal gracefully and intelligently with small distortions and errors in the input.

To make systems more robust against noisy input, probabilistic techniques for the restoration of distorted signals, which have shown to be quite effective in speech recognition, need to be adapted and generalized to new applications. However, training simple-minded statistical models on massive amounts of data will often not be feasible. By now, statistical language models that incorporate grammatical knowledge are able to give slight improvements over n-gram approaches, and it seems plausible to expect that future improvements of these will be easier to use in specific situation where training data is scarce. Large vocabularies, many types of distortions, and the need to use fine-grained contextual knowledge for improved predictive models constitute significant research challenges. Most likely, there will be some synergy between language models used in speech and similar models that will be developed for low-level processing and correction of written ill-formed input.

Once the segmentation into basic units has been performed, the next step is to identify suitable lexical entries for each token and, in cases where more than one entry applies, to determine which one is most appropriate in the given context. This process is called part-of-speech disambiguation or POS tagging and is usually done with statistical models or machine-learning approaches trained on manually tagged data. Current technology achieves rather high accuracy on newspaper text, but again, performance suffers significantly when a model trained on a certain set of data is applied to text from a different domain. As the output of the POS tagger is typically used as input to subsequent modules, tagging errors may hamper the correct analysis of much more than the affected word. Research on high-quality POS tagging will face problems that are similar to those of language modelling: It requires detailed information about a large number of rare words that may be quite specific to the given domain and application, which is difficult to construct, no matter which road to lexical acquisition is taken. Any effort that will support the construction, distribution, sharing and re-use of large, domain-specific lexical resources will doubtlessly also help to improve the accuracy of POS tagging on text from these domains.

The next step in the analysis of text is to identify groups of words that belong together and refer to one semantic entity. Often, these phrases contain names, and for many practical applications, it is important to classify these expressions according to the type of entity they denote (Person, City, Company, etc.). Depending on the application, the classification may be more or less fine-grained. Again, it is obvious that improved lexical knowledge will help to improve the performance of named entity recognition. But we cannot in all cases rely on a lexical resource to cover the relevant entities. A text may discuss the opening of a new company, which will therefore not be contained in the lexicon. To handle such cases intelligently, we need mechanisms that can exploit contextual clues for the correct classification of unknown entities and we need effective mechanisms that propagate information about new entities into the lexical repositories, so that the system as a whole learns from the texts it sees, similar to the way a human reader would do.

Syntactic Analysis

The goal of syntactic analysis is to break down given textual units, typically sentences, into smaller constituents, to assign categorical labels to them, and to identify the grammatical relations that hold between the various parts.

In most applications of language technology the encoded linguistic knowledge, i.e. the grammar, is separated from the processing components. The grammar consists of a lexicon, and rules that syntactically and semantically combine words and phrases into larger phrases and sentences.

Several language technology products on the market today employ annotated phrase-structure grammars, grammars with several hundreds or thousands of rules describing different phrase types. Each of these rules is annotated by features and sometimes also by expressions in a programming language.

The resulting systems might be sufficiently efficient for some applications but they lack the speed of processing needed for interactive systems, such as applications involving spoken input, or systems that have to process large volumes of texts, as in machine translation.

In current research, a certain polarization has taken place. Very simple grammar models are employed, e.g. different kinds of finite-state grammars that support highly efficient processing. Some approaches do away with grammars altogether and use statistical methods to find basic linguistic patterns. Other than speed, these shallow and statistically trained approaches have advantages in terms of robustness, and they also implicitly perform disambiguation, i.e. when more than one analysis is possible, they make a decision for one reading (which of course may be the wrong one).

On the other end of the scale, we find a variety of powerful linguistically sophisticated representation formalisms that facilitate grammar engineering. These systems are typically set up in a way that all logically possible readings are computed, which increases the clarity (no magic heuristics hidden in procedures), but also slows down the processing. Despite their nice theoretical properties it has so far been difficult to adapt these systems to the needs of real-world applications, where speed, robustness, and partial correctness in typical cases are more urgent than theoretical faithfulness and depth of analysis.

How will this situation evolve? The two approaches will continue to compete for potential applications, and the current advantage for shallow approaches will diminish as more ambitious applications get within reach, and as languages are used that require richer analysis.

This will give incentives for shallow approaches to struggle for higher accuracy and more detailed analyses, whereas the deep processing will be forced to find workable solutions for the problems with speed and robustness. In the ideal case, more fine-grained forms of integration will be found, i.e. hybrid systems that will keep the advantages of both worlds as far as possible.

The simplest integration will just use shallow analysis as a fallback mechanism when deep analysis fails. In this case, results from both approaches need to be translated into one common representation, and the development of such a "common denominator" will be a significant challenge. To achieve an even more fine-grained cooperation between both approaches, deep analysis may be equipped with the ability to locally fall back to more superficial processing, driven by the need to deal with a specific problem in the input. Vice versa, the results of shallow analysis might be combined into a more detailed structure incrementally, based on rules from a deep grammar. Also analyses of corpus data obtained with shallow tools can be mined for linguistic knowledge that is then fed into resources used by a deep parser, and vice versa.

Research challenges will be how to find syntactic parsers that are at the same time fast, robust, deliver a detailed analysis that is correct with high probability and that are easily to adapt to special domains.

Semantic Analysis

The goal of semantic analysis is to assign meanings to utterances, which is an essential precondition for most applications of NLP. However, what level of abstraction is required in this phase depends on the difficulty of the task. Extraction of answers to simple factual questions from a given text will require less depth in analysis than the summarization of a lengthy treatise in few paragraphs.

We can dissect the task of semantic analysis into several subtasks, depending on the linguistic level where it takes place. Most important are the semantic tagging of ambiguous words and phrases, and the resolution of referring expressions.

The disambiguation of word senses needs to identify the meaning that should be assigned to a given word. The hardest part of this task is to define the set of meanings that should be considered in this task, i.e. to select the appropriate granularity for the conceptualization. The emergence of standardized, large-scale ontological resources will help to solve this part of the task, as the concepts that appear in such ontologies are a natural choice for the meanings of single words or simple phrases. Additionally, multilingual corpora that are aligned on the level of words and phrases can serve as an approximation to sense-tagged corpora, so draft ontologies and models for sense disambiguation can be extracted from these.

Considerable efforts in defining useful evaluation metrics for sense disambiguation are pursued in the ongoing SENSEVAL activities. So far, the methods used by the participants of SENSEVAL are mostly based on simple statistical classification using features extracted from the context of word occurrences. To the extent to which robust, high quality systems for syntactic analysis will appear, this will also help to obtain improved accuracy in the semantic disambiguation.

The resolution of referring expression such as pronouns or definite noun phrases is the ability to identify their target, which may be expressions that appear prior in the text, abstractions of material that appeared earlier, or entities that exist independently from the text in existing

background knowledge. Seen in a more general way, the task is to cull out objects and events from multimedia sources (text, audio, video). An example challenge includes extracting entities within media and correlating those across media. For example this might include extracting names or locations from written/spoken sources and correlating those with associated images. Whereas commercial products exist to extract named entities from text with precision and recall in the ninetieth percentile, domain independent event extractors work at best in the fiftieth percentile and performance degrades further with noisy, corrupted, or idiosyncratic data.

Therefore work on the resolution of referring expression and the identification of entities in text and multimedia documents remains important fields of activity for the future.

Discourse and Dialogue

Extracting the knowledge contained in documents and understanding and generating natural dialog behavior requires more than the resolution of local semantic ambiguities. Intelligent analysis needs to consider the global argumentative structure of documents and discourse, and dialogs need to be analyzed for pragmatic content.

Computational work in discourse has focused on two different types of discourse: extended texts and dialogues, both spoken and written, yet there is a clear overlap between these two: dialogues contain text-like sequences spoken by a single individual and texts may contain dialogues. But application opportunities and needs are different. Work on text is of direct relevance to document analysis and retrieval applications, whereas work on dialogue is of import for human-computer interfaces regardless of the modality of interaction. Both are divisible into segments (discourse segments and phrases) with the meaning of the segments being more than the meaning of the individual parts.

The main focus of the research is the interpretation beyond sentence boundaries, the intentional and informational approach.

According to the informational approaches, the coherence of discourse follows from semantic relationships between the information conveyed by successive utterances. As a result, the major computational tools used here are inference and abduction on representations of the propositional content of utterances.

According to the intentional approaches the coherence of discourse derives from the intentions of speakers and writers and understanding depends on recognition of those intentions.

One difficulty is to build models of human-machine-dialog when initially only examples of human-human interaction exist, which may not be relevant. Bootstrapping suitable models will therefore require Wizard-of-Oz studies with simulated systems.

Natural Language Generation

In many of the applications mentioned above, systems need to produce high-quality natural language text from computer-internal representations of information. Natural language generation can be decomposed into the tasks of text planning, sentence planning and surface realization. Text planners select from a knowledge pool which information to include in the output and out of this create a text structure to ensure coherence. On a more local scale, sentence planners organize the content of each sentence, massaging and ordering its parts.

Surface realizers convert sentence-sized chunks of representation into grammatically correct sentences.

Generator processes can be classified into points on a range of sophistication and expressive power, starting with inflexible canned methods and ending with maximally flexible feature combination methods. It is safe to say that at the present time one can fairly easily build a single-purpose generator for any specific application, or with some difficulty adapt an existing sentence generator to the application, with acceptable results. However, one cannot yet build a general-purpose sentence generator or a non-toy text planner. Several significant problems remain without sufficiently general solutions:

- Lexical selection is one of the most difficult problems in generation. At its simplest this question involves selecting the most appropriate single word for a given unit of input. However as soon as the semantic model approaches a realistic size and as soon as the lexicon is large enough to permit alternative locutions, the problem becomes very complex. The decision depends on what has already been said, what is referentially available from context, what is most salient, what stylistic effect the speaker wishes to produce and so on. What is required: development of theories about and implementations of lexical selection algorithms, for reference to objects, events states, etc., and tested with large lexical.

- Discourse structure (see also there) So far, no text planner exists that can reliably plan texts of several paragraphs in general. What is required: Theories of the structural nature of discourse, of the development of theme and focus in discourse, and of coherence and cohesion; libraries of discourse relations, communicative goals and text plans: implemented representational paradigms for characterizing stereotypical texts such as reports and business letters; implemented text planners that are tested in realistic non-toy domains.

- Sentence planning: Even assuming the text planning problem is solved, a number of tasks remain before well-structured multi-sentence text can be generated: These tasks, required for planning the structure and content of each sentence, include: pronoun specification, theme signaling, focus signaling, content aggregation to remove unnecessary redundancies, the ordering of prepositional phrases, adjectives, etc. What is required: Theories of pronoun use, theme and focus selection and signaling, and content aggregation; implemented sentence planners with rules that perform these operations; testing in realistic domains.

- Domain modeling: a significant shortcoming in generation research is the lack of large, well-motivated application domain models, or even the absence of clear principles by which to build such models. A traditional problem with generators is that the inputs are frequently hand-crafted, or are built by some other system that uses representation elements from a fairly small hand-crafted domain model, making the generator's inputs already highly oriented toward the final language desired….What is required: Implemented large-size (over 10.000 concepts) domain models that are useful both for some non-linguistic application and for generation; criteria for evaluating the internal consistency of such models; theories on and practical experience in the linking of generators to such models: lexicon of commensurate size.

Probably the problem least addressed in generator systems today is the one that will take the longest to solve. This is the problem of guiding the generation process through its choices when multiple options exist to handle any given input.

The generator user has to specify not only the semantic content of the desired text, but also its pragmatic – interpersonal and situational – effects. Very little research has been performed on this question beyond a handful of small-scale pilot studies. What is required: Classifications of the types of reader characteristics and goals, the types of author goals, and the interpersonal and situational aspects that affect the form and content of language; theories of how these aspects affect the generation process; implemented rules and/or planning systems that guide generator systems' choices; criteria for evaluating appropriateness of general text in specified communicative situations.

Effective presentations require the appropriate selection of content, allocation to media, and fine grained coordination and realization in time and space. Discovery and presentation of knowledge may require mixed media (e.g., text, graphics, video, speech and non-speech audio) and mixed mode (e.g., linguistic, visual, auditory) displays tailored to the user and context. This might include tailoring content and form to the specific physical, perceptual, or cognitive characteristics of the user. It might lead to new visualization and browsing paradigms for massive multimedia and multilingual repositories that reduce cognitive load or task time, increase analytic depth and breadth, or simply increase user satisfaction. A grand challenge is the automated generation of coordinated speech, natural language, gesture, animation, non-speech audio, generation, possibly delivered via interactive, animated lifelike agents. Preliminary experiments suggest that, independent of task performance, agents may simply be more engaging/motivating to younger and/or less experienced users.

Ontologies

Large-scale ontologies are becoming an essential component of many applications including standard search (such as Yahoo and Lycos), e-commerce (such as Amazon and eBay), configuration (such as Dell and PC-Order), and government intelligence (such as DARPA's High Performance Knowledge Base program). As discussed in the preceding paragraphs, ontologies will constitute a major source of knowledge needed for several levels of NLP.

Ontologies are increasingly seen as an important vehicle for describing the semantic content of web-based information sources and they are becoming so large that it is not uncommon for distributed teams of people to be in charge of the ontology development, design, population, and maintenance.

Ontologies define a vocabulary for researchers who need to share common understanding of the structure of information in a domain. It includes machine-interpretable definitions of basic concepts in the domain and relations among them. The principal reasons to use an ontology in machine translation (MT) and other language technologies are to enable source language analyzers and target language generators to share knowledge, to store semantic constraints and to resolve semantic ambiguities by making inferences using the concept network of the ontology. An ontology contains only language independent information and many other semantic relations as well as taxonomic relations.

Though the utility of domain ontologies is now widely acknowledged in the IT (Information Technology) community, several barriers must be overcome before ontologies become practical and useful tools. One important achievement would be to reduce the time and cost of identifying and manually entering several thousand concept descriptions by developing

automatic ontology construction. Another important task is to find arrangements that make development and sharing of ontologies commercially attractive.

Some challenges for ontology research:

Work on ontologies needs to provide generally applicable top-ontologies that cover most important core concepts that will be needed for many domains. Extensions to new domains could then start by enriching these top-ontologies in a specific direction, reducing the initial effort for creating new ontologies, for merging independently developed extensions, and for rapid customisation of existing ontologies.

This requires that ontology-creators are willing to share parts of their work and find suitable processes to organize cooperation. It also requires the development of standards for the languages in which ontologies are specified and can be interchanged (e.g. along the lines of the OIL proposal). Here, the challenge is to find suitable compromises between expressive power and depth on one hand and ease of use on the other hand. Ideally, one specification language should be able to cover the whole spectrum up to advanced knowledge representation as used in the CYC project.

Incremental improvement of ontologies needs to be facilitated by specialized tools for easy visualization and modification. These tools (and the representations they work on) need to be domain-independent and suited even for casual users, and their design needs to be based on a user-centred process view.

It must be easy to plug in ontologies into various NLP-based tools such as tools for information extraction, organization and annotation of document collections (semantic Web), environments for terminology management and controlled language. This will permit to audit the contained knowledge in manifold ways, and will allow for rapid quality improvement.

What is required: tools that support broad ranges of users in (1) merging of ontological terms from varied sources, (2) diagnosis of coverage and correctness of ontologies, and (3) maintaining ontologies over time.

Lexicons

Lexical knowledge – knowledge about individual words in the language – is essential for all types of natural language processing. Developers of machine translation systems, which from the beginning have involved large vocabularies, have long recognized the lexicon as a critical (and perhaps the critical) system resource. As researchers and developers in other areas of natural language processing move from toy systems to systems which process real texts over broad subject domains, larger and richer lexicons will be needed and the task of lexicon design and development will become a more central aspect of any project.

A basic lexicon will typically include information about morphology and on the syntactic level, the complement structures of each word or word sense. A more complex lexicon may also include semantic information, such as a classification hierarchy and selectional patterns or case frames stated in terms of this hierarchy. For machine translation, the lexicon will also have to record correspondences between lexical items in the source and target language; for speech understanding and generation, it will have to include information about the pronunciation of individual words. For this purpose the overall lexicon architecture and the representation formalism used to encode the data are important issues.

No matter if we want to build an ontology or a lexicon, in general for this kind of high-quality semantic knowledge base, manual processing is indispensable. Traditionally computer lexicons have been built by hand specifically for the purpose of language analysis and generation. However, the needs for larger lexicons are now leading to efforts for the development of common lexical representations and co-operative lexicon development.

The area is ripe – at least for some levels of linguistic description – for reaching in the short term a consensus on common lexical specifications. We must expand the experiences with the sorts of semantic knowledge that could be effectively used by multiple systems. We must also recognize the importance of the rapidly growing stock of machine-readable text as a resource for lexical research. The major areas of potential results in the immediate future seem to lie in the combination of lexicon and corpus work. There's a growing interest from many groups in topics such as sense tagging or sense disambiguation on very large text corpora, where lexical tools and data provide a first input to the systems and are in turn enhanced with the information acquired and extracted from corpus analysis.

Machine Learning

As mentioned above, the acquisition of knowledge continues to impose on of the biggest difficulties to the application of NLP technologies. This holds both for linguistic knowledge (grammars lexicons) and for world knowledge (ontologies, facts). In order to make extensions of NLP to new domains possible, the acquisition process needs to be supported by algorithms that can exploit existing textual material and extract knowledge of various types from it.

Approaches to these methods can be found in various fields of research, such as statistical language models, bilingual alignment, grammar induction, statistical parsing, statistical classification technology, Bayesian networks and other ML methods used in artificial intelligence research, data mining techniques etc.

Due to the specific nature of lexical information, it is important to pick or develop methods that scale to large vocabularies and large sets of features and that can exploit multiple sources of evidence in a good way. Also, the methods need to be able to use a rich set of existing background knowledge, so that no effort is wasted in re-discovering what was already known.

It is important to have methods that can use richly annotated training data, but do not require that large datasets have to be annotated in this way. Instead, methods should be able to draw a maximum of advantage from raw data without annotation using unsupervised learning approaches. Also, it will be important to guide the effort of human annotation so that time is spent in the most efficient way, using active learning methods. Tools and processes for managing annotation projects (including assessment of quality levels) need to be developed and shared on a broad basis.

Whenever possible, one should try to use models that contain explicit linguistic representations (ideally organized along different strata) so that partial reuse of models and rapid adaptation to slightly different is facilitated.

4. Milestones

Some relevant items not included in Bernsen 2000.

Basic technologies

Short term

- accurate syntactic analysis for well-formed input from specific domains
- simple methods for minimizing annotation effort during domain adaptation
- ML algorithms that combine active and unsupervised learning for optimal exploitation of data
- generally applicable annotation schemes for semantic markup of text
- standards for encoding and exchange of ontological resources emerge
- top-level ontologies generally available
- tools for semi-automatic construction and population of ontologies from text
- tools for simple semantic enrichment of Web pages
- approaches to markup of discourse structure and pragmatics

Medium term

- improved methods for minimizing annotation effort during domain adaptation
- tools for adaptation of syntactic analysis to specific application with minimal human effort
- accurate syntactic analysis for slightly ill-formed input for restricted domains
- improved syntactic analysis of input with uncertainties (word lattices)
- machine learning methods that exploit and extend existing knowledge sources
- sufficiently accurate semantic analysis of free text from restricted domains
- generic schemes for the annotation of pragmatic content
- schemes for annotation of discourse and document structure
- generally usable ontologies exist for many domains
- NL generation verbalizes information extracted/deduced from multiple sources for QA
- Agent/user models for dialogs of moderate complexity

Long term

- accurate syntactic analysis for ill-formed input from multiple domains
- sufficiently accurate semantic analysis of free text from multiple domains
- recognition of pragmatic content in text and dialog
- NL generation produces stylistically adequate and well-structured text

Systems

Short term

- QA systems are able to answer simple factual questions
- Summarization system produce well-formed extracts from short documents
- automated e-mail response systems deliver high-quality replies in easy cases
- MT for information assimilation

Medium term

- QA systems that deduce answers from information in multiple sources
- Summarization systems are able to merge multiple documents
- Summarization systems are able to deliver different types of summaries
- Integration of translation memories with MT enables fast domain-adaptation
- Mixed-initiative dialogue systems for services and e-commerce

Long term

- Translator's workbenches based on TM, MT, and multi-modal input facilities
- QA systems that are able to explain their reasoning

5. Recommendations for NLP research in Europe

1. Build and make publicly available at low cost large-scale multilingual lexical resources, with broad coverage, generic enough to be reusable in different application frameworks
2. To turn special attention to the development of better ontologies which are reusable across domains in order to encode static world knowledge
3. Creation of large common accessible multilingual corpora of syntactical and semantically annotated data annotated also beyond sentence boundaries

4. Encourage development of statistical and machine-learning methods that facilitate bootstrapping of linguistic resources

5. Common standards will improve the effectiveness of people's cooperation, the identification of the requirements for the system specification, the inter-operability among systems and the possibility of re-using and sharing system components.

6. Integration of language processing into the rest of cognitive science, artificial intelligence and computer science e.g. some ambitious projects centered on NL but combining various techniques and different areas of AI. New type of projects: Very different for scale, ambition and timeframe

7. Establishment of centers of excellence as focus points for projects for a period of five to ten years.

8. Encourage systematic evaluations (but how ?)

6. References

- Berners-Lee, T. (2001) The Semantic Web, Scientific American (5/2001)

- Bernsen , N.O. (2000) Speech-Related Technologies. Where will the field go in 10 years? roadmap workshop, Katwijk

- Burger, J. e.a. (2000) Issues, Tasks and Program Structures to Roadmap Research in Question & Answering, Memo National Institute of Standards and Technology, Gaithersburg

- Carbonell, J. e.a. (2000) Vision Statement to Guide Research in Q&A and Text Summarization, Memo National Institute of Standards and Technology, Gaithersburg

- Cole,R.A. (Ed.). (1997) Survey of the State of the Art in Human Language Technology Cambridge University Press, Cambridge

- Declerck, Th., Wittenburg, P., Cunningham, H. (2001) The Automatic Generation of Formal Annotations in a Multimedia Indexing and Searching Environment, ACL Workshop, Toulouse

- Delannoy, J.-F. (2001) What are the points? What are the stances? Decanting for question-driven retrieval and executive summarization, ACL Meeting, Toulouse

- Fensel, D. Hendler, J., Lieberman, H. ,Wahlster, W. (2000) Dagstuhl-Seminar: Semantics for the WWW, Dagstuhl, Germany

- Grishman, R. and Calzolari, N. Lexicons in Survey of the State of the Art in Human Language Technology, Cambridge University Press, Cambridge

- Grosz, B. (1997) Discourse and Dialogue in Survey of the State of the Art in Human Language Technology, Cambridge University Press, Cambridge

- Heisterkamp, P., (2000) Speech Technology in the year 2010, roadmap workshop, Katwijk
- Hirschman, L. and Thompson, H.S. (1997) Evaluation in Survey of the State of the Art in Human Language Technology, Cambridge University Press, Cambridge
- Hovy, E., (1997) Language Generation in Survey of the State of the Art in Human Language Technology, Cambridge University Press, Cambridge
- Kang, S.-J. and Lee, J.-H. (2001) Semi-Automatic Practical Ontology Construction by using Thesaurus, Computational Dictionaries, and Large Corpora, ACL workshop Toulouse
- Kay, M. (1997) Machine Translation: The Disappointing Past and Present. In: Survey of the State of the Art in Human Language Technology, Cambridge University Press, Cambridge
- Kay, M. (1997) Multilinguality. In: Survey of the State of the Art in Human Language Technology, Cambridge University Press, Cambridge
- Knight, K. (2001) Language Modeling for Good Generation, Workshop on Language Modeling and Information Retrieval, Pittsburgh
- Krauwer, St., (2000) Going from 'what' to 'why' across language barriers in the unified distributed information space. Roadmap workshop, Katwijk
- Maybury, M.T. and Mani, I., (2001) Automatic Summarization, ACL Meeting Toulouse
- Maybury, M.T., (2001) Human Language Technologies for Knowledge Management: Challenges and Opportunities, ACL Meeting, Toulouse
- Pardo, J.M., (2000) How will language and speech technology be used in the information world of 2010? Research challenges & Infrastructure needs for the next ten years. Report on the Roadmap Workshop, Katwijk aan Zee
- Staab, St., (2001) Knowledge Portals, ACL Meeting, Toulouse
- Stock, O. (2000) Processing Natural Language from 2000 to 2010, roadmap workshop, Katwijk
- Velardi, P. and Missikoff, M. and Basili, R. (2001) Identification of relevant terms to support the construction of Domain Ontologies, ACL workshop Toulouse
- Uszkoreit; H. (2001) Crosslingual Language Technologies for Knowledge Creation and Knowledge Sharing, Toulouse
- Zaenen, A. and Uszkoreit, H. (1997) Language Analysis and Understanding. In: Survey of the State of the Art in Human Language Technology, Cambridge University Press, Cambridge

The Workshop Programme

Agenda (Morning Session)

Technical Papers (**8:00-9:20**)

8:00-8:20 *'Multilingual Terminology Databanks for Web Mining*'
António Ribeiro, Universidade Nova de Lisboa

8:20-8:40 *'Grammar Learning by Partition Search*'
Anja Belz, ITRI University of Brighton

8:40-9:00 *'A Semantic-driven Approach to Hypertextual Authoring'*
R. Basili, A. Moschitti, M.T. Pazienza and F.M. Zanzotto, University of Rome, Tor Vergata

9:00-9:20 *'Advantages of ontology-based user profiling in the NAMIC project'*
Jan De Bo and Ben Majer, VUB STARLab, University of Brussels

9:20-10:05 Invited Talk *'Methods, representation and linguistic bias in Adaptive IE'*
Roberto Basili, University of Rome, Tor Vergata

10:05-10:25 Coffee

Technical Papers (**10:25-11:45**)

10:25-10:45 '*Description of Events: An Analysis of Keywords and Indexical Names'*
Khurshid Ahmad, Paulo C F de Oliveira, Pensiri Manomaisupat, Matthew Casey and Tugba Taskaya, University of Surrey

10:45-11:05 *'Learning IE patterns: a terminolgy extraction perspective.'*
Roberto Basili, Maria Teresa Pazienza, Fabio Massimo Zanzotto, University of Rome, Tor Vergata

11:05-11:25 *'Unsupervised Event Clustering in Multilingual News Streams'*
Martijn Spitters and Wessel Kraaij, Department of Multimedia Technology & Statistics, The Netherlands

11:25-11:45 *'Large-scale Multilingual Information Extraction'*
R. Catizone, A. Setzer and N. Webb, Univeristy of Sheffield

11:45-12:30 Invited Talk*: 'User Driven Information Extraction for the Web'*
Fabio Cirevegna, University of Sheffield

12:30-1:30 Panel and Round Table on *Adaptive Technologies and their implications on advanced HLT applications (IR, IE, Q&A and KM)*

panelists:

Nino Varile (EC Commission)
F. Gardin (AISoftware)
Y. Wilks (Univeristy of Sheffield)
M.T. Pazienza (University of Rome, Tor Vergata)
Francesco Danza (Knowledge Stones S.p.A.)
Remi Zajac (Systran Software)

Workshop Organisers

Roberta Catizone, University of Sheffield
Roberto Basili, University of Rome, Tor Vergata
Maria-Teresa Pazienza, University of Rome, Tor Vergata
Maria-Vittoria Marabello, Knowledge Stones S.p.A., Rome

Workshop Programme Committee

Roberta Catizone	University of Sheffield
Walter Daelemans	CNTS/Language Technology Group, Antwerp
M. V. Marabello	KnowledgeStones S.p.A
M. T. Pazienza	University of Rome, Tor Vergata
G. Rigau	Polytechnical University of Catalunia
Horatio Rodriguez	Polytechnical University of Catalunia
A. Setzer	University of Sheffield
N. Webb	University of Sheffield
Y. Wilks	University of Sheffield
Rémi Zajac	Systran Software, CA
F.M. Zanzotto	University of Rome, Tor Vergata

Table of Contents

Author Index

Multilingual Terminology Databanks for Web Mining

António Ribeiro*, Gabriel Lopes* and João Mexia[+]

Universidade Nova de Lisboa
Faculty of Sciences and Technology, Department of *Informatics/[+]Mathematics
Quinta da Torre, Monte da Caparica, P–2829–516 Caparica, Portugal
{ambar, gpl}@di.fct.unl.pt

Abstract

This paper presents on-going research on a methodology to build multilingual terminology databanks from parallel texts in several languages. Web mining is a potential application for these databanks for they allow not only multilingual document content modelling but also multilingual retrieval of documents matching a user's query in different languages. We start by describing a methodology to align parallel texts and to extract multiword term Translation Equivalents, using language independent and statistically supported techniques. Next, we present some approaches for multilingual mining in order to provide the context for this work. Finally, we discuss how these multilingual terminology databanks can be used in the framework of multilingual mining.

1. Introduction

In an increasingly multilingual Web, monolingual web search engines have become unable to mine the web and retrieve simultaneously documents written in different languages for a query made in a particular language. It may be the case that the most relevant documents are not written in the language the query was made; the user may not know what language to choose to retrieve the best documents. Current monolingual search engines cannot help on this. Thus, it would be wise to have multilingual web miners which would allow multilingual web searches and provide the user either the original document if the user understands the language it is written in or a translation of the document in a language selected by the user. Multilingual web search engines must be able to cope with Cross-Language Information Retrieval (CLIR) if they are to satisfy their customers. CLIR addresses precisely the possibility of making queries in one language and retrieving relevant documents in other languages (Brown *et al.*, 2000).

Google Inc., the company that owns the popular Google web search engine, has recently released a note stressing an increase in the number of web pages written in languages other than English: "Of the 2 billion web pages in Google's index [http://www.google.com], more than a quarter are in languages other than English" (Google Inc., 2001a). In fact, English is steadily becoming less *the* language of the Web. As more web pages are written in other languages, web searches are doomed to be confined to the documents written in the same language of the query if web search engines are not able to handle searches in multilingual documents. In a world which promotes information exchange, this seems to do the opposite through divisions and to raise the issue whether a 'Multilingual Information Society'[1] can actually be real.

Although users' experience says that they are better off with English for Web searches world wide, a press release also from Google Inc. (2001c) reports a growing trend in the number of web searches done in languages other than English in its own web search engine.

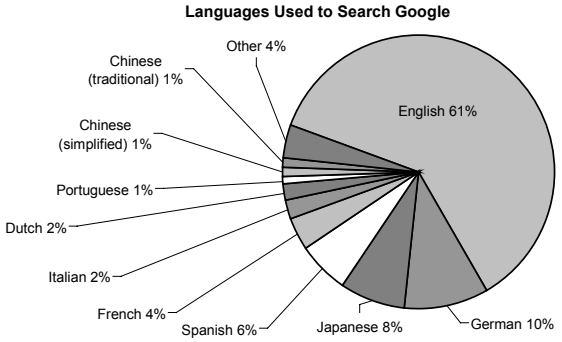


Figure 1: Languages used for searches with the Google web search engine in 2001 October (Google Inc., 2001c).

As Figure 1 shows, more than a third of the web searches done in 2001 October in the Google web search engine were in languages other than English[2].

This current trend emphasises the use of web engines for searches in various languages and again puts this work into perspective. In Europe alone it is often the case that each country has its own set of official languages and possibly even other regional languages, as it is the case of Spain (Basque, Castilian – commonly referred to as Spanish –, Catalan and Galician), Switzerland (French, German, Italian and Romansh) or the United Kingdom (English, Gaelic and Welsh). Consequently, should a query be done in one of these particular languages, a monolingual web search engine is bound to limit the query to documents written only in the query language.

This paper proposes using multilingual terminology databanks in order to model the contents of multilingual documents and, thus, to provide multilingual access. It describes a method to build multilingual terminology databanks from parallel texts in order to provide an extra multilingual layer for web search and enable searches in documents written in several languages.

[1] This is the name of a programme supported by the European Commission, which aims at protecting and safeguarding pluralism, diversity and the principle of equality among all languages (http://www.hltcentral.org/page-762.0.shtml).

[2] The same trend had also been noticed earlier in 2001 August (Google Inc., 2001b), before the 2001 September 11 attacks in New York, USA, which could have biased the results since more users would be choosing English to search, for example, 'World Trade Centre' or 'Anthrax'.

This paper is structured as follows: the next section gives an overview of what parallel texts are and how they can be aligned. Section 3 describes how to build multilingual terminology databanks from the aligned parallel texts. Section 4 describes how the multilingual databanks can be used for multilingual mining and presents several methodologies that have been proposed. Finally, section 5 presents some conclusions and section 6 draws some future work.

2. Aligning Parallel Texts

In this section we will describe several approaches to parallel texts alignment. First, we start by describing what parallel texts are. Section 2.2 and 2.3 present previous sentence and word alignment methodologies. Section 2.4 describes the alignment method we used.

2.1. Parallel Texts

Parallel texts are sets of texts which are translations of each other in different languages, like the proceedings of the Canadian Parliament – the Canadian Hansards –, which are published in both English and French, or the Official Journal of the European Communities published in the eleven official languages of the European Union[3]. They have proven to be rich linguistic resources for multilingual text processing and they have become available in a wide range of languages,

However, before it is possible to use them to identify translations of multilingual terms, parallel texts must be aligned first. Text alignment aims at establishing correspondences between parallel texts automatically, either between paragraphs, sentences, or even at sub-sentential level between text segments, phrases, words or sequences of characters.

There have been mainly two approaches to alignment of parallel texts: *sentence alignment* establishes correspondences between sentences only and *word alignment* tries to go a bit deeper into sub-sentential level by establishing correspondences between text segments or words.

2.2. Previous Sentence Alignment Techniques

Back in the early 1990s, sentences were considered as the basic units for alignment. Texts were split into sequences of sentences and alignment algorithms would attempt at making correspondences between the sentences in the parallel texts.

Kay and Röscheisen (1993) were the first to propose an alignment methodology. They assumed that for two sentences written in different languages to correspond, the words in them must also correspond. Their algorithm started by suggesting a tentative alignment of sentences by aligning the first and the last ones of each parallel text. Then, equivalent words were used to align the others. Two words were considered equivalent if they tended to co-occur in the same tentatively aligned sentences. A measure of similarity was computed and if it scored higher than a specific value, it would mean those words were indeed translations. Finally, sentences were aligned if the number of words associating them was greater than an empirically defined threshold.

In other alternative approaches, less knowledge based, sentences were aligned if they had a proportional number of words (Brown *et al.*, 1991) or characters (Gale and Church, 1991). Each of these authors started from the fact that long sentences tend to have long translations and, conversely, short sentences tend to have short translations. This correlation was the basis for their statistical models. Their algorithms would group sequences of sentences till they had proportional sizes

Brown *et al.* (1991, p. 175) remarked that the error rate was slightly reduced from 3.2% to 2.3% when using some linguistic knowledge like time stamps, question numbers and author names found in the parallel texts. This confirmed the fact that it was sufficient to look at sentence lengths in order to align sentences. Extra linguistic knowledge did not improve the results significantly.

Simard *et al.* (1992) proposed a sentence alignment algorithm which would first align text segments based on the length-based algorithm suggested by Gale and Church (1991), and, if it did not produce a 'clear' single best alignment of two text segments, it would proceed into a second pass counting the number of *cognates* shared between them.

According to the Longman Dictionary of Applied–Linguistics, a *cognate* is "a word in one language which is similar in form and meaning to a word in another language because both languages are related" (Richards *et al.*, 1985, p. 43). For example, the words *Parliament* and *Parlement*, in English and French respectively, are cognates. However, if two words have the same or similar forms in two languages but different meanings, they are called false cognates or false friends (Richards *et al.*, 1985, p. 103). For example, the English word *library* and the French word *librairie* are an example of false cognates (Melamed, 1999, p. 114): *library* is translated as *bibliothèque* in French and, conversely, *librairie* as *bookstore* in English.

Simard *et al.* (1992) used a simple rule to test if two words were cognates by checking whether their first four characters were identical (Simard *et al.*, 1992, p. 71), as in *Parliament* and *Parlement*. This simple heuristic proved to be quite useful, providing a great number of lexical cues for alignment though it has some shortcomings. According to it, the English word *government* and the French word *gouvernement* are not cognates. Also, *conservative* and *conseil* ('council'), in English and French respectively, are wrongly considered as cognates (Melamed, 1999, p. 113). The rule is sensitive to variations in the first four letters but it does not distinguish different word endings.

In order to align English and Chinese sentences, Wu (1994) also used the method based on proportional lengths. He also began by applying a method similar to the one used by Gale and Church (1991) and reported results not much worse than those expected by this algorithm. Still, he claimed sentence alignment precision over 96% when the method incorporated a seed bilingual lexicon of words commonly found in the texts to be aligned (e.g. names of months, like *December* and its equivalent in Chinese 十二月). So, again Wu's work confirmed that the use of lexical cues would be beneficial for alignment.

The problem with the alignment algorithms which rely solely on sentence sizes is that they tend to break down

[3] Danish (da), Dutch (nl), English (en), Finnish (fi), French (fr), German (de), Greek (el), Italian (it), Portuguese (pt), Spanish (es) and Swedish (sv).

when sentence boundaries are not clearly marked in the parallel texts. Sentences need to be clearly identified which means taking the most advantage of the cues provided by full stops. Full stops have to be clearly interpreted in order to check whether they mark a sentence boundary. However, that is not always the case.

Gale and Church (1991, p. 179) reported that only 53% of the full stops found in the Wall Street Journal were used to mark sentence boundaries. Full stops may be part of abbreviations (*Dr. A. Bromley*), numbers (*1.3%*), they are not usually found in headlines (*Tyre production*), they may not even exist because they were not added, or they were either lost or mistaken for noise in the early days when electronic versions of parallel texts were still rare and texts needed to be scanned.

2.3. Previous Word Alignment Techniques

Word alignment is much more fine-grained than sentence alignment since it is no longer done just at sentence level but at word level. Aligned text segments are shorter and, thus, it becomes easier to establish correspondences. However, in contrast with sentence alignment algorithms which permit a margin of tolerance for occasional wrong word matches since sentences generally have many words, they are no longer 'safety nets' for word level alignment. Consequently, the penalty on wrong word matches becomes higher and achieving a high precision becomes harder. Should word alignment be the goal, the alignment algorithm must be more 'careful' in order to avoid wrong word matches.

Church (1993) showed that by adding some lexical information, alignment of parallel text segments was possible without requiring sentence delimiters. He exploited the notion of orthographic cognates proposed earlier by Simard *et al.* (1992). He used a similar rule: use equal 4-grams in order to find 'cognate' (similar) sequences of characters in the parallel texts, i.e. sequences of four characters which are equal in the texts. The method built a graph where a dot at co-ordinates (x, y) meant that there was a match between the 4-grams in positions x and y of both texts. The reliable dots were filtered using an empirically estimated search space.

Fung and Church (1994) dropped the requirement for clear sentence boundaries on a case-study for English-Chinese. It was also the first time alignment procedures were being tested on texts between non-Latin languages and without finding sentence boundaries. Each parallel text was split into K pieces and word correspondences were identified by analysing their distribution across those pieces. In particular, a binary vector of occurrences with size K (hence, the K-vec) would record the occurrence of a word in each of the pieces. Should the word occur in the i-th piece of the text, then the i-th position of the vector would be set to '1'. Next, the K-vecs of English and Chinese words were compared in order to find whether two words corresponded. In this way, it was possible to build a rough estimate of a bilingual lexicon to feed the algorithm of Church (1993). In this case, dots would be drawn in the graph each time two translations occurred.

This method was extended in Fung and McKeown (1994). It was also based on the extraction of a small bilingual dictionary based on words with *similar distributions* in the parallel texts. However, instead of K-vecs, which stored the occurrences of words in each of the K pieces of a text, Fung and McKeown (1994) used vectors that stored the distances between consecutive occurrences of a word (DK-vec's). For example, if a word appeared at offsets (2380, 2390, 2463, 2565, ...), then the corresponding distances vector would be (10, 73, 102, ...). Should an English word and a Chinese word have similar distance vectors, then they would be used as potential cues for alignment.

In Simard and Plamondon (1998), sentences were aligned using 'isolated' cognates as anchors, i.e. cognates that were not mistaken for other cognates within a text window whose width was set to 30% of the text size. Yet, the alignment algorithm would start by aligning words. Each occurrence of a cognate became a dot in a graph according to its offset in each of the parallel texts. Some of those points were filtered if they lied outside an empirically defined search space which would mean they were "not in line" with their neighbouring points. The heuristic values used were found empirically so as to provide the best results and make the best selection of the good alignment cues.

Melamed (1999) also used orthographic cognates. His algorithm filtered noisy correspondence points, i.e. points which were not reliable, according to several heuristics which helped define what a good anchor was. In order to measure word similarity, he defined the ratio of the Longest Common Sub-sequence of characters as follows:

$$Ratio(w_1, w_2) = \frac{Length(Longest\ Common\ Sub\text{-}Sequence(w_1, w_2))}{Max(Length(w_1), Length(w_2))}$$

where w_1 and w_2 are the two words to be compared (Melamed, 1999, p. 113). This measure compares the length of the longest common sub-sequence of characters with the length of the longest token. For example, for *government* and *gouvernement*, the ratio is 10 (the length of *government*) over 12 (the length of *gouvernement*) whereas the ratio is just 6 over 12 for *conservative* and *conseil* ('council'). This measure tends to favour long sequences similar to the longest word and to penalise sequences which are too short compared to a long word. So, for this very reason, it fails to consider *gouvernement* and *governo* in French and Portuguese as cognates because *governo* is shorter. Their ratio is also 6 over 12.

For alignment purposes, Melamed (1999) selects all pairs of words which have a ratio above a certain threshold, empirically selected. Still, this comparison measure seems to provide better results than the one first proposed by Simard *et al.* (1992) but it is not also based on a statistically supported study.

2.4. The Alignment Methodology

In contrast with the previous approaches, Ribeiro *et al.* (2000) present a statistically supported method for word alignment of parallel texts which does not require either clearly delimited sentences or previous linguistic knowledge of the texts languages. It was applied to parallel texts in the 11 official languages of the European Union and also to parallel texts in Portuguese and Chinese (Ribeiro *et al.*, 2001a).

In particular, the alignment methodology selects alignment points using filters based on linear regression lines properties. The points are generated from the offsets of lexical cues provided by equal tokens (like numbers, proper names, punctuation marks) which occur with the

same frequency within a parallel text segment. Since the algorithm is recursive, even if some token happens not to have the same 'global' frequency, it may end up being used as an alignment point in a 'local' analysis of smaller text segments.

This algorithm was later extended in Ribeiro *et al.*, (2001b) to handle typical sequences of characters common to a particular pair of languages. Instead of using heuristics to identify cognate words or of using particular sizes of *n*-grams of characters to find similar sequences of characters, they made statistical data analyses of contiguous and non-contiguous sequences of characters to extract associated character units from each pair of languages. They were able to find typical sequences of characters in the beginning of words, such as •*Comis*, for *Comissão* and *Comisión* ('Commission') in Portuguese and Spanish, in the middle of words, as in *f_rma* which matches both *information* and *informação* in English and Portuguese respectively, or across word boundaries, as *i_re•ci* for the Portuguese–French pair as in *livre•circulação* and *libre•circulation* ('free movement').

The average alignment precision is over 90% for aligned parallel texts in Portuguese with all the other official languages of the European Union. This is the precision of a word alignment algorithm which, in contrast with other algorithms, does not rely on language specific knowledge, lists of stop words to avoid noise generated by frequent words or extra seed bilingual lexicons.

3. Building Multilingual Databanks

Aligned parallel texts are ideal sources to extract Translation Equivalents for they provide the correspondences between the original text and their translations in other languages. They allow easily the examination of the way specific words or terms are translated into other languages. Consequently, they can reduce the amount of effort necessary to build Translation Databanks.

For this experiment we used a sample of parallel texts from three sources: records of the Written Questions to the European Commission (ELRA, 1997), records of Debates in the European Parliament (ELRA, 1997) and Judgements of The Court of Justice of the European Communities in all the languages of the European Union.

In order to identify relevant multiword units, it has been common practice to do it by hand coding regular syntactic patterns, like the sequence 'Noun Noun' (e.g. *Web Mining*). Finite state automata are then used to recognise typical sequences of words in the texts which comply with these patterns. For example, Daille (1995) used several syntactic patterns to identify terms with two words such that they were either two nouns or a noun and an adjective, as in *liaison par satellite* ('satellite link') or *station terrienne* ('earth station'). Fung and McKeown (1997) also used specific syntactic patterns to extract multiword terms in order to compile a list of reliable pairs of translations for a further extension to their previous alignment algorithms (Fung and McKeown, 1994).

Although terms are generally covered by some characteristic patterns, this work has not started from a particular set of patterns so as not to constrain the structure of the multiword units.

In order to build the Multilingual Terminology Databank, we extracted terms from the parallel texts using a methodology described in da Silva *et al.* (1999). This methodology is based on the idea that the more cohesive a group of *n* words is, the higher its cohesiveness score. The algorithm assumes that the score of a good multiword unit must be a local maximum, i.e. the cohesion of the set of *n* words is higher than any subset of *n*–1 words contained in it and higher than the cohesion of any superset of *n*+1 words which contains it. Thus, the algorithm is able to select, for example, *common rules and standards* as a relevant multiword term but not *common rules and* or *common rules and standards for*, because the scores of these multiword units are lower. The figure below shows some extracted terms:

English	French	Portuguese
combined nomenclature	*autorités douanières*	*autoridades aduaneiras*
customs authorities	*États membres*	*Estados – Membros*
intervention agency	*matières nucléaires*	*materiais nucleares*
Member States	*nomenclature combinée*	*Nomenclatura Combinada*
nuclear material	*organisme d'intervention*	*organismo de intervenção*

Table 1: A sample of extracted multiword terms in English, French and Portuguese.

The methodology has proven to be quite adequate to be used across several languages. In this way, we were able to capture multiword terms for each language and build databanks of terms. However, it still remains to be seen how the relations between them can be established, i.e. how to build the multilingual terminology databank of equivalent translations.

The key issue in the extraction of Translation Equivalents is to find a correlation between co-occurrences of terms in the aligned parallel texts. In general, if two terms co-occur often in aligned text segments, then they are likely to be *equivalent*.

The alignment of parallel texts splits them into small aligned text segments and reduces the number of words / terms that must be checked for co-occurrence in each parallel text segment. The shorter the segments, the better. In order to identify Translation Equivalents, the *distribution similarity* of words / terms must be analysed in the aligned segments.

Following the conventional information retrieval methodology (Salton and McGill, 1983), the information on the occurrence of words (or terms) is usually represented in vector forms. For example, if a word *w* occurs in segments 1, 2 and 5 out of a total of five segments, then the following *occurrences vector* is built: *w* = (1, 1, 0, 0, 1). In this binary vector, each '0' and '1' represents the absence and presence of the word *w* in each of the five segments.

In this way, a set of occurrence vectors can be built for each of the terms found. Next, for each pair of source and target terms a *co-occurrence vector* is built where the *i*-th position of the vector is set to '1' if both terms occur in the *i*-th aligned text segment. Next, a *contingency table* is built for each pair of source–target terms by counting the number of '0's and '1's in the occurrences vectors.

n: 162347	*Επιτροπή των Ευρωπαϊκών Κοινοτήτων*	*× Επιτροπή των Ευρωπαϊκών Κοινοτήτων*
Comissão das Comunidades Europeias	*a*: 499	*b*: 102
× Comissão das Comunidades Europeias	*c*: 96	*d*: 161650

Table 2: Contingency table for the pair *Επιτροπή των Ευρωπαϊκών Κοινοτήτων* (Epitropé tos Europaikós Koinotétos) and *Comissão das Comunidades Europeias* ('Commission of the European Communities').

This table stores the *number of aligned segments* that contain:

a: both terms;
b: the Portuguese term but not the Greek term;
c: the Greek term but not the Portuguese term; and,
d: neither of those terms.

These amounts can be computed from the occurrences vectors as follows:

n: the size of the occurrences vectors;
a: the number of '1's in the co-occurrence vector;
b: the number of '1's found in the Portuguese term occurrences vector minus *a*;
c: the number of '1's found in the Greek term occurrences vector minus *a*; and,
d: $n - a - b - c$.

The difference between the total number of occurrences of both words may result either from different translations made by the translators themselves and / or from some occasional misalignment. Different translations may be due to syntactic constraints or to alternative translations the human translator decided to make.

Several measures of similarity have been proposed to use the information in the contingency tables in order to analyse the similarity of words and identify Translation Equivalents. We have used the Average Mutual Information as this similarity measure has proven to be appropriate for the task of identifying Translation Equivalents. The Average Mutual Information is computed as follows:

$$I(X;Y) = \sum_{x=\{0,1\}} \sum_{y=\{0,1\}} p(X=x, Y=y) \log_2 \left(\frac{p(X=x, Y=y)}{p(X=x)p(Y=y)} \right)$$

where X and Y are the two terms to be tested as translations. This formula is in contrast with the Specific Mutual Information, which is quite sensitive to rare co-occurrences, and which corresponds only to the last term of the sum. In this formula, $p(x = 1, y = 0)$ is the probability that term X occurs but term Y does not. Figure 2 shows some Translation Equivalents extracted in English, Greek and Portuguese.

Since we have used a general purpose terminology extractor, it extracts not only domain specific terms but also general language patterns. This happens because it tends to capture typical sequences of tokens independently of whether they are domain specific or not. The extractor was not developed to identify domain specific terminology though it is able to extract it too. We believe that by clustering documents and feeding those clusters of documents independently to the extractor it will be possible to distinguish domain specific terms from general language patterns. da Silva *et al.* (2001) proposes an unsupervised and language independent method to cluster documents to be used for this task.

Finally, by *re-feeding* the extracted Translation Equivalents back into the aligner it is possible to increase the number of potential anchors and, consequently, the number of new lexical cues available for the generation of correspondence points. The more correspondence points, the more fine-grained the alignment can be and the better the extracted equivalents can be. This means that alignment precision may improve. This is especially important for pairs of languages which share few lexical cues which can be used for alignment (like Portuguese and Chinese, as an extreme case).

4. Mining Multilingual Documents

Mining multilingual documents is a generalisation of the problem of mining documents that contain expressions which do not match exactly the ones in the query text. Fluhr (1995) made a survey of several approaches used for Multilingual Information Retrieval.

One traditional approach consists of using a *controlled vocabulary* both to index and retrieve documents, like the one used by Reuters or the Eurovoc (1995). Each document is indexed with a set of descriptors and queries are performed using this set of keywords. Queries are reformulated into the other languages by looking up the translations of the descriptors in a multilingual databank which contains the translations of each descriptor in the other languages.

English	Greek	Portuguese
JUDGMENT OF THE COURT	ΑΠΟΦΑΣΗ ΤΟΥ ΔΙΚΑΣΤΗΡΙΟΥ	ACÓRDÃO DO TRIBUNAL DE JUSTIÇA
Advocate General	γενικός εισαγγελέας	advogado – geral
Language of the case	Γλώσσα διαδικασίας	Língua do processo
Commission of the European Communities	Επιτροπή των Ευρωπαϊκών Κοινοτήτων	Comissão das Comunidades Europeias
Member States	κρατών μελών	Estados – membros
Act of Accession	Πράξεως Προσχωρήσεως	Acto de adesão
President of the Chamber	πρόεδρο τμήματος	presidente de secção
First Chamber	πρώτο τμήμα	Primeira Secção

Figure 2: A sample of Translation Equivalents obtained from the aligned texts in English, Greek and Portuguese.

However, the problem with this approach is that it limits queries to using the set of descriptors available instead of using full text words. An alternative method builds a matrix which links full text words to the set of controlled descriptors. This matrix can be built either manually or automatically by *learning* from previously indexed texts – a *text categorisation* task (Yang 1999). Once a query is posted, this matrix is looked up in order to find which descriptors are more associated with the words in the query. Finally, the translations of the descriptors are looked up in the multilingual databank in order to reformulate the query in the other languages.

Nevertheless, the use of controlled languages means that queries are somehow limited to the set of descriptors available. Alternative approaches can either translate the query – *query reformulation through translation* – or even the whole set of documents. Although there is some debate on the benefits and disadvantages of each one, reformulating the query through translation seems to be the simplest strategy since the latter option requires translating each document into all the other languages, which does not scale up well. Still, a query translated with errors may yield disappointing results if it has unresolved lexical ambiguities.

Anyhow, should parallel texts be available, they can become quite helpful. Some approaches exploit this fact by retrieving not only the documents most similar to the query posted but also their parallel versions. Then, the parallel texts can be used as a secondary query to retrieve similar untranslated documents in the other languages and even more parallel documents in the original query language should they be available.

In order for the search engine to retrieve the documents most similar to a query, several approaches have been suggested though most of them are based on the Vector Space model (Salton and McGill, 1983). In this model, documents are represented in a n-dimensional space, where n is the number of different words found in the texts – the term-document matrix. Both queries and documents are represented with n-dimensional vectors of term weights. Usually, terms are weighted using TF×IDF, the term frequency × the inverse document frequency of a term, i.e. the inverse of the number of documents in which the term occurs. Then, a document is considered relevant for a query if the query and the document vectors are *similar*. The similarity of two vectors can be computed using the cosine measure.[4]

In contrast with the previous model, the Generalised Vector Space Model (Wong *et al.*, 1985) takes into account the fact that terms are correlated. The assumption of this model is that two words are semantically similar if they tend to occur in the same documents, i.e. have similar document vectors in the term-document matrix. This Generalised model bears this in mind.

In the Pseudo-Relevance Feedback model, the initial query is expanded by adding to it terms found in the first set of retrieved documents, assuming that the top ranking documents are indeed relevant. This new extended query is posted again to the search engine in order to retrieve more documents (Salton and Buckley, 1990). Should there be parallel versions available for these documents, they can be used instead. This is the extension of the monolingual Pseudo-Relevance Feedback approach to multilingual retrieval suggested by Carbonell *et al.* (1997).

The Latent Semantic Indexing model (Deerwester *et al.*, 1990) is the next step after the previous model. It is also sensitive to co-occurrences of terms in the same document when it computes the similarity between the query and each document. The whole set of documents is *reduced* so that a smaller set is more representative for the content of the documents. This model was adapted to multilingual retrieval by Dumais *et al.* (1996), using parallel texts for training.

As for the approaches which expand the query and reformulate it with a translation, Machine Translation systems would probably be a good option and be able to provide good translations if queries were usually formulated as sentences or paragraphs. However, queries tend to be short and users tend to give isolated words for which Machine Translation systems performance degrades. Some alternative strategies have been suggested:

- look up each query word in a bilingual dictionary and use all possible translations;
- use a sentence aligned corpus and expand the query using every sentence in which all the query words co-occur; and,
- use an aligned corpus to build a translation databank.

The work presented in this paper fits in this last alternative. In this case, it becomes important to have good multilingual terminology databanks; otherwise, the reformulation of the query through the translation may not be correct if terms are not properly identified and translated. Carbonell *et al.* (1997) made an evaluation of several multilingual retrieval methods and concluded that query expansion by translation using a corpus-based 'translation matrix' provided the best results even when compared with a general purpose dictionary. Another reason why this approach seems to be better than a Machine Translation system is that it is easier to build a databank of translations for a new language, given parallel texts are available, than it is to build a Machine Translation system for the new language.

Thus, rather than translating each of the query words individually and providing all their possible translations, or using all sentences in which the words occur, the multilingual databank provides a simple means to make accurate translations of terms and, consequently, reduce their ambiguity. Furthermore, there are times when not even combining each of the possible word translations individually provides a possible compound translation, like *border crossing point* and *poste frontière* ('border post') in French, *hang gliding* and *asa delta* ('delta wing') in Portuguese, or even the common English phrasal verbs like *put up with* whose word for word translation are hardly combinable for other languages.

Thus, once a query is posted to a search engine, the multilingual databank of terminology can be used to translate the query terms into the available languages and posting subsequently monolingual searches in order to find relevant documents in the other languages. Had each of the query words been translated word for word, all alternative translations would have to be used which may lead to a long list of possible translations combinations. This increases the search space as more documents are bound to contain each of the words individually rather

[4] For a simple introduction, see, for example, Manning and Schütze (1999).

than the full correct translation. Also, a word for word translation may lead to no valid translation at all as shown with the examples above.

5. Conclusions

Currently, web search engines hardly support multilingual retrieval. Should a query be made in a language for which no relevant document exists, it will be unsuccessful. In the near future, it should be possible to access information independently of the language of the user and independently of the language in which the source text is written. This is what multilingual retrieval promises.

In this paper, we have made a small contribution to it. We have focussed on building multilingual terminology databanks from aligned texts in order to use them for multilingual retrieval. Compound words are particularly important in technical fields where their translation cannot be usually done word for word.

This paper has presented a methodology to extract terminology Translation Equivalents from aligned parallel texts so as to add a multilingual layer to search engines and allow multilingual searches by query expansion through translation to the other languages. In particular, this paper has described language independent and statistically supported methodologies to align parallel texts, extract multiword terms and find translation equivalents in the aligned texts. None of the techniques used assumes any language specific knowledge nor requires human hand coding of linguistic information.

We believe that by providing a multilingual terminology databank to multilingual search engines, it becomes possible to make reliable multilingual searches of compound terms as attested by Carbonell *et al.* (1997). Instead of building a databank of word translations in several languages, this work reports on the generation of a multilingual databank of multiword units from parallel texts. This makes translation of queries which contain compound terms less liable to errors. Also it reduces the search space of documents since terms can be identified in the query and translated as a unit instead of translating each of the words individually and retrieving documents which contain any of the possible translations.

6. Future Work

We need to make comparative evaluations on the retrieval performance on multilingual retrieval systems enhanced by the multilingual databank extracted from the parallel texts. It would also be interesting to check whether sub-sentential aligned text segments might be of some help for the translation of queries when the translation databank is not able to provide a translation. This strategy would simultaneously combine a query expansion approach based on a databank of translations with a query expansion approach based on aligned sub-sentential text segments.

As for the terminology extractor, its input needs to be normalised in order to avoid word variants. This will improve the accuracy of the translation equivalents extracted. In addition, it will increase the number of extracted translations by eliminating the sparse data problem due to alternative word variants. There are problems with highly inflectional languages like Greek, Finnish or even Portuguese. For example, the English adjective *public* can be translated into Portuguese as *público*, *pública*, *públicos*, or *públicas*, depending on the gender and number of the noun it qualifies. As a result, Translation Equivalents of terms which suffer variants tend to have low scores. We believe we can accomplish this task, also using language independent methods, by extracting typical sequences of characters using a methodology similar to the one used to extract typical sequences of characters in the alignment algorithm reported in Ribeiro *et al.* (2001b). This would identify common sequences of characters for inflected words though it may become harder for words which suffer radical changes when inflected. The verb *to be* is an English extreme case where a single word has eight variants: *be* / *being* / *am* / *are* / *is* / *was* / *were* / *been*.

Last but not the least, we need to distinguish domain specific terms from general language patterns. For this, we will use a method proposed by da Silva *et al.* (2001) to cluster documents according to domain specificity.

7. Acknowledgements

This work was partly funded by project Tradaut-pt.

8. References

Brown, P., Lai, J. and Mercer, R. (1991). Aligning Sentences in Parallel Corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics* (pp. 169–176). Berkeley, California, USA.

Brown, R., Carbonell, J. and Yang, Y. (2000). Automatic Dictionary Extraction for Cross-Language Information Retrieval. In J. Véronis (ed.), *Parallel Text Processing: Alignment and Use of Translation Corpora* (pp. 275–298). Dordrecht, The Netherlands: Kluwer Academic Publisher.

Carbonell, J., Yang, Y., Frederking, R., Brown, R. , Geng, Y. and Lee, D. (1997). Translingual Information Retrieval: A Comparative Evaluation. In *Proceedings of the Fifteenth International Joint Conference on Artificial Intelligence – IJCAI 97* (pp. 708–715). Volume I. Nagoya, Japan.

Church, K. (1993). Char_align: A Program for Aligning Parallel Texts at the Character Level. In *Proceedings of the 31st Annual Meeting of the Association for Computational Linguistics* (pp. 1–8). Columbus, Ohio, USA.

Daille, B. (1995). Combined Approach for Terminology Extraction: Lexical Statistics and Linguistic Filtering. In *UCREL (University Centre for Computer Corpus Research on Language) Technical Papers*. 5. Lancaster, United Kingdom: University of Lancaster, Department of Linguistics.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T. and Harshman, R. (1990). Indexing by Latent Semantic Analysis. *Journal of the American Society of Information Science*, 41 (6), 391–498.

Dumais, S., Landauer T. and Littman, M. (1996). Automatic Cross-Linguistic Information Retrieval Using Latent Semantic Indexing. In *Proceedings of the 19th Annual International ACM (Association for Computing Machinery) – SIGIR (Special Interest Group in Information Retrieval) Conference on Research and Development in Information Retrieval – SIGIR'96* (pp. 16–23). Zurich, Switzerland.

ELRA – European Language Resources Association (1997). *Multilingual Corpora for Co-operation*, Disk 2 of 2. Paris, France, 454 MBytes.

Eurovoc (1995). *Thesaurus Eurovoc – Volume 2: Subject-Oriented Version*. Annex to the index of the Official Journal of the EC. Luxembourg, Office for Official Publications of the European Communities. Retrieved from http://europa.eu.int/celex/eurovoc on Mon, 2002 Apr 15.

Fluhr, C. (1995). Multilingual Information Retrieval. In R. Cole, J. Mariani, H. Uszkoreit, A. Zaenen and V. Zue (eds.), *Survey of the State of the Art in Human Language Technology*. Retrieved from http://cslu.cse.ogi.edu/HLTsurvey/Ch8Node7.html#Section85 on Mon, 2002 Apr 15.

Fung, P. and Church, K. (1994). K-vec: A New Approach for Aligning Parallel Texts". In *Proceedings of the 15th International Conference on Computational Linguistics – Coling'94* (pp. 1096–1102), Kyoto, Japan.

Fung, P. and McKeown, K. (1994). Aligning Noisy Parallel Corpora across Language Groups: Word Pair Feature Matching by Dynamic Time Warping. In *Technology Partnerships for Crossing the Language Barrier: Proceedings of the First Conference of the Association for Machine Translation in the Americas* (pp. 81–88). Columbia, Maryland, USA.

Fung, P. and McKeown, K. (1997). A Technical Word- and Term-Translation Aid Using Noisy Parallel Corpora across Language Groups. *Machine Translation*, 12 (1–2), 53–87. Dordrecht, The Netherlands: Kluwer Academic Publisher.

Gale, W. and Church, K. (1991). A Program for Aligning Sentences in Bilingual Corpora. In *Proceedings of the 29th Annual Meeting of the Association for Computational Linguistics* (pp. 177–184). Berkeley, California, USA (short version). Also (1993) *Computational Linguistics*, 19 (1), 75–102 (long version).

Google Inc. (2001a). *Search 3 Billion Documents Using Google*. Retrieved from http://www.google.com/3.html on Tue, 2001 December 11.

Google Inc. (2001c). *Google Zeitgeist – Search Patterns, Trends, and Surprises According to Google*. Google Press Centre: Zeitgeist, 2001 October issue. Retrieved from http://www.google.com/press/zeitgeist/zeitgeist-oct.html on Wed, 2002 January 9.

Google Inc._(2001b). *Google Zeitgeist – Search Patterns, Trends, and Surprises According to Google*. Google Press Centre: Zeitgeist, 2001 August issue. Retrieved from http://www.google.com/press/zeitgeist/zeitgeist-aug.html on Wed, 2002 January 9.

Kay, M. and Röscheisen, M. (1993). Text-Translation Alignment. *Computational Linguistics*, 19 (1), 121–142.

Landauer, T. and Littman, M. (1990). Fully-Automatic Cross-Language Document Retrieval Using Latent Semantic Indexing. In *Proceedings of the 6th Annual Conference of the University of Waterloo Centre for the New Oxford English Dictionary* (pp. 40–62). Waterloo, Canada.

Manning, C. and Schütze, H. (1999). *Foundations of Statistical Natural Language Processing* (680 p.). 4th edition, Cambridge, Massachusetts, USA: The MIT Press.

Melamed, I. (1999). Bitext Maps and Alignment via Pattern Recognition. *Computational Linguistics*, 25 (1), 107–130.

Ribeiro, A., Lopes, G. and Mexia, J. (2000). Using Confidence Bands for Parallel Texts Alignment. In *Proceedings of the 38th Conference of the Association for Computational Linguistics* (pp. 432–439). Hong Kong, China.

Ribeiro, A., Lopes, G. and Mexia, J. (2001a). Extracting Translation Equivalents from Portuguese-Chinese Parallel Texts. *Journal of Studies in Lexicography*, 11 (1), 118–194. Seoul, South Korea: Yonsei University.

Ribeiro, A., Dias, G., Lopes, G. and Mexia, J. (2001b). Cognates Alignment. In B. Maegaard (ed.), *Proceedings of the Machine Translation Summit VIII – MT Summit VIII – Machine Translation in the Information Age* (pp. 287–292). Santiago de Compostela, Spain.

Richards, J., Platt, J. and Weber, H. (1985). *Longman Dictionary of Applied Linguistics*. London, United Kingdom: Longman.

Salton, G. and McGill, M. (1983). *Introduction to Modern Information Retrieval*. New York, USA: McGraw-Hill (448 p.).

Salton G. and Buckley, C. (1990). Improving Retrieval Performance by Relevance Feedback. *Journal of the American Society for Information Science*, 41 (4), 182–188.

da Silva, J., Dias, G., Guilloré, S. and Lopes, J. (1999). Using Localmaxs Algorithms for the Extraction of Contiguous and Non-contiguous Multiword Lexical Units. In P. Barahona and J. Alferes (eds.), *Progress in Artificial Intelligence – Lecture Notes in Artificial Intelligence*, 1695 (pp. 113–132). Berlin, Germany: Springer-Verlag.

da Silva, J., Mexia, J., Coelho, C. and Lopes, J. (2001). Document Clustering and Cluster Topic Extraction in Multilingual Corpora. In N. Cercone, T. Lin and X. Wu (eds.), *Proceedings of the Institute of Electrical and Electronics Engineers (IEEE) 2001 International Conference on Data Mining – ICDM'01* (pp. 513–520). San Jose, California, USA: IEEE Computer Society Press.

Simard, M. and Plamondon, P. (1998). Bilingual Sentence Alignment: Balancing Robustness and Accuracy. *Machine Translation*, 13 (1), 59–80. Dordrecht, The Netherlands: Kluwer Academic Publisher.

Simard, M., Foster, G. and Isabelle, P. (1992). Using Cognates to Align Sentences in Bilingual Corpora. In *Proceedings of the Fourth International Conference on Theoretical and Methodological Issues in Machine Translation TMI-92* (pp. 67–81). Montréal, Canada.

Wu, D. (1994). Aligning a Parallel English–Chinese Corpus Statistically with Lexical Criteria. In *Proceedings of the 32nd Annual Conference of the Association for Computational Linguistics* (pp. 80–87), Las Cruces, New Mexico, USA.

Wong, S., Ziarko, W. and Wong, P. (1985). Generalized Vector Space Model in Information Retrieval. In *Proceedings of the 8th Annual International ACM (Association for Computing Machinery) – SIGIR (Special Interest Group in Information Retrieval) Conference on Research and Development in Information Retrieval – SIGIR'85* (pp. 18–25). New York, USA.

Yang, Y. (1999). An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval*, 1, 69–90.

Grammar Learning by Partition Search

Anja Belz

ITRI
University of Brighton
Lewes Road
Brighton BN2 4GJ, UK
Anja.Belz@itri.brighton.ac.uk

Abstract

This paper describes Grammar Learning by Partition Search, a general method for automatically constructing grammars for a range of parsing tasks. Given a base grammar, a training corpus, and a parsing task, Partition Search constructs an optimised probabilistic context-free grammar by searching a space of nonterminal set partitions, looking for a partition that maximises parsing performance and minimises grammar size. The method can be used to optimise grammars in terms of size and performance, or to adapt existing grammars to new parsing tasks and new domains. This paper reports an example application to optimising a base grammar extracted from the Wall Street Journal Corpus. Partition Search improves parsing performance by up to 5.29%, and reduces grammar size by up to 16.89%. Parsing results are better than in existing treebank grammar research, and compared to other grammar compression methods, Partition Search has the advantage of achieving compression without loss of grammar coverage.

1. Introduction

Grammar Learning by Partition Search is a computational learning method that constructs probabilistic grammars optimised for a given domain or parsing task. The main idea behind this method is that new grammars can be derived from existing ones by simple operations on nonterminal sets. Automatically carrying out different combinations of such operations and testing the derived grammars' size and performance makes it possible to automatically optimise the grammars.

The main practical applications of Grammar Learning by Partition Search are the optimisation of an existing grammar's size and performance, and the adaptation of existing grammars to new tasks. Results for optimising a base grammar extracted from the Wall Street Journal Corpus (WSJC) are reported here, while results for adapting the same base grammar to different noun phrase extraction tasks are reported elsewhere (Belz, 2002).

This paper is organised in two main sections. Section 2. describes Grammar Learning by Partition Search. Section 3. reports experiments and results for NP identification and NP chunking.

2. Learning PCFGs by Partition Search

Partition Search Grammar Learning starts from the idea that new context-free grammars (CFGs) can be created from old simply by modifying the nonterminal sets, *merging* and *splitting* subsets of nonterminals. For example, for certain parsing tasks it is useful to *split* a single verb phrase category into verb phrases that are headed by a modal verb and those that are not, whereas for other parsing tasks, the added grammar complexity is avoidable. In another context, it may not be necessary to distinguish noun phrases in subject position from first objects and second objects, making it possible to *merge* the three categories into one.

The usefulness of such split and merge operations can be measured by their effect on a grammar's size (number of rules and nonterminals) and performance (parsing accuracy on a given task). Grammar Learning by Partition Search automatically tries out different combinations of merge and split operations and therefore can automatically optimise a grammar's size and performance on a given task.

2.1. Preliminary definitions

Definition 1 Set Partition

A partition of a nonempty set A is a subset Π of 2^A such that $\emptyset$ is not an element of Π and each element of A is in one and only one set in Π.

The partition of A where all elements are singleton sets is called the *trivial partition* of A.

Definition 2 Probabilistic Context-Free Grammar[1]

A Probabilistic Context-Free Grammar (PCFG) is a 4-tuple (W, N, N_S, R), where W is a set of terminal symbols, N is a set of nonterminal symbols, $N_S \in N$ is a start symbol, and $R = \{(r_1, p(r_1)), \ldots (r_m, p(r_m))\}$ is a set of rules with associated probabilities. Each rule r_i is of the form $n \rightarrow \alpha$, where n is a nonterminal, and α is a string of terminals and nonterminals. For each nonterminal n, the values of all $p(n \rightarrow \alpha_i)$ sum to one, or: $\sum_{i:(n \rightarrow \alpha_i, p(n \rightarrow \alpha_i) \in R} p(n \rightarrow \alpha_i) = 1$.

2.2. Generalising and Specialising PCFGs through Nonterminal Set Operations

2.2.1. Nonterminal merging

Consider two PCFGs G and G':

[1] This definition is for PCFGs with a single start symbol, to simplify the definition of PCFG Partitioning below.

$G = (W, N, N_S, R)$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N =$ { S, NP-SUBJ, VP, NP-OBJ }
$N_S =$ S
$R =$ { (S -> NP-SUBJ VP, 1),
(NP-SUBJ -> NNS, 0.5),
(NP-SUBJ -> DET NN, 0.5),
(VP -> VBD NP-OBJ, 1),
(NP-OBJ -> NNS, 0.75),
(NP-OBJ -> DET JJ NNS, 0.25) }

$G' = (W, N', N_S, R')$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N' =$ { S, NP, VP }
$N_S =$ S
$R' =$ { (S -> NP VP, 1),
(NP -> NNS, 0.625),
(NP -> DET NN, 0.25),
(VP -> VBD NP, 1),
(NP -> DET JJ NNS, 0.125) }

Intuitively, to derive G' from G, the two nonterminals NP-SUBJ and NP-OBJ are merged into a single new nonterminal NP. This merge results in two rules from R becoming identical in R': both NP-SUBJ -> NNS and NP-OBJ -> NNS become NP -> NNS. One way of determining the probability of the new rule NP -> NNS is to sum the probabilities of the old rules and renormalise by the number of nonterminals that are being merged[2]. In the above example therefore $p(\texttt{NP -> NNS}) = (0.5 + 0.75)/2 = 0.625$[3].

An alternative would be to reestimate the new grammar on some corpus, but this is not appropriate in the current context: merge operations are used in a search process (see below), and it would be expensive to reestimate each new candidate grammar derived by a merge. It is better to use any available training data to estimate the original grammar's probabilities, then the probabilities of all derived grammars can simply be calculated as described above without expensive corpus reestimation.

The new grammar G' derived from an old grammar G by merging nonterminals in G is a generalisation of G: the language of G', or $L(G')$, is a superset of the language of G, or $L(G)$. E.g., in the above example, det jj nns vbd det jj nns is in $L(G')$ but not in $L(G)$. For any sentence $s \in L(G)$, the parses assigned to s by G' form a superset of the set of parses assigned to s by G. The probabilities of parses for s can change, and so can the probability ranking of the parses, i.e. the most likely parse for s under G may be different from the most likely parse for s under G'. Finally, G' has the same number of rules as G or fewer.

2.2.2. Nonterminal splitting

Deriving a new PCFG from an old one by splitting nonterminals in the old PCFG is not quite the exact reverse of deriving a new PCFG by merging nonterminals. The difference lies in determining probabilities for new rules. Consider the following grammars G and G':

$G = (W, N, N_S, R)$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N =$ { S, NP, VP }
$N_S =$ S
$R =$ { (S -> NP VP, 1),
(NP -> NNS, 0.625),
(NP -> DET NN, 0.25),
(VP -> VBD NP, 1),
(NP -> DET JJ NNS, 0.125) }

$G' = (W, N', N_S, R')$,
$W =$ { NNS, DET, NN, VBD, JJ }
$N' =$ { S, NP-SUBJ, VP, NP-OBJ }
$N_S =$ S
$R' =$ { (S -> NP-SUBJ VP, ?),
(S -> NP-OBJ VP, ?),
(NP-SUBJ -> NNS, ?),
(NP-SUBJ -> DET NN, ?),
(NP-SUBJ -> DET JJ NNS, ?) }
(VP -> VBD NP-SUBJ, ?),
(VP -> VBD NP-OBJ, ?),
(NP-OBJ -> NNS, ?),
(NP-OBJ -> DET NN, ?),
(NP-OBJ -> DET JJ NNS, ?) }

To derive G' from G, the single nonterminal NP is split into two nonterminals NP-SUBJ and NP-OBJ. This split results in several new rules. For example, for the old rule NP -> NNS, there now are two new rules NP-SUBJ -> NNS and NP-OBJ -> NNS. One possibility for determining the new rule probabilities is to redistribute the old probability mass evenly among them, i.e. $p(\texttt{NP -> NNS}) = p(\texttt{NP-SUBJ -> NNS}) = p(\texttt{NP-SUBJ -> NNS})$. However, then there would be no benefit at all from performing such a split: the resulting grammar would be larger, the most likely parses remain unchanged, and for each parse p under G that contains a nonterminal NT participating in a split, there would be at least two equally likely parses under G'.

The new probabilities cannot be calculated directly from G. The redistribution of the probability mass has to be motivated from a knowledge source outside of G. One way to proceed is to estimate the new rule probabilities on the original corpus — provided that it contains the information on the basis of which a split operation was performed in extractable form. For the current example, a corpus in which objects and subjects are annotated could be used to estimate the probabilities of the rules in G', and might yield the following rule set R' (which reflects the fact that in English, the NP in a sentence NP VP is (usually) a subject, whereas the NP in a VP consisting of a verb followed by an NP is an object):

$R' =$ { (S -> NP-SUBJ VP, 1),
(S -> NP-OBJ VP, 0),
(NP-SUBJ -> NNS, 0.5),
(NP-SUBJ -> DET NN, 0.5),
(NP-SUBJ -> DET JJ NNS, 0) }
(VP -> VBD NP-SUBJ, 0),
(VP -> VBD NP-OBJ, 1),
(NP-OBJ -> NNS, 0.75),
(NP-OBJ -> DET NN, 0),
(NP-OBJ -> DET JJ NNS, 0.25)

[2] Reestimating the probabilities on the training corpus would of course produce identical results.

[3] Renormalisation is necessary because the probabilities of all rules expanding the same nonterminal sum to one, therefore the probabilities of all rules expanding a new nonterminal resulting from merging n old nonterminals will sum to n.

Definition 3 PCFG Partitioning

Given a PCFG $G = (W, N, N_S, R)$ and a partition Π_N of the set of nonterminals N, the PCFG derived by partitioning G with Π_N is $G' = (W, \Pi_N, N_S, R')$, where:

$$R' = \Big\{ (a_1 \rightarrow a_2 \ldots a_n,\ p) \ \Big|\ \big\{(b_1^1 \rightarrow b_2^1 \ldots b_n^1,\ p^1),\ \ldots (b_1^m \rightarrow b_2^m \ldots b_n^m,\ p^m)\big\} \in \Omega,\ a_1 \in \Pi^N, b_i^j \in a_1,\ \forall i, 2 \leq i \leq n\ \big(\text{ either } a_i = b_i^j \in W, \text{ or } a_i \in \Pi^N, b_i^j \in a_i\ \big),\ p = \big(\textstyle\sum_{j=1}^m p^j\big)/|a_1|\ \Big\}, \text{ and}$$

Ω is a partition of R such that each $O \in \Omega$ contains all and only elements from R $(b_1^1 \rightarrow b_2^1 \ldots b_n^1,\ p^1),\ \ldots (b_1^m \rightarrow b_2^m \ldots b_n^m,\ p^m)$ for which the following holds: $\forall i, 1 \leq i \leq n\ \big(\text{ either } b_i^1 = b_i^2 = \ldots = b_i^m \in W, \text{ or } \{b_i^1, b_i^2, \ldots b_i^m\} \subseteq P,\ P \in \Pi_N\ \big)$.

With rules of zero probability removed, G' is now identical to the original grammar G in the example in the previous section.

2.3. Partition Search

A PCFG together with merge and split operations on the nonterminal set defines a space of derived grammars which can be searched for a new PCFG that optimises some given objective function. The disadvantage of this search space is that it is infinite, and each split operation requires the reestimation of rule probabilities from a training corpus, making it computationally much more expensive than a merge operation.

However, there is a simple way to make the search space finite, and at the same time to make split operations redundant. The resulting method, Grammar Learning by Partition Search, is described in this section. First, the merge operation that was informally introduced in the last section is generalised and defined formally. Next, search space, search task and objective function are discussed, and finally, the search algorithm is presented.

2.3.1. PCFG Partitioning

An arbitrary number of merges can be represented by a partition of the set of nonterminals. For the example presented in Section 2.2.1. above, the partition of the nonterminal set N in G that corresponds to the nonterminal set N' in G' is { {S}, {NP-SBJ, NP-OBJ}, {VP} }. The original grammar G together with a partition of its nonterminal set fully specifies the new grammar G': the new rules and probabilities, and the entire new grammar G' can be derived from the partition together with the original grammar G. The process of obtaining a new grammar G', given a base grammar G and a partition of the nonterminal set N of G will be called PCFG Partitioning[4].

In the examples in Sections 2.2.1. and 2.2.2., a notational convention was tacitly adopted which is also used in the formal definition of PCFG Partitioning (Definition 3). As a result of merging, NP-SUBJ and NP-OBJ become a single new nonterminal. This new nonterminal was represented above by the set of merged nonterminals {NP-SBJ, NP-OBJ} (in the partition), as well as by a new symbol string NP (in the definition of G'). The two representations are treated as interchangeable: the new nonterminal is represented either as a set or a nonterminal symbol[5].

The definition of PCFG Partitioning can be paraphrased as follows[6]. Given a PCFG $G = (W, N, N_S, R)$ and a partition Π_N of the set of nonterminals N, the PCFG derived from G by Π_N is $G' = (W, \Pi_N, N_S, R')$. That is, the set of terminals remains the same, and the new set of nonterminals is just the partition Π_N[7]. Ω is the partition of R in which all production rules are grouped together that become identical as a result of the nonterminal merges specified by Π_N. Then, the new set of probabilistic rules R' contains one element $(a_1 \rightarrow a_2 \ldots a_n,\ p)$ for each element $\big\{(b_1^1 \rightarrow b_2^1 \ldots b_n^1,\ p^1), \ldots (b_1^m \rightarrow b_2^m \ldots b_n^m,\ p^m)\big\}$ of Ω, such that the following holds between them: a_1 is a nonterminal from Π^N and contains all the b_1^j; for the other a_i, either a_i is a nonterminal and contains all the b_i^j, or it is a terminal and is identical to b_i^j. The new rule probability p is the sum of all probabilities $p^j, 1 \leq j \leq m$ renormalised by the size of the set a_1.

2.3.2. Search space

As stated previously, the search space for Grammar Learning by Partition Search can be made finite and search-

[4]The concept of context-free grammar partitioning in this paper is not directly related to that in (Korenjak, 1969; Weng and Stolcke, 1995), and later publications by Weng et al. In these previous approaches, a non-probabilistic CFG's *set of rules* is partitioned into subsets of rules. The partition is drawn along a specific nonterminal NT, which serves as an interface through which the subsets of rules (hence, subgrammars) can communicate after partition (one grammar calling the other). In the calling subgrammar, NT in RHSs is prefixed vt to denote that it is a 'virtual terminal'. In the following example from (Luk et al., 2000, p. +2), partitioning grammar G along the nonterminal NP yields subgrammars G_S and G_{NP} as follows:

G:	G_S:	G_{NP}:
1. S → NP VP	INPUT = {vtNP}	INPUT = ∅
2. NP → n	OUTPUT = {S}	OUTPUT = {NP}
3. NP → det n	1. S → vtNP VP	1. NP → n
4. NP → NP PP	2. VP → v vtNP	2. NP → det n
5. PP → prep NP	3. NP → prep vtNP	3. NP → NP PP
6. VP → v NP		

[5]The name assigned to the new nonterminal in the current implementation of PCFG Partitioning is the longest common prefix of the old nonterminals that are merged followed by an indexation tag (to distinguish otherwise identical names).

[6]This definition is for PCFGs with one start symbol. In the current implementation of Partition Search, PCFGs are permitted to have multiple start symbols and these can be merged with other nonterminals. The probability of a new start symbol resulting from a merge is the renormalised sum of the probabilities of only the start symbols participating in the merge.

[7]Recall previous comments about this notational convention.

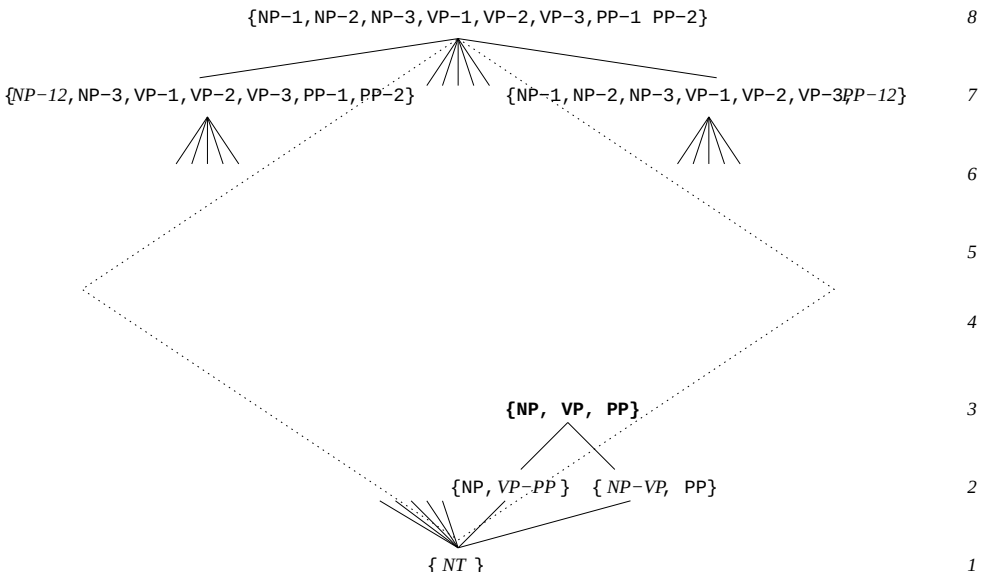


Figure 1: Simple example of a partition search space.

able entirely by merge operations (grammar partitions).

Making the search space finite: The number of merge operations that can be applied to a nonterminal set is finite, because after some finite number of merges there remains only one nonterminal. On the other hand, the number of split operations that can sensibly be applied to a nonterminal *NT* has a natural upper bound in the number of different terminal strings dominated by *NT* in a corpus of evidence (e.g. the corpus the PCFG was trained on). For example, when splitting the nonterminal NP into subjects and objects, there would be no point in creating more new nonterminals than the number of different subjects and objects found in the corpus.

Given these bounds, there is a finite number of distinct grammars derivable from the original grammar by different combinations of merge and split operations. This forms the basic space of candidate solutions for Grammar Learning by Partition Search.

Making the search space searchable by grammar partitioning only: Imposing an upper limit on the number and kind of split operations permitted not only makes the search space finite but also makes it possible to directly derive the *maximally split nonterminal set* (Max Set). Once the Max Set has been defined, the single grammar corresponding to it — the *maximally split Grammar* (Max Grammar) — can be derived and retrained on the training corpus[8].

The set of points in the search space corresponds to the set of partitions of the Max Set. Search for an optimal grammar can thus be carried out directly in the partition space of the Max Grammar.

Structuring the search space: The finite search space can be given hierarchical structure as shown in Figure 1 for an example of a very simple base nonterminal set {NP, VP, PP}, and a corpus which contains three different NPs, three different VPs and two different PPs.

At the top of the graph is the Max Set. The sets at the next level down (level 7) are created by merging pairs of nonterminals in the Max Set, and so on for subsequent levels. At the bottom is the *maximally merged nonterminal set* (Min Set) consisting of a single nonterminal *NT*. The sets at the level immediately above it can be created by splitting *NT* in different ways. The sets at level 2 are created from those at level 1 by splitting one of their elements. The original nonterminal set ends up somewhere in between the top and bottom (at level 3 in this example).

While this search space definition results in a finite search space and obviates the need for the expensive split operation, the space will still be vast for all but trivial corpora. In Section 3.3. below, alternative ways for defining the Max Set are described that result in much smaller search spaces.

2.3.3. Search task and evaluation function

The input to the Partition Search procedure consists of a base grammar G_0, a base training corpus C, and a task-specific training corpus D^T. G_0 and C are used to create the Max Grammar G. The **search task** can then be defined as follows:

> Given the maximally split PCFG $G = (W, N, N_S, R)$, a data set of sentences D, and a set of target parses D^T for D, find a partition Π_N of N that derives a grammar $G' = (W, \Pi_N, N_S, R')$, such that $|R'|$ is minimised, and $f(G', DD^T)$ is maximised, where f scores the performance of G' on D as compared to D^T.

The size of the nonterminal set and hence of the grammar decreases from the top to the bottom of the search space. Therefore, if the partition space is searched top-down, grammar size is minimised automatically and does not need to be assessed explicitly.

In the current implementation, the **evaluation function** f simply calculates the F-Score achieved by a candidate

[8] This can be done as follows: for each nonterminal *N*, count the number n of different terminal strings it dominates in the training corpus, and tag each occurrence of *NT* with a number *NT*-1, ... *NT*-n; duplicate the rules containing *NT* correspondingly, and calculate the rule probabilities.

grammar on D as compared to D^T. The F-Score is obtained by combining the standard PARSEVAL evaluation metrics *Precision* and *Recall*[9] as follows: $2 \times Precision \times Recall/(Precision + Recall)$.

An existing parser[10] was used to obtain Viterbi parses. If the parser failed to find a complete parse for a sentence, a simple grammar extension method was used to obtain partial parses instead (Schmid and Schulte im Walde (2000, p. 728) use an almost identical method).

2.3.4. Search algorithm

Since each point in the search space can be accessed directly by applying the corresponding nonterminal set partition to the Max Grammar, the search space can be searched in any direction by any search method using partitions to represent candidate grammars.

In the current implementation (see pseudo-code representation in Procedure 1), a variant of beam search is used to search the partition space top down. A list of the n current best candidate partitions is maintained (initialised to the Max Set). For each of the n current best partitions a random subset of size b of its children in the hierarchy is generated and evaluated. From the union of current best partitions and the newly generated candidate partitions, the n best elements are selected and form the new current best set. This process is iterated until either no new partitions can be generated that are better than their parents, or the lowest level of the partition tree is reached. In each iteration the size of the nonterminal set decreases by one.

The size of the search space grows exponentially with the size i of the Max Set. However, the complexity of the Partition Search algorithm is only $O(nbi)$, because only up to $n \times b$ partitions are evaluated in each of up to i iterations[11].

Procedure 1
P_SEARCH$\big((W, N, N_S, R), D, D^T, n, x, b\big)$

```
1:  Stop ← FALSE
2:  P_intermediate ← INITIALISE(N)
3:  EVALUATE(P_intermediate, G, D, D^T, x)
4:  while not Stop do
5:    P_current ← SELECT(P_intermediate)
6:    P_new ← GENERATE(P_current, b)
7:    if P_new = EMPTYLIST then
8:      Stop ← TRUE
9:    else
10:     P_intermediate ← APPEND(P_current, P_new)
11:     EVALUATE(P_intermediate, G, D, D^T, x)
12:   end if
13: end while
14: return P_current
15:
16: Subprocedure INITIALISE(N)
17: return set containing trivial partition of N
18:
19: Subprocedure SELECT(P_intermediate, n)
20: return n best elements from P_intermediate
21:
22: Subprocedure GENERATE(P_current, b)
23: Return_val ← EMPTYLIST
24: for all p ∈ P_current do
25:   List ← generate b random elements of
      {Π | Π is a partition of p and |Π| = |p| − 1}
26:   Return_val ← APPEND(Return_val, List)
27: end for
28: return Return_val
29:
30: Subprocedure EVALUATE(P_intermediate, G, D, D^T, x)
31: for all p ∈ P_intermediate do
32:   G' ← partition grammar G with p
33:   D^A ← parse data D with G'
34:   score of p is F-Score of D^A against D^T
35: end for
```

3. Grammar Optimisation with Partition Search

3.1. Testing and Training Data

Sections 15–18 of WSJC were used for deriving the base grammar and as the base training corpus, and different randomly selected subsets of Section 1 from the same corpus were used as task-specific training corpora during search. Section 20 was used for final performance tests.

The Brill Tagger was used for POS tagging testing data, and achieved an average accuracy of 97.5% (as evaluated by `evalb`).

3.2. Base grammar

A simple treebank grammar[12] was derived from Sections 15–18 of the WSJ corpus by the following procedure:

1. Iteratively edit the corpus by deleting (i) brackets and labels that correspond to empty category expansions; (ii) brackets and labels containing a single constituent that is not labelled with a POS-tag; (iii) cross-indexation tags; (iv) brackets that become empty through a deletion.
2. Convert each remaining bracketting in the corpus into the corresponding production rule.
3. Collect sets of terminals W, nonterminals N and start symbols N_S from the corpus. Probabilities p for rules $n \to \alpha$ are calculated from the rule frequencies C by Maximum Likelihood Estimation: $p(n \to \alpha) = \frac{C(n \to \alpha)}{\sum_i C(n \to \alpha^i)}$.

This procedure creates the base grammar *BARE* which has 10,118 rules and 147 nonterminals.

3.3. Restricting the search space further

The simple method described in Section 2.3.2. for defining the maximally split nonterminal set (Max Set) tends to result in vast search spaces. Using parent node (PN) information to create the Max Set is much more restrictive and linguistically motivated. The Max Grammar *PN* used in the experiments reported below can be seen as making use of *Local Structural Context* (Belz, 2001): the independence assumptions inherent in PCFGs are weakened by making

[9] I used the `evalb` program by Sekine and Collins (`http://cs.nyu.edu/cs/projects/proteus/evalb/`) to obtain Precision and Recall figures.

[10] LoPar (Schmid, 2000) in its non-head-lexicalised mode. Available from `http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/LoPar-en.html`.

[11] As before, n is the number of current best candidate solutions, b is the width of the beam, and i is the size of the Max Set.

[12] The term was coined by Charniak (1996).

the rules' expansion probabilities dependent on part of their immediate structural context (here, its parent node). To obtain the grammar *PN*, the base grammar's nonterminal set is maximally split on the basis of the *parent node* under which rules are found in the base training corpus[13]. Several previous investigations have demonstrated improvement in parsing results due to the inclusion of parent node information (Charniak and Carroll, 1994; Johnson, 1998b; Verdú-Mas et al., 2000).

Another possibility is to use the base grammar *BARE* itself as the Max Grammar. This is a very restrictive search space definition and amounts to an attempt to optimise the base grammar in terms of its size and its performance on a given task without adding any information. Results are given below for both *BARE* and *PN* as Max Grammars.

In the current implementation of the algorithm, the search space is reduced further by avoiding duplicate partitions, and by only allowing merges of nonterminals that have the same phrase prefix `NP-*, VP-*` etc.

The Max Grammars end up having sets of nonterminals that differ from the bracket labels used in the WSJC: while the phrase categories (e.g. `NP`) are the same, the tags (e.g. `*-S, *-3`) on the phrase category labels may differ. In the evaluation, all labels starting with the same phrase category prefix are considered equivalent.

3.4. Optimising the WSJ treebank grammar

Base grammar *BARE* achieves an F-Score of 74.09 on the full parsing task. Since the base grammar is just the treebank grammar extracted from the WSJC, and the task it is optimised for is full parsing, the most directly related research is other treebank grammar research. Over the last few years, a range of research projects — e.g. Charniak (1996), Cardie & Pierce (1998), Johnson (1998a, 2000), Krotov et al. (2000) — have looked at probabilistic grammars that have been directly derived from bracketted corpora.

Because the number of rules in treebank grammars is very large at least in the case of the WSJC, and because their parsing performance moreover tends to be not very good, some techniques are usually applied to reduce grammar size and to improve performance. All approaches edit the corpus in some way, e.g. eliminating single child rules, empty category rules, functional tags, co-indexation tags, and punctuation marks. Different compression methods (such as eliminating rules with frequency less than some n) have been investigated that reduce the size of grammars without too much loss of performance (in particular by Charniak and Krotov et al.). To improve parsing performance, e.g. Charniak relabels auxiliary verbs with a separate POS-tag and incorporates a "right-branching correction" into the parser to make it prefer right-branching structures.

Several other grammar building and training methods are similar to treebank grammar construction: Bod & Scha's Data-Oriented Parsing method which extracts tree fragments rather than rules from corpora, and Memory-Based Learning methods (Daelemans et al.) for building parsing systems from corpora.

Table 1 shows results achieved by Partition Search with grammars *BARE* and *PN* as Max Grammars. The first column shows the Max Grammar used in a given batch of experiments. The second column indicates the type of result, where the Max Grammar result is the F-Score, grammar size and number of nonterminals of the Max Grammar itself, and the remaining results are the average and single best results achieved by Partition Search. The third and fourth columns show the number of iterations and evaluations carried out before search stopped. Columns 5–8 show details of the final solution grammars: column 5 shows the evaluation score on the training data, column 6 the overall F-Score on the testing data, column 7 the size, and the last column gives the number of nonterminals.

The best size reduction result was 35 nonterminals and 8,409 rules with an increase in the F-Score to 74.54 (as compared to the baseline of 74.09). The best performance increase result was an F-Score of 78.01 (compared to the baseline of 74.09), accompanied by an increase in the number of rules to 15,608.

The best performance result reported here is better than the best results reported by Charniak (1996) and Krotov et al. (2000), even though the previous results were obtained after using ca. 10/11 of the WSJ corpus as a training set compared to 3/25 used here (UF = unlabelled F-Score, LF = labelled F-Score):

	UF	LF
Krotov et al. (2000)	79.12	76.09
Charniak (1996)	79.59	–
Optimised *PN*-Grammar	80.74	78.01

During Partition Search, only those nonterminal subsets are merged that result in an improved or unchanged F-Score. In the case of the grammar *BARE* as the Max Grammar, merging means eliminating phrase subcategories reflecting grammatical function. As can be seen from the results in Table 1, a lot of grammatical function information can be eliminated without affecting parsing results: of the 147 original phrase subcategories, only just over a third remain on average. In the case of the *PN* Max Grammar, search found it a lot harder to eliminate parent node subcategories without worsening parsing performance: just over half of the 970 parent node subcategories remain on average. The biggest part of the improvement in parsing performance is due to way the Max Grammar is defined, i.e. to the addition of parent node information (from 74.09 to 77.8 F-Score).

3.5. General comments

Partition Search is able to reduce grammar size by merging groups of nonterminals (hence groups of rules) that do not need to be distinguished for a given task. It is able to improve parsing performance firstly by grammar generalisation (a partitioned grammar parses a superset of the sentences parsed by the base grammar), and secondly by reranking parse probabilities (the most likely parse for a sentence under a partitioned grammar can differ from its most likely parse under the base grammar).

[13] The parent node of a phrase is the category of the phrase that immediately contains it.

Max Grammar		Iter.	Eval.	F-Score (subset)	F-Score (WSJC S 1)	Size (rules)	Nonterms
BARE	Max Grammar result:				74.09	10,118	147
	Average:	90.2	2,000.8	83.43	74.05	9,288.4	59.8
	Best (size and F-score):	114	2,686	86.04	74.54	8,409	35
PN	Max Grammar result:				77.8	16,480	970
	Average:	426.8	10,559.4	87.8734	77.87	15,850	545.2
	Best (size):	656	16,335	91.38	77.81	15,403	316
	Best (F-score):	625	15,554	89.44	78.01	15,608	347

Table 1: Results for partition tree search on full parsing task, WSJC Section 1 (averaged over 5 runs, variable parameters: $x = 100, b = 5, n = 5$).

Automatic methods for optimising and adapting grammars for new tasks and domains are particularly useful because context-free grammars cannot be learnt from scratch from data. At the very least, an upper bound must be placed on the number of nonterminals allowed. Even when that is done, there is no likelihood that the grammars resulting from an otherwise unsupervised method will look anything like a linguistic grammar whose parses can provide a basis for semantic analysis. The present method preserves the basic phrase structures and categories of the base grammar, while reranking its parse sets and eliminating phrase distinctions not needed for a given task.

Preliminary tests revealed that results were surprisingly constant over different combinations of variable parameter values, although training subset size of less then 50 meant unpredictable results for the complete WSJC Section 1. For a random subset of size 50 and above, there is an almost complete correspondence between subset F-Score and Section 1 F-Score, i.e. higher subset F-Score almost always means higher Section 1 F-Score.

Unlike other grammar compression methods (Charniak, 1996; Krotov et al., 2000), Partition Search achieves lossless compression, in the sense that the compressed grammars are guaranteed to be able to parse all of the sentences parsed by the original grammar.

Compared to other approaches using parent node information (Charniak and Carroll, 1994; Johnson, 1998b; Verdú-Mas et al., 2000), the approach presented here has the advantage of being able to select a subset of all parent node information on the basis of its usefulness for a given parsing task. This saves on grammar complexity, hence parsing cost.

4. Conclusions and Further Research

Grammar Learning by Partition Search was shown to be an efficient method for constructing PCFGs optimised for a given parsing task. The best grammar constructed by Partition Search in the reported experiments outperforms the best existing results for nonlexicalised parsing by a significant margin. In the same experiments, grammar size was reduced by up to 16.89%. Partition Search has the advantage of reducing grammar size without loss of grammar coverage while achieving an improvement in grammar performance.

In future research, the P_SEARCH procedure will be used as a testbed for comparing the performance of different search techniques, including greedy depth-first search and genetic search (only some of the subprocedures need to be replaced for each new search type). Further research will also look at additionally incorporating lexicalisation, and testing a wider set of variable parameter combinations.

5. Acknowledgements

The research reported in this paper was in part funded under the European Union's TMR programme (Grant No. ERBFMRXCT980237).

6. References

A. Belz. 2001. Optimising corpus-derived probabilistic grammars. In *Proceedings of Corpus Linguistics 2001*, pages 46–57.

A. Belz. 2002. Learning grammars for noun phrase extraction by partition search. In *Proceedings of LREC Workshop on Event Modelling for Multilingual Document Linking*.

Claire Cardie and Darren Pierce. 1998. Error-driven pruning of treebank grammars for base noun phrase identification. In *Proceedings of COLING-ACL '98*, pages 218–224.

Eugene Charniak and Glenn Carroll. 1994. Context-sensitive statistics for improved grammatical language models. Technical Report CS-94-07, Department of Computer Science, Brown University.

Eugene Charniak. 1996. Tree-bank grammars. Technical Report CS-96-02, Department of Computer Science, Brown University.

M. Johnson. 1998a. The effect of alternative tree representations on tree bank grammars. In *Proceedings of the Joint Conference on New methods in Language Processing and Computational Natural Language Learning (NeMLaP-3/CoNLL'98)*, pages 39–48.

Mark Johnson. 1998b. PCFG models of linguistic tree representations. *Computational Linguistics*, 24(4):613–632.

A. J. Korenjak. 1969. A practical method for constructing LR(*k*) processors. *Communications of the ACM*, 12(11).

A. Krotov, M. Hepple, R. Gaizauskas, and Y. Wilks. 2000. Evaluating two methods for treebank grammar compaction. *Natural Language Engineering*, 5(4):377–394.

Po Chui Luk, Helen Meng, and Fuliang Weng. 2000. Grammar partitioning and parser composition for natural language understanding. In *Proceedings of ICSLP 2000*.

H. Schmid and S. Schulte Im Walde. 2000. Robust German noun chunking with a probabilistic context-free grammar. In *Proceedings of COLING 2000*, pages 726–732.

H. Schmid. 2000. LoPar: Design and implementation. Bericht des Sonderforschungsbereiches "Sprachtheoretische Grundlagen für die Computerlinguistik" 149, Institute for Computational Linguistics, University of Stuttgart.

Jose Luis Verdú-Mas, Jorge Calera-Rubio, and Rafael C. Carrasco. 2000. A comparison of PCFG models. In *Proceedings of CoNLL-2000 and LLL-2000*, pages 123–125.

F. L. Weng and A. Stolcke. 1995. Partitioning grammars and composing parsers. In *Proceedings of the 4th International Workshop on Parsing Technologies*.

A Semantic-driven Approach to Hypertextual Authoring

R. Basili, A. Moschitti, M.T. Pazienza, F.M. Zanzotto

University of Rome Tor Vergata,
Department of Computer Science, Systems and Production,
00133 Roma (Italy),
{basili, moschitti, pazienza, zanzotto}@info.uniroma2.it

Abstract

Even if the use of the hypertext paradigm is nowadays very diffused, its potential benefits are not completely exploited by the community of the users. This is particularly evident in the case of the news agencies. The major reasons for the above limitation are the high costs for manually creating and maintaining the sets of complete links of a large-scale hypertext. This is especially true for news agencies. Therefore, in this paper we propose a method to address the problem of the automatic construction of the hyper-links based on Information Extraction techniques that enable documents (mainly news items) to be represented in a canonical form, hereafter called *objective representation* (OR). Our hyper-linking method is presented after the analysis of the traditional approaches to the same problem. We will describe the notion of objective representation and the formalism to express the linking constraints. Finally, we will sketch our future research work in the area.

1. Introduction

Even if the use of the hypertext paradigm is nowadays very diffused, its potential benefits are not completely exploited by the community of the users. This is particularly evident in the case of the news agencies. A survey, reported in (Outing, 1996), found that there were 1,115 commercial newspaper online services world-wide, 94% of which used a simplified version of hypertext which does not provide the full use of the hypertext capabilities of the WWW. The users may be able to navigate to a particular article in the current edition of an online paper by using hypertext links, but they must then read the entire article to find the information that interests them. The documents are dead ends in the hypertext, rather than offering starting points for explorations. In order to truly reflect the hypertext nature of the Web, links should to be placed within and between the documents.

The major reasons for the above limitation is, as (Westland, 1991) has pointed out, the high costs for manually creating and maintaining the sets of complete links of a large-scale hypertext. This is especially true for news agencies, given the volume of articles produced every day. Aside from the time-and-money aspects of building such large hypertexts manually, humans are inconsistent in assigning hypertext links between the paragraphs of documents (Ellis et al., April 1994; Green, 1997). That is, different linkers disagree with each other as to where to insert hypertext links into a document.

The cost and inconsistency of manually constructed hypertexts does not necessarily mean that large-scale hypertexts can never be built. It is well known in the IR community that humans are inconsistent in assigning index terms to documents, but this has not hindered the construction of automatic indexing systems intended to be used for very large collections of documents.

The taxonomy of link types given in (Allan, 1995) is very useful to understand the problem of the automatic construction of hyperlinks since it classifies links according to the abilities required for an eventual manual construction. Links are classified according the following classes:

- *Pattern Matching links*, which are easy link to discovered as they can be found through a pattern-matching algorithm. An example of these is glossary links or links between proposition.
- *Automatic links*, which can be in part captured by traditional Information Retrieval techniques. For example links among documents discussing about the same topics.
- *Manual links*, which require text analysis at level of Natural Language Understanding.

While the first two types of links have been approached successfully the third one is judged by Allan (Allan, 1995) to be inaccessible to automatic hypertext construction.

In this paper we propose a method to address the problem of the automatic construction of the "manual" links as defined in (Allan, 1995). The proposed method is based on Information Extraction techniques that enable documents (mainly news items) to be represented in a canonical form, hereafter called *objective representation* (OR). This latter describes some of the important information contained in the documents, mainly the named entities and the domain events found in the target document. Therefore, this document representation allows to draw more motivated inter-document hyper-links since a declarative language for describing linking constraints can be settled over it. Linking rules, i.e. the rules that justify a link between two documents, are in fact written as constraints over the related ORs. The detection of the domain events and of the named entities relies on a knowledge-based IE system composed by a robust parser (Basili et al., 2000b) and a discourse interpreter (Gaizauskas and Humphreys, 1997). As any IE system, this linking methodology requires a large domain knowledge base. The overall approach foresees the methods for the automatic extraction of this knowledge in an unsupervised fashion (Basili et al., 2000a; Basili et al., 2002).

Our hyper-linking method is presented in Sec. 3. after the analysis of the traditional approaches to the same problem (Sec. 2.). We will describe the notion of objective representation and the formalism to express the linking constraints. Finally, we will sketch the future work (Sec. 4.).

2. Traditional Approaches

In literature the automatic construction of hypertext is based on classical IR techniques to measure the relatedness of document couples. Only a *bag of words* are used for expressing the document contents. This results in a poor set of link type manageable in automatic way. In (Allan, 1995) is presented a reformulated taxonomy of links (Trigg, 1983) in order to identify the link type achievable with an automatic approaches. The set of link type has been divided into three major categories based upon whether or not their identification can be carried out automatically (with the IR current technology). The three categories are *Pattern-matching*, *Automatic* and *Manual*. Unfortunately, some types of links straddle the boundaries of the taxonomy, depending upon the document collection being linked.

Pattern-matching Links is a large class of link types. They can be found easily using simple pattern-matching techniques. An obvious example of such a link type is *definition* that can be found by matching words in a document to entries in a dictionary. In almost cases, these links are from a word or phrase to a small documents. They do not take into account the context of the definition so the destination document may be the same for the word or phrase searched for; no matter where the word or phrase occurs. Structural links belong to the pattern-matching category. They are those that represent layout or possibly logical structure of a document. For example, links between chapters or sections, links from a reference to a figure to the figure itself, and links from a bibliographic citation to the cited work, are all structural links. They can be discovered by mark-up codes embedded in the text. Pattern-matching links form a class that is computationally simple for automatic detection.

Automatic Links are links which cannot typically be located trivially using patterns, but which the automatic IR techniques can identify with marked success. Typical automatic links that can be identified are:

- *Revision links* are a fairly straightforward class of relationship between texts, including both ancestor and descendent relationships.
- *Summary and expansion links* are inverses of one another. A summary link type is attached to a link that starts at a discussion of a topic and has as its destination a more condensed discussion of the same topic. Equivalence links represent strongly related discussions of the same topic.
- Tangent links are equivalence links that relate topics in an unusual or tangential manner (often by comparison with other links). For example, a link from a document about *Sivlio Berlusconi* as Italy Prime Minister to one about Milan football club (whose Berlusconi is the president) would be a tangential link.
- Aggregate links are those that group together several related documents. An aggregate link may in fact have several destinations, allowing the destination documents to be treated as a whole when desirable.

Manual links are those which are judged by the IR community unable to be located without human intervention. The natural language understanding researchers have had some significant success within constrained subject areas, so some manual links could be automatically described within those limited domains. Unfortunately, those techniques are not yet extensible to a general setting, so this class of link types seems to remain inaccessible to automatic approaches. Manual links include those which connect documents which describe circumstances under which one document occurred, those which collect the various components of a debate or argument, and those that describe forms of logical implication (caused-by, purpose, warning, and so on).

2.1. A more semantic based approach

An attempt to extend the boundaries of automatic links towards the manual links has been done in (Green, 1997). In this work an automatic method for the construction of hypertext links via *lexical chains* has been carried out. Lexical chains capture the semantic relations between words that occur throughout a text. Each chain is a set of related words that captures a portion of the cohesive structure of a text. By considering the distribution of chains within an article it is possible to build links between the paragraphs. A link is activated if the similarity score of the chains contained in two different articles overcome a predefined threshold. The method comprises three steps: determining the lexical chains in a text, building links between the paragraphs of articles, and building links between articles. A comparison of this methodology with the traditional IR techniques resulted in higher user satisfaction. Lexical chains allow to retrieve a wider set of link type. As an example let us consider two documents that speak about the same fact with different words. The scalar product (a wide used IR metrics in the Vector Space Model) between the two documents would be very low as the documents have different *bag of words*. This prevents the activation of a relatedness link. On the contrary lexical chains refer to the meaning of words. They use synonyms of words in texts so their similarity between documents will be higher.

Lexical chains seems to solve some of IR problem in discovering links but some problems remain unsolved:

- The link type of two documents, which have similar lexical chains, is unknown. We could claim as an explanation that the documents contain some related semantic information. However this explanation is too generic as it is valid for each generated link.
- *Consequence links* remain unsolved. It is not possible specify the consequence relation between two documents for two main reasons: a) The lexical chains of the premised tend to be very different from the consequence. b) These links are directional while the similarity between chains is symmetric.

- Ambiguity and data sparseness affect the precision in discovering valid chains. So we can expect a lot of wrong links.

In next Section it is presented a different approach that solve the two first problems. It provides a methodology for capturing the unsolved link as well as the explanation for them. The third problem has been bound using domain knowledge for conceptualise the information.

3. A "semantic-driven" hyper-linking method

The above approaches mainly relate documents if they are enough similar according to the chosen document representation space, i.e. the bag-of-word abstraction or the lexical chain model. Therefore, according to these approaches "relatedness" is the only reason why two documents may be hyper-linked together. However, this notion of relatedness does not give the possibility of defining user-oriented hypertexts. Each user has to be exposed to the same hypertext regardless his information needs. For instance, the above approaches may relate the two news items in Fig. 1 because of the fact that in the two documents the *Intel* stem increases the relatedness of the two documents. However, the link user is not aware of the reason why the two documents are related and, while reading the first news item, he has not hints that may suggest if the related news article is of any interest to him. The justification of the link may be more easily highlighted if the domain relevant information is captured, i.e. the fact that both the first item and the second one describe an *Intel acquisition activity*.

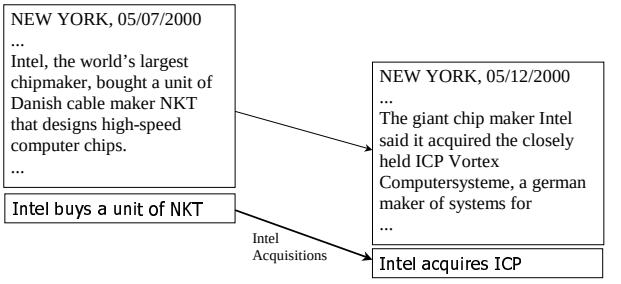


Figure 1: An example of justified link

The facts justifying the hyper-link between the two documents are respectively:

- Intel buys a unit of NKT
- Intel acquires ICP

It is worth noticing that a very precise information is needed for linking the two documents, i.e. the "equivalence" of *buy* and *acquire*. This information may be also used in an IR based hyper-linker using a query expansion technique but the justification of the link is still very difficult.

Furthermore, this notion of relatedness limits the possibility of linking documents. For instance in Fig. 2,

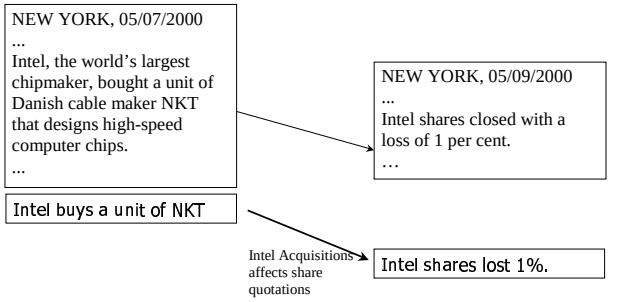


Figure 2: A complex justified link

the link between the two documents is justified by the fact that an *Intel acquisition* affects the *share prices* of a particular period of time. The facts justifying such a kind of document relation are respectively:

- Intel buys a unit of NKT
- Intel shares lost 1%

Such a kind of link is very difficult to capture if the analysis is not based on the more structured document representation.

The automatic hyper-linking method we propose is then based on an abstraction of the document, the objective representation (OR) that describes in a canonical form the salient information carried by the document. This objective representation, due to its nature, may be also considered language independent. Therefore, it enables the automatic hyper-linking between documents of different languages. Both his canonical representation, i.e. the OR, and the language for defining the linking constraints are described in the following sections.

3.1. The objective representation

The quality of the hyper-links that may be drawn in such a method strictly depends on the assumed representation of the document content. Furthermore, it is crucial that the intended information is actually captured by the IE system.

The objective representation we have defined is not too far from the actual document content and aims to represent the relevant document information with respect to a given knowledge domain. In particular, given a document *D*, its OR contains the named entities and the main events of the document *D*. These latter mainly represent particular domain relevant verb phrases that appear in the document. Both the named entities and the events are classified according to a knowledge representation scheme related to a target domain.

The objective representation is then a couple $OR(D) = (NEs, Events)$ where NEs is the set of the categorised named entities of D while the $Events$ is the set of the categorised events. Each event in $Events$ has the following form:

$$EventType(Verb, Arguments) \quad (1)$$

where $EventType$ is the type of the event, $Verb$ is the actual verb that appears in the document and $Args$ are the arguments of the verb according to the event type. Each argument representation carries its syntactic/semantic relation, the actual lexical of its semantic governor, and the type of this latter. For instance, the documents in Fig. 1 should contains respectively in their ORs the following events:

- `buy_event( agent(company,Intel), patient(object,a_unit_of_NKT))`
- `buy_event( agent(company,Intel), patient(company,ICP))`

Naturally, the efficacy of the OR strictly depends on the nature of the information that is contained in the knowledge base. The method for extracting such a knowledge and for the definition of the equivalence between different surface forms is described in (Basili et al., 2002).

3.2. Typing links using events: a declarative formalism

Once an objective representations of documents are available it is possible to write down a set of rules that can activate several links that traditional IR techniques (see Section 2.) cannot capture. However it is not possible to define general linking rules valid for each domains and for each user needs. As an example consider two documents: d_0 that speaks about Ferrari race in the grand prix of Imola and d_1 in which it is stated that FIAT market shares increase their quotation. If a user wants know all the facts which cause the event in d_1 (e.g. the document d_0) some knowledge about the correlation between FIAT and Ferrari have to be draw (i.e Ferrari is a part of FIAT and winning a race increases the share value of a Company).

Thus a systems that really wants to afford hypertext construction including links of third type (see Section 2.) should provide both a set of general rules and a set of specific rules. Moreover, the specific rules should be customisable to satisfy a wide range of user needs. These rules will be then used by the linking algorithm to draw links among documents.

3.2.1. The linking rule formalism

We have adopted a declarative formalism in which the rules and the knowledge required are easy to be written by the final user. The rules are expressed by a logical formalism.

The events in the OR are coded by means of Prolog predicates of the following type:

```
ev(EVENT_CATEGORY, EVENT_LEX,[
  arg(AGENT, AGENT_CATEGORY,
        AGENT_LEX),
  arg(DIROBJ, DIROBJ_CATEGORY,
         DIROBJ_LEX),
  arg(MODIFIER1, HANDLE1, LEX1),
  ...,
  arg(MODIFIERm, HANDLEm, LEXm)
]).
```

The first two arguments of the predicate `ev` are the category and the lexical of the *event* (i.e. the category and the lexical of the action accomplished by the object versus the direct object). The third argument is a set of participants (agent and direct object and modifiers), expressed as list of Prolog predicates. The category of the agent (AGENT_CATEGORY), the category of direct object (DIROBJ_CATEGORY) as well as their lexical form (AGENT_LEX and DIROBJ_LEX) are included in the predicative description of the event argument (`arg`).

Linking rules should therefore describe when two news items have to be linked together. These are written over the objective representation of the investigated documents. In particular, they exploit the notion of event. Linking rules are then Prolog predicates defining a linking criteria that motivates the existence of an link among the source and the target news items from which events are derived. Linking rules define all the constraints that the participants of two events must satisfied for generating a link between them. Each generated link has therefore a *LINK_TYPE* that is determined by the application of a specific rule. In order to compile a linking rule a list of pre-defined constraints, expressed as predicates, needs to be defined. The constraints act over the basic constitutes of an event (i.e. event lexical/category, subject, object and modifiers). In particular as the event category and lexical are supposed to have a different semantic from subject, object and modifier, two type of constraints have been defined. More precisely a linking rule is a Prolog predicate of the form:

```
lrule( LINK_TYPE,
       SOURCE_EVENT_CATEGORY,
       TARGET_EVENT_CATEGORY,
       SET_OF_EVENT_CONSTRAINTS,
       SET_OF_ARGUMENT_CONSTRAINTS)
```

where:

- *LINK_TYPE*, is the type of the link that is generated by such a rule.
- *SOURCE_EV._CATEGORY* and *TARGET_EV._CATEGORY* are the category of events involved in the linking rule. For example in case of an event that relates to a meeting and another event that relates to an acquisition of stocks in that meeting, it would be useful to have a linking rule characterised by MEETING_EVENT as category of source event and BUY_EVENT as category of target event.
- *SET_OF_EVENT_CONSTRAINTS* is the set of constraints to be activated on the event category/lexical information of the source and target events.
- *SET_OF_ARGUMENT_CONSTRAINTS* is the set of constraints to be activated over the arguments of the source and target events.

Given the above description a linking rules which expresses correlation between the participants of a meeting and a company acquisition in the meeting could be:

```
lrule('Acquisition during a meeting',
       MEETING_EVENT, BUY_EVENT,
```

```
        SET_OF_EVENT_CONSTRAINTS,
        SET_OF_ARGUMENT_CONSTRAINTS)
```

The *SET_OF_ARG._CONSTRAINTS* specify relation between the participants of the meeting and those that acquire something. The *SET_OF_EVENT_CONSTRAINTS* specify the relation between MEETING_EVENT and BUY_EVENT as well as the lexicals associated to them.

3.2.2. Expressing constraints in the linking rules

The aims of the constraints are to select the properties of the participants and the properties of the event categories. These constraints compositionally build linking rules. A simple set of constraints is:

- *Category Identity*: two participants must be of the same category. This implies that two entity must belong to the same class. For example IBM and INTEL are both companies so they belong to the company category. If a Category identity constraint is included inside a SET_OF_EVENT_CONSTRAINTS, it casts different events to be in the same category. If we use this constraint leaving unspecified the event category we are grouping together event of the same category.
- *Lexical Identity*: the participants must have the same lexical e.g. the participant *Bill Gates* is the same lexical in *Bill Gates buy IBM* and in *Bill Gates get married*. A rule based on the category identity constraint would not be useful in the above case as a lot person get married. The Lexical Identity for the set of event constraint is less meaningful. However it can be used to specify the relation involved in a couple of events more precisely. For example if we have the event *Bill gates sell IBM*, its category will be *acquisition*. This information would not useful if we want build a rule for capturing document about company selling.
- *Conceptual Similarity*, it is an extension of the category identity type. In this case categories are grouped in a hierarchical structures. It is possible to express a relation of parents among participants.

Given the above constraints the following events:

```
ev(MEETING_EVENT, invite,[
  arg(AGENT, Company, Intel),
  arg(DIROBJ, person, Bill Gates),
  arg(MODIFIER, in, Seattle)
]).
ev(BUY_EVENT, acquire,[
   arg(AGENT, person, Bill Gates),
   arg(DIROBJ, company, Intel)
]).
```

two sample rules for capturing the link type *Acquisition during a meeting* are:

```
lrule('Acquisition during a meeting',
       MEETING_EVENT, BUY_EVENT,
       [],
       [lex_id(AGENT,DIROBJ)]
).
```

```
lrule('Acquisition during a meeting',
       MEETING_EVENT, BUY_EVENT,
       [],
       [lex_id(DIROBJ,AGENT)]
).
```

It is worth noticing that the above rule involves general events so the information about participants has to be more specific (i.e. lexical information about participants is needed). This pushes for the use of *lex_id* constraint.

Another generic rule is that groups document speaking about a target agent doing whatever action. For example the events in which Bill Gates buy something could be captured by the following rule:

```
lrule('Same participants rule',
       _,_,
       [cat_id()],
       [lex_id(AGENT,AGENT)]
).
```

In the above rule the only requirement is the same agent in the linking documents. The agents in the target events have to do an action of the same category type (e.g. Acquisition event, Announce event, Market strategy events,...).

When is needed grouping together documents in which a target action is carried out, it is possible to use the category constraints for the agent and object (i.e. the cat_id constraint). For example a linking rule in which agents of the same category make acquisitions of object of the same category is the following:

```
lrule('Person acquire Company',
       BUY_EVENT, BUY_EVENT,
       [],
       [cat_id(AGENT,AGENT),
        cat_id(DIROBJ,DIROBJ)]
).
```

3.3. The linking algorithm

Once the linking rules formalism has been developed it is possible to design the linking algorithm. This should takes as input the ORs of two documents: the source and the target. For each couple of events in the source and in the target, the linking rule database $LRDB$ is considered. If some rule is matched a link is generated and it is stored in a link DB. The rules are composed of some basic constraints that act on the constituents of an event. In this way, if an extended list of basic constraints is available it is possible for the user to define several linking rules. The rule can be described in an external data file so new rules can be added to the similarity model without re-designing the entire architecture.

The linking algorithm takes as input two documents, one is the source S and the second is the target T, and given their sets of events, respectively $Ev(S)$ and $Ev(T)$, check if any couple $< ES, ET >$, where $ES \in Ev(S)$ and $ET \in Ev(T)$, satisfy any of the linking rules contained in $LRDB$.

The algorithm is composed of the following steps:

```
function Link(text S, text T) returns Linkset
begin
   Linkset L = ∅ ;
   Ev(S) = BuildEv(S) ;
   Ev(T) = BuildEv(T) ;
   for each (ES, ET) ∈ Ev(S) × Ev(T)
      begin
      while ((R = SelectNextRuleFor(ES, ET)) != NULL)
         begin
         R = (RuleType, SEvCat, TEvCat, CatConstr, ArgConstr);
         if ( ApplyCatConstr(CatConstr, ES, ET) == true )
         begin
         boolean sat = true;
         while (SArg, TArg) ∈ nextArg(ES, ET) AND sat )
            sat = ApplyArgConstr(ArgConstr, SArg, TArg) ;
            if (sat)
               AddLink(ES, ET, RuleType, L)
         end
      end
   end
return L ;
end
```

4. Conclusions and future work

In this paper, we have presented a methodology for the automatic hyper-linking among news items. The presented approach is based on Information Extraction techniques that give the possibility of building semantically motivated links among documents. This approach is more expressive that the traditional approaches to the problem that allows the automatic construction of links only between related documents. The approach has been used to build the Namic prototype (EU-founded project NAMIC, News Agencies Multilingual Information Categorization, IST-99 12392).

As the approach is rather different from the pre-existing the comparison is hard. We will therefore compile, according to our definition of the task, a large test set that should enable the validation of the methodology and of the implemented system.

5. References

James Allan. 1995. Automatic hypertext construction. Technical Report TR95-1484, 13.

Roberto Basili, Maria Teresa Pazienza, and Michele Vindigni. 2000a. Corpus-driven learning of event recognition rules. In *Proc. of the Workshop on Machine Learning for Information Extraction , held jointly with ECAI2000*, Berlin, Germany.

Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. 2000b. Customizable modular lexicalized parsing. In *Proc. of the 6th International Workshop on Parsing Technology, IWPT2000*, Trento, Italy.

Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. 2002. Lerning ie patterns: a terminological perspective. In *Proc. of the Workshop on Event Modelling for Multilingual Document Linking, held jointly with 3rd LREC*, Canary Islands, Spain.

David Ellis, Jonathan FurnerHines, and Peter Willett. April 1994. The creation of hypertext linkages in fulltext documents: Parts i and ii. Technical Report RDD/G/142.

Robert Gaizauskas and Kevin Humphreys. 1997. Using a semantic network for information extraction. *Natural Language Engineering*, 3, Parts 2 & 3:147–169.

S. Green. 1997. *Automatically generating hypertext by computing semantic similarity*. Ph.D. thesis, Department of Computer Science, University of Toronto.

Steve Outing. 1996. Newspapers online: The latest statistics. *AEditor and Publisher Interactive [Online]*.

Randall H. Trigg. 1983. *A networkbased approach to text handling for the online scientific community*. Ph.D. thesis, University of Maryland.

J. Christopher Westland. 1991. Economic constraints in hypertext. *Journal of the American Society for Information Science*.

Ontology-based author profiling of documents

Jan De Bo, Mustafa Jarrar, Ben Majer, Robert Meersman

VUB STARLab
Vrije Universiteit Brussel
Pleinlaan 2
Brussels Belgium
{jdebo, mjarrar, bmajer, meersman}@vub.ac.be
Telephone:+32 2 6293487 Fax:+32 2 6293525
Home Page: http://www.starlab.vub.ac.be/

Abstract

In this paper we present the advantages of using an ontology service for the modelling of user profiles in the EC FP5 IST project NAMIC (IST-1999-12392). By means of an ontology server people set up user profiles, which are in fact views, i.e. *specifications of queries* on the ontology. These views are constructed using a JAVA API, which forms the *commitment layer* of the ontology, built on top of an ontology base. In NAMIC an ontology server is used to establish a link between the lexical object representations, generated by the natural language processors (NLP) on the one hand and the user's interest, specified through the selection of relevant concepts and facts of the ontology on the other. This allows to specify a user profile independently of language, categorization and NLP specific "world models". Users then set up a profile consisting of events, agents participating in these events and other content information in which they are interested in. For instance, a journalist writing articles about financial issues may be interested in related documents containing a "raise event" of company shares. If he has specified those conditions in his profile he will be able to retrieve resources which contain events that are semantically related to that kind of event pattern. User profiles in NAMIC do not have to be static. The results of processing by the NLPs of a document the user is currently working on, may be used to construct a dynamic profile, which may contain events specific for that document. This way a user's profile can be dynamically adapted to his current interests. We also developed a tool which illustrates the creation of user profiles using ontological concepts and facts.

1. Introduction and Motivation

In this paper we present results derived from our work in the NAMIC project. Within the NAMIC project the main objective was to develop advanced technologies of Natural Language Processing for multilingual news customization and broadcasting throughout distributed services, which represents one of the major problems for International and National News Agencies (NA) as well as for the spread of Web technologies. Within their own business cases, NAs need to integrate in their own repositories news distributed by other NAs usually in different languages and according to different classification standards. Mismatching is at language level, since different languages are used, as well as at the conceptual, as the organization/storage of news proceeds according to diverging schemes. The volume and richness of this information has, however, a catch: it can overwhelm the pressed user-journalist that may be looking for a particular type(s) of information. This is a well-known problem in an information-rich environment, and especially in the case of (large) sets of hyperlinked documents, often referred to as the "lost-in-hyperspace syndrome".

Several aspects have been researched to improve searching, browsing and retrieval of information. In the information retrieval approach, several techniques ranging from string matching to advanced lexical analyses systems are used in order to understand the implicit semantics and thus the relevancy of the data that will be retrieved. On the other side, in the artificial intelligence and database approaches, such as for example the semantic web, the semantics (and the syntax) of the data are explicitly defined and linked with knowledge bases as ontologies, which help to make precise queries or for reasoning,. Experience shows that the accuracy of extracting the implicit semantics and the relevancy of the data is low, e.g. a search using regular search engines results in a huge amount of information, especially for large volume information resources such as the web, expanding queries to improve recall may also cause huge result sets. On the other hand, defining the semantics of the information explicitly, and reasoning about them in order to retrieve relevant information is an expensive task, and the scalability is very low. Therefore, we believe that combinations of these two approaches will be very fruitful for the improvement of information retrieval, as will be argued in the next sections of this paper.

Within the NAMIC project the User Domain Profiling System (UDPS) allows defining of user profiles for the filtering of news streams according to the specific interests of a user which for NAMIC, primarily would be journalists or other text writers. These user profiles are then used to exclude irrelevant items from a constant stream of documents before these documents are presented to the user.

As will be argued later in this paper, the use of an ontology has critical improvements: IR systems will gain from ontologies richer knowledge representation and modelling capabilities, improved recall by expanding the queries according to well-defined and consistent relationships in the ontology and improved precision by allowing the definition of personalised profile systems as queries against (an) ontologie(s) in order to include or exclude (a) certain type(s) of information.

Structure of the paper. In section 2 we give an introduction of what an ontology is and its critical added value for NLP based systems. Section 3 then gives the definition of a user profile and explains more details about the advantages of using ontology-based information filtering systems such as user profiles. Section 4 demonstrates the implementation done in the Namic Project and Section 5 draws preliminary conclusions and

maps ongoing and future work. Section 6 then places all acknowledgements.

2. Using ontology with NLPs

In this section we will illustrate the advantages of ontologies and their potential role in several aspects of information retrieval and how they can be used in defining user profiles.

Ontology[1] in computer science is a branch of knowledge engineering, where agreed semantics of a certain domain are represented formally in a computer resource, which then enables sharing of information and interoperation between systems. Representing the semantics (as a formal interpretation) of a certain domain implies the conceptualisation of the domain objects and their interrelationships in a declarative way, so that they can be processed, shared, and reused among different applications. Note that an ontology is more than a taxonomy or classification of terms, since it includes richer relationships between terms, e.g. "part-of, location-of, value-of, synonym-of…"(Figure 1). An ontology provides a higher level of knowledge[2], where the ontology terms are chosen carefully, consistently, and with a higher level of abstraction.

In the DOGMA model described summarily below, we separate relevant ontological relationship knowledge as set extensions of context-specific binary fact types called *lexons.* These express (within this assumed context) plausible relationships between concepts, using lexical terms in a given language; we implicitly assume that these terms are aligned with a lexicon ("terminology base") that is agreed among all users of the ontology (Jarrar, 2002).

Example. The following –very partial ontology (Tables 1,2,3)- could be lexons in some arbitrary hopefully self-understood syntax, the format for the purpose of textual illustration being (*#contextid*) <term1>[<role label><term2>]; details or omitted in this paper. The ontology base, which contains the set of lexons of the modelled domain, is also known by the symbol, Ω.

(*#my_company-ID*) employee
is_a person
has first_name
has last_name
has empl-id
has_birth date
has salary
works_in department

Table 1

(*#my_company-ID*)salary
is_a salary
reviewed_in month

Table 2

(*#employment-ID*) salary
has amount_in-$
expressed_in currency
converted_to currency
earned_by employee

Table 3

Through the use of ontologies one is able to express semantic relations between terms, rather than is the case with ordinary categorisations. To express these meaningful relations between different terms we need advanced modelling methodologies, like the ORM conceptual modelling language. We chose ORM for its rich constraint vocabulary and well-defined semantics. Within STARLab we also developed an XML-based ORM markup language (ORM-ML) as a means of exchanging data semantics between different agents. (Demey et al, 2002)

The enormous growth of the Web causes search engines to return a large number of pages to the user for a single search. It is time consuming for the user to traverse the list of pages just to find the relevant information. We claim that information filtering systems based on ontologies will assist the user by filtering the data stream and delivering more *relevant* information to the user. Below are a few examples of how this can be achieved. We will discuss these topics in section 3 in more detail.

IR will benefit from ontologies more than terminology bases/resources since the knowledge is more formally represented than in term bases, which facilitates the representation, maintenance, and dissemination of terminological data and makes these data reusable by computer systems in various applications. Recall and precision of search operations will be improved using ontologies to model the knowledge contained in a system. Recall will be improved by exploiting the rich structure of an ontology and specifying generic queries (Guarino, 1999). The semantics in an ontology makes it quite attractive for query expansion, because there is a strong need to expand queries with relevant terms and meaningful relations which contain a lot of semantics, for instance to include subtopics or to personalize the query according to a user's personal interests. Precision will be increased through the disambiguation of terms and the ability to navigate through the ontology for the selection of more specific queries (Guarino, 1999).

While ontologies offer highly advanced modelling capabilities our experience indicates that, in the domain of Natural Language Processors (NLPs), ontologies will mostly be lighter, and therefore less expressive, than in other applications such as for example reasoning systems where the reasoning rules (defined as a logical theory in the *commitment layer*; containing for example the following constraint ORM.Mandatory(employee has_birth date)) are the most important part of the ontology, while NLP applications may see the lexons in the ontology base as canonically and linguistically structured expressions.

Furthermore, the context will provide added value to disambiguate (or approximate) the meaning of terms and relations.

Usage of an ontology also offers advantages for multilingual Information Retrieval. Since the ontology is a shared agreement about a (abstract) conceptualization it is in principle independent of a particular natural language

[1] In philosophy, Aristotle defined ontology as the science of being.

[2] The Knowledge Level is a level of description of the knowledge of an agent that is independent of the symbol-level representation used internally by the agent, (Gruber, 1995)

(Of course, one needs in general natural language to negotiate and specify such an agreement). Thus an ontology should be able to support multilingual retrieval of information by allowing the definition of conceptual queries, which are not natural language specific. Relevant information can then be retrieved through the matching of a query with the conceptual information/knowledge extracted from document corpora in other languages.

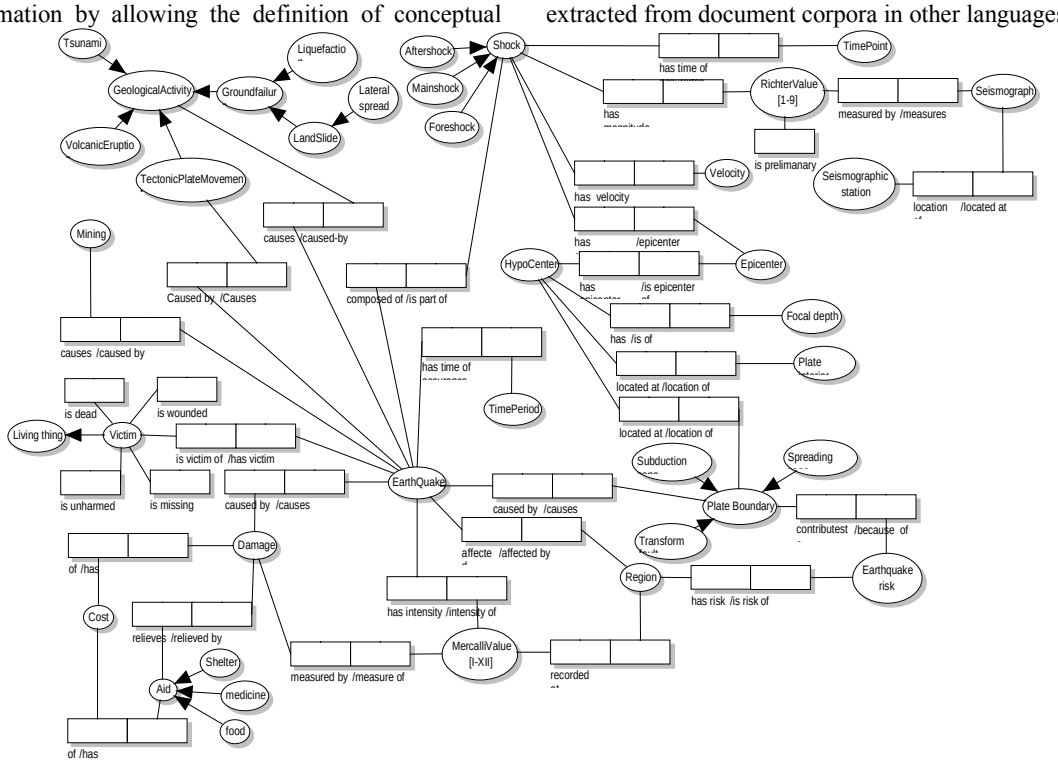


Figure. 1

3. Profiling system

While search engines find relevant items from a constant stream of documents, personalized information filtering systems generally embody one or more of a user's interests via a user profile, which ultimately improve the precision. Filtering systems are often classified into one of two categories, depending on the manner in which the documents are filtered. *Cognitive systems*, also referred to as content-based systems(Pretschner et al,1999), choose documents based on the characteristics of their contents, while *social systems*, also referred to as collaborative filtering systems, select documents based on recommendations and annotations of other users (Pretschner et al,1999; Abuzir et al,2001; Abuzir et al, 2002).

An efficient and semantics-based filtering mechanism is desirable in order to improve the precision of the results. Individual users (or a certain audience/class/social group… of users) will specify in profiles which kind of information should be included or excluded. A profile may for instance contain the following filter " 'Company acquisition event' and 'IBM' ", expressing the user is interested in all company acquisition events and in all events involving IBM.

Note that such a profile is not just a set of arbitrary keywords that may lead to inconsistent filtering, but forms a consistent and well defined filter mechanism, based on the same semantics as the query engine (or NLP). Therefore, we define a user profile as a specification of a query on the ontology. A profile enables a user to specify his interests and expresses this way what kind of documents he is interested in. A user profile is composed out of one or more filters, where each filter specifies which class(es) of information the user wishes to include or exclude. Within the NAMIC project in particular, the profile specifies those conditions that should result in "exactly" news items of interest of the journalist-user.

Defining a user profile as a query on the ontology thus implies the specification and adoption of a query language/system. Therefore, defining such profiles depends on how the relations between ontological concepts will be interpreted, e.g. one may decide to include all of a class' subclasses automatically. Within the NAMIC project we have chosen to specify a query as a composition of logic combinations using concepts and binary relations from the ontology, in which the concepts and the taxonomic relationships between them are seen as forming a kind of frame-based system (Karp, 1993).

Often queries are very broad. Consider for example the query "EU Framework 5". With a database as large as the Web, there will be thousands of documents that are related to EU Framework 5. If a query can be expanded with the user's interests, the search results are likely to be more narrowly focused. However, this is a difficult task since query reformulating needs to expand the query with relevant terms. If the expansion terms are not chosen appropriately, even more irrelevant documents will be returned to the user. By taking the semantics of the domain into account, it turns out user profiles can be an excellent source of knowledge to expand the query. By specifying an ontological concept in a user profile, a user implicitly selects all concepts from the ontology which

inherit from this concept and ignores all parent concepts (assuming the relation between the concepts is SubClassOf).

The ontology is separated from the objective representations used by the natural language processors. Since the user profile is a query on the ontology, this separation hides the user from the potentially large amount of objective representations used by the NLPs. The advantage of the independence between the underlying objective representations and the user setting up his profile is that he does not have to be aware of the different objective representations of the NLPs. The ontology can thus be seen as an intermediate level shielding the different representations of the NLPs from the user. Once the ontology is built, natural language processors will have to adapt their objective representations to it. This way a query on the ontology, can be considered to interact independently with the objective representations generated by various natural language processors.

Because of the multilingual data resources, development of different natural language processors (in NAMIC, English, Spanish and Italian) is required. This was done by the universities of Sheffield, Rome (Tor Vergata) and Catalonia (Universitat Politècnica de Catalunja). The user profiling system, introduced in NAMIC, however enables the user to specify language-independent queries, but still gives the possibility to get back related documents in all languages provided by the news agencies.

As mentioned before a user has the possibility to specify his interests in a static profile by selecting the appropriate relations and concepts from the ontology. It is however quite possible that a journalist's interests change while working on a particular news story. Therefore the user has to adapt his profile according to his current needs and interests instead of having to create an other additional profile. User profiles, developed within the NAMIC project, can be dynamically adapted. Indeed, as part of the NAMIC profile services, a journalist has the possibility to create a local profile according to the text he is currently working at, because it is likely that he will be interested in retrieving documents containing events, or knowledge related to agents participating in events which he has already entered in his text. The user is given the possibility to update his current static profile according to this new profile, making his own profile change dynamically. This prevents the user from having to manually annotate his own article of text by adding (ontologically derived) concepts and relations to his static profile, assumedly saving time and improving consistency.

4. Implementation

The ontology service in NAMIC provides the possibility to store, edit and retrieve ontological information that models (partial) semantics relevant to the project's domain and in particular the ability to define user profiles based on these semantics.

In order to satisfy the requirements mentioned above we developed a tool, with the following classical two-tier client/server architecture, illustrated in Figure 2.

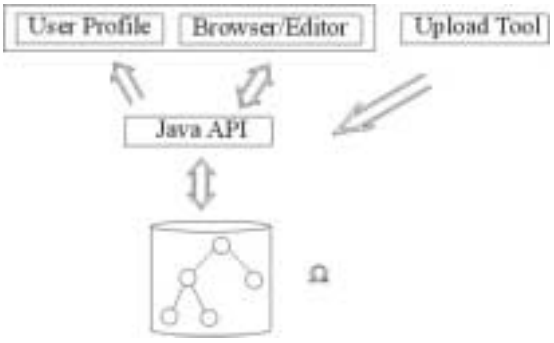


Figure 2

- At the bottom of Figure 2, there is a storage facility for the ontology (in a database)
- Above that, an intermediate API layer establishes communication between various tools and the ontology.
- At the top, support tools like browsers, editors and user profiles are implemented.

In our paper we will use the term 'objective representations' of the natural language processors to refer to Event Matching patterns, which are described in detail in (Basili et al) .The process of ontology engineering begins with the development of a base model that provides a framework for the integration of other different, individual resources. The creation of this ontology base can be viewed as a conceptual modelling task, based on ontology merging and alignment of the available resources. The result contains the fundamental concepts based upon the natural language processors' objective representations, that are generally useful for the project. For instance, consider the following verb syntactic frame: 'person – sells – attribute' as an example of an objective representation from the NLPs' event matching rules. The verb syntactic frame which is not considered to be an ontological concept, is mapped to 'Company Acquisition event'. The occurrence of this verb syntactic frame in a document then results in the detection of a 'Company Acquisition event'.

The individual resources that are considered for their incorporation into the NAMIC ontology were the following:

- The IPTC category system (IPTC)
- The EuroWordNet base concepts (EuroWordNet toplevel concepts) (Vossen, 1998)
- Named Entity lists (Stevenson et al)
- Event Types (Basili et al)

In order to integrate the natural language processors' objective representations of the different individual resources into the ontology, an alignment process needed to be performed between those different representations. Categories, events and named entities are aligned with EuroWordNet base concepts, by establishing mappings between the involved concepts of the different resources considered for integration in the ontology. This is illustrated in Figure 3; the alignment mappings are depicted as double-sided arrows.

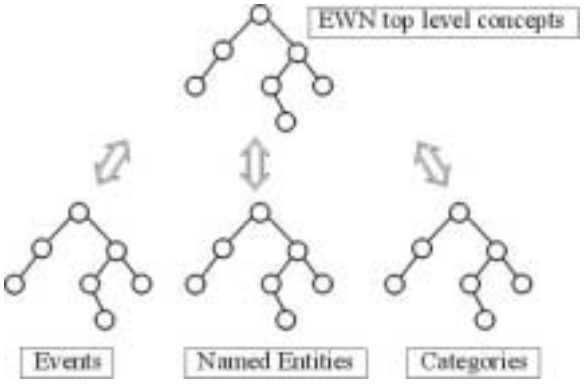


Figure.3

Because an ontology is a *shared* agreement about (a conceptualisation of) the world, aligning different ontologies with one another is required in order to obtain agreement between the concepts of the different ontologies. In order to develop tools automating this activity, good context formalisms will undoubtedly become helpful here but within the scope of NAMIC we had to align the different ontological concepts manually. At this state of the art it is as yet unrealistic to expect that merging or alignment at the semantic level could be performed completely automatically. A prototype of a tool to assist ontology merging and alignment has been built by the Stanford Medical Informatics department of Stanford University. This tool, based on the SMART algorithm, is an extension of the Protégé (Noy, 1999) ontology-development environment.

For the purposes of NAMIC we have also developed a simple custom tool (OntoNAMIC) to make the ontology available for browsing, editing and setting up user profiles.

The browser window consists out of a left pane and a right pane. The left pane is responsible for browsing through the ontology, while the content appearing in the right pane depends on whether one has selected the class view, diagram view or profile view on the toolbar of the application.

When the domain expert (i.e. typically *not* the journalist) selects the Classview, all the lexons containing the selected concept on the left will be displayed in the right pane. Choosing the Diagram view enables one to drag and drop concepts from the left pane into the right. By double-clicking on this dropped concept an ORM diagram appears, displaying all the lexons of which the concept is a part. ORM is a well-known conceptual modelling language (Halpin, 2001) here "re-used" (in part; some interesting modifications are needed that however will be the subject of a separate paper) to represent part of the ontology. In the diagram, ovals represent *entity types*, the rectangles are arbitrary (uninterpreted) relationships between them, and arrows are (interpreted) *is-a relations*. The important point is that it is possible to map such models to and from lexon-based ontologies, which provides two immediate benefits: a graphical and formally founded notation, and existing tools that already support it, such as Microsoft's VisioModeler for ORM. Because of our earlier experience with this particular method and tools for database design (De Troyer et al, 1995), we have adopted it as a prototypical research and implementation tools and techniques environment for ontology construction.

One then sets up a user profile by choosing the profile view on the toolbar. Remember a user expresses his interests in his profile by specifying a query on the ontology, i.e. as a composition of logical combinations of the desired events, EWN concepts, named entities and categories from the ontology. The resulting implied logical expression will then specify which documents satisfy the profile. This is illustrated in Figure 4.

5. Future work

Although we have now chosen to use a rather simple query language for setting up the user profiles, it is our aim for future work to develop a more sophisticated conceptual query language (for instance similar to RIDL (Verheyen et al,1982)), to specify queries on the ontology.

6. Acknowledgements

This work was supported by the European Commission's IST Project NAMIC (IST-1999-12392). We would also like to acknowledge contributions by our partners in this project Agenzia ANSA S.C.R.A.L., The University of Sheffield, University of Roma Tor Vergata, Universitat Politècnica de Catalunja, Vrije Universiteit Brussel, Comité International des Télécomunications de Presse, Itaca s.r.l., Agencia EFE, S.A. and Financial Times.

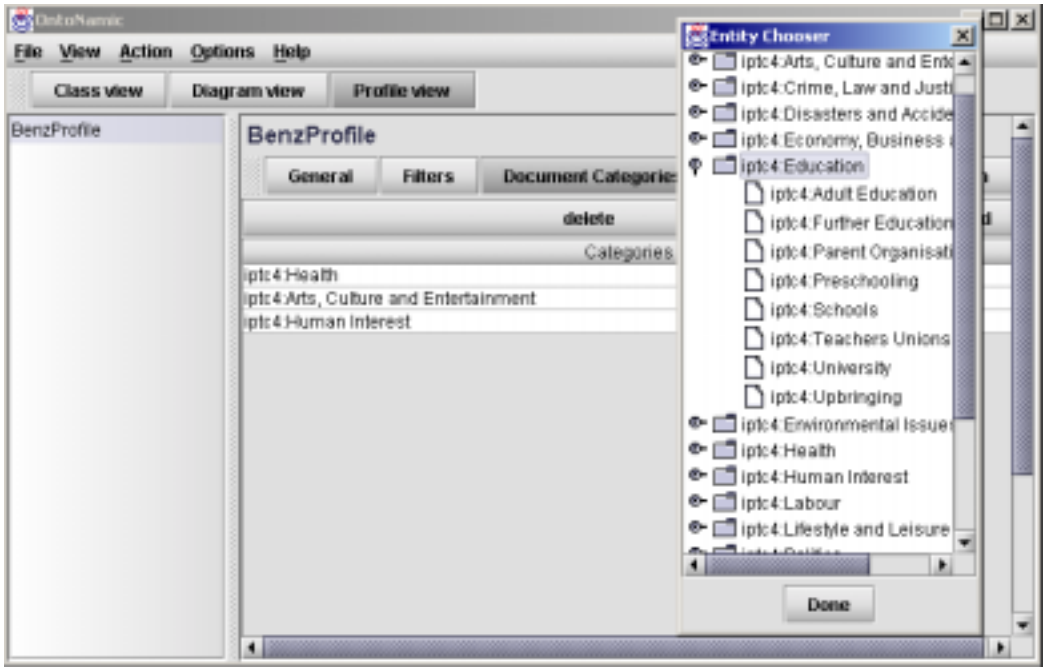


Figure 4

7. References

Abuzir Y. and Vandamme F: "E-Newspaper Classification and Distribution Based on user profiles and Thesaurus" , SSGRR 2002w - International Conference on Advances in Infrastructure for e-Business, e-Education, e-Science, and e- Medicine on the Internet, January 21 - 27, 2002 L'Aquila (Italy).

Abuzir Y., Vervenne D., Kaczmarski D. and Vandamme F.: "E-mail messages classification and user profiling by the use of semantic thesauri", in CIDE 2001 - in Proceedings CIDE 2001 Conference - 4th International Conference on the Electronic Document, Toulouse - FRANCE Oct. 2001.

R. Basili, R. Catizone, L. Padro, M.T. Pazienza, G. Rigau, A. Setzer and N. Webb, Y. Wilks and F. Zanzotto, 2001 Multilingual Authoring: the NAMIC approach, Human Language Technology and Knowledge Management, EACL/ACL Workshop, Toulouse, France

Demey J., Jarrar M., Meersman R., Exchanging ORM Schemas Using a conceptual Markup Language, Submitted to ER2002

De Troyer, O., Meersman, R., 1995 : “A logic Framework for a Semantics of Object Oriented Data Modeling” , in: Proceedings of Entity Relationship and OO Modelling Conference, Papazoglou et al. (eds.) Springer LNCS.

Guarino, N., Masolo, C., and Vetere, G. 1999. OntoSeek: Content-Based Access to the Web. IEEE Intelligent Systems, 14(3): 70-80.

Gruber T., 1995 “Toward principles for the design of ontologies used for knowledge sharing”, International Journal of Human-Computer Studies, 43(5/6).

IPTC, http://www.iptc.org/ -> Subjects -> Subject reference system

Jarrar M., Meersman R., 2002 Practical Ontologies and their Interpretations in Applications - the DOGMA Experiment, Submitted to WWW02

Karp, P. 1992. The Design Space of Frame Knowledge Representation Systems. Technical Report 520, SRI International Artificial Intelligence Center

Noy, N.F., and Musen, M.A. (1999). SMART: Automated Support for Ontology Merging and Alignment. Submitted to the Twelth Workshop on Knowledge Acquisition, Modeling, and Management, 1999. Banff, Canada

Pretschner, A., and Gauch, S. 1999. *Ontology based personalized search*. In Proc. 11th IEEE Intl. Conf. on Tools with Artificial Intelligence, pp. 391--398

Stevenson, M. and Gaizauskas R: Using Corpus-derived Name Lists for Named Entity Recognition.; Applied Natural Langauge Processing and the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL-2000) Seattle, WA

Terry Halpin : Information Modeling and Relational Databases: From Conceptual Analysis to Logical Design. Morgan Kaufmann Publishers, 2001. ISBN 1-55860-672-6

Verheyen, G. and van Bekkum, P.: "NIAM, aN Information Analysis Method", in:IFIP Conference on Comparative Review of Information Systems Methodologies,T.W. Olle, H. Sol, and A. Verrijn-Stuart (eds.), North-Holland (1982).

Vossen P (eds), 1998; EuroWordNet: A Multilingual Database with Lexical Semantic Networks, Kluwer Academic Publishers, Dordrecht (Netherlands)

Description of Events: An Analysis of Keywords and Indexical Names

Khurshid Ahmad, Paulo C F de Oliveira, Pensiri Manomaisupat, Matthew Casey and Tugba Taskaya

Department of Computing, University of Surrey
Guildford, Surrey. GU2 7XH
UK
(k.ahmad@surrey.ac.uk)

Abstract

Event modelling requires a good understanding of the modes used in communicating the events, including natural language, graphs and images. A case study of financial market movement, where text, or news wires, and graphical information, or a financial time series, were correlated, is described. This leads to a need for automatic text classification: a method based on unsupervised neural networks and autonomous assignment of keywords is described. These are preliminary results of an EU 5th Framework Project –GIDA (No. IST 2000-31123). Methods of corpus linguistics and terminology are used to underpin the methods.

1. Introduction

An event is defined as a significant occurrence or happening, or more specifically, as in physics, an event is a phenomenon or occurrence located at a single point in space time. In the late 20th century a tautological compound *news event* makes the meaning of the word 'event' even more explicit. A description of an event names the persons, places, things, or ideas in relation to the significant occurrence, happening or phenomenon: Osama bin Laden is frequently named in relation to terrorism; the Financial Times/Stock Exchange 100 (top companies) index (FTSE) is named in relation to the British, and possibly, EU economy; *relativity* and Einstein in relation to 20th century physics.

Reports of terrorism, stock market movements, and developments in theoretical physics, use written language, photographs, time series of financial transactions, graphs of key variables, and other symbol systems. Reports of events, political, economic, scientific or leisure, for instance, are crafted using a range of semiotic systems – from natural language to images, from time series to icons. An event, when described in natural language, involves the deliberate frequent use, and at times deliberate censoring of names related to the significant occurrence or phenomenon. For a specific event, described over a period of time in a number of texts, some persons, things or ideas are mentioned more or less frequently depending on their influence on the event. An event, perhaps at the lexical level of linguistic description, is a cluster of keywords or terms related to the specific area of human activity – terrorism, finance and commerce, physics, or football for example.

The names of (significant) persons, things or ideas act as an index to an event, an index which has linguistic rendering but can equally be referred through the use of other semes – images, graphs, mathematical symbols, circuit diagrams are some of the other indexical semes. Keywords-in-context (KWIC), largely common nouns sometimes qualified by adjectives, can be used to categorise documents related to a special subjects or, perhaps indirectly, to a specific events.

For us, event modelling requires an understanding of keywords and a collation of indexical names. For computer-based event modelling, involving information extraction and retrieval, and text understanding, it is important (a) to automatically identify and verify new keywords and indexical names, (b) to be able to note nuances of, and changes in, use of the keywords and the indexical names, and (c) to correlate the information in text and in graphs through the use of indexical names and keywords.

News streams provided by organisations like Reuters or Bloomberg comprise a range of keywords and indexical names that may change from one news item to the next; an event modeller will need to filter the news from such a diverse information resource. Specialist information providers deliver not only news texts but also supply, for example, time series of changes in value of stocks, shares, currencies, bonds and other financial instruments.

We have a narrower focus than other authors in information extraction (see for example Gaizauskas et al, 1995 and Maybury et al, 1995) in that we are looking for changes in key financial instruments that are reported in financial news-wires. The news coverage of these instruments is of two types: first, there is a daily report about changes in the value (numerical) of the instruments for instance, one can see time series comprising historic data about the changes in values of currencies; second, the manner in which the value of the instruments changes depends on the reports relating, directly or indirectly, to the instrument. The reports, for example, about war or economic uplift/downturn, affect the value of the instruments. Some authors claim that there is a correlation between 'good' or 'bad' news relating to the instrument and its potential numerical value. In Section 2 we take this discussion further.

The news report is one of the most commonly occurring linguistic expressions. Despite being a good example of open-world data, a news report is a contrived artefact: each report has a potentially attention grabbing headline; the opening few sentences generally comprise a good summary of the contents of the report; there are *slots* for the date of origin and slots for photographs and other graphic material. This contrived artefact is highly focused and highly perishable, and usually contains references to one or more persons, places, events or actions. Automatic categorisation of news stories is of substantial interest to in a range of applications (Mani 1998) to information retrieval communities, and to major news vendors

supplying *on-line news*; Section 3 takes up this story further and we conclude in Section 4.

2. Keyword and Indexical Name Correlation

Generally, information is delivered to financial market operatives via electronic mail, newspaper, or company announcement briefings or company annual reports. Whatever its source, the information in the news is an important component in making investment decisions (Figure 1). Equally important are events like natural disasters or terrorist activities for example.

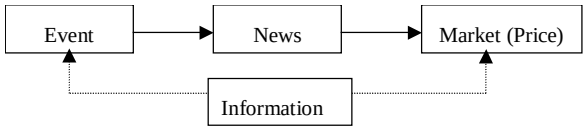


Figure 1: The relationship between Events, News and Markets (price) through Information.

For example, the terrible events of September 11, 2001, have had a catastrophic effect on financial markets world wide (See Figure 2). Various national economic indicators –indexical names – show the reaction on the date; there has been a decline in the value of these indices before that date and indeed a resurgence in the value afterwards.

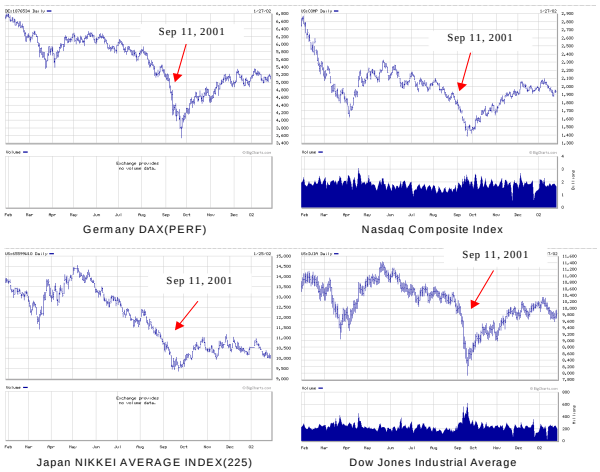


Figure 2: Movement from Feb 2002 to Jan 2002. Note the dip on and around Sep 11th 2001.

According to the Dow-Hamilton Theory (Rhea 1994), there are three kinds of price movements or market movements: (i) *Primary movement* which lasts from few months to many years and represents the broad trend within a market; (ii) *Secondary movement* last from a few weeks to few months and may sometimes be contrary to the primary movement; and, (iii) *Daily Fluctuations* can move with or against the primary trend and exist for a few hours to a few days.

2.1. Market Movement and Market Sentiment

Our work, sponsored in part by the EU-sponsored GIDA project (Project No. IST 2000-31123), focuses on primary movements. We report on some initial work that attempts to changes in an index, FTSE100, with changes in 'market sentiment' as expressed in news reports about the UK economy specifically and reports about the Wall Street indices. The later has substantial influence on the UK economy. Financial analysts use sophisticated political, economic and psychological analysis to determine the reaction of market operatives and to predict the possible trading decisions of the operatives. Reports related to the sentiment use a range of metaphors to express the state of a market and its possible movements. Francis Knowles has written about the use of *health metaphors* used in the financial news reports: markets are full of *vigour* and are *strong* or the markets are *anaemic* or are *weak* (1996); most newspapers also use *animal* metaphors – there are *bull* markets and *bear* markets, the former refer to expansion, and indirectly to fertility, and the later to shy, retiring and grizzly behaviour much like that reported about bears in popular press and in literature for children. Indeed, there are fairly literal words that express the sentiment, as reported in the news wires, about the markets: financial instruments *rise*, *fall*, markets *boom*, *go bust*, and there are *gains*, *losses* within the markets, economies *slowdown*, suffer *downturns*, whole industry sectors maybe *hardpressed*. Table 1 contains examples of good and bad news in a typical Reuters news stream:

Mainly Good News Stories	Rather Bad News Stories
Naval shipbuilder and military contractor Vosper Thornycroft has **boosted** its civil arm by buying facilities manager Merlin Communications (Nov 14, 2001)	Heavyweight banking and oil stocks have **dropped** up the leading share index as investors bet on fresh interest rate cuts.' (Nov 21, 2001).
The FTSE 100 stock index looks set to open **stronger** today after Wall Street added to **gains** seen at the London close and with U.S. stock index futures boosted by rumours that Osama bin Laden had been captured.'(Nov 15, 2001).	The European Commission has **slashed** its official growth forecasts for the euro zone [..], predicting the most serious **slowdown** since the 1990s recession, with **lower growth** in 2002 than this year.' (Nov 21, 2001).
Builder McCarthy & Stone has posted a 13 percent **rise** in annual pre-tax profits, built on **strong** sale prices for its retirement homes [..], but cautions that the **boom** may be over.(Nov 15 2001).	The FTSE 100 fell today, amid concern about how the U.S. economic **downturn** will hurt technology stocks and British Airways' operations. (Dec 10, 2001).
Leading shares are expected to **rise** again after Wall Street steamed **higher** overnight and the market basked in a feel-good glow, dealers said.' Nov 14, 2001).	Britain's economy appears to be sailing along relatively smoothly despite the global **slowdown** and a string of high-profile job **layoffs** (Oct 22, 2001).
'Leading shares have edged **higher** in early trade, **boosted** by **gains** in technology stocks in response to a Wall Street rally and **positive** expectations for the economic outlook.' (Jan 4, 2002).	'The **hard-pressed** manufacturing sector has recorded its biggest monthly production **drop** in almost a decade, **sinking deeper** into **recession**. (Nov 5, 2001).

Table 1. Examples of 'good' and 'bad' news stories in Reuters News Wires (Oct 2001-January 2002)

The above table contains examples of how the market is moving. But here we have free natural language complete with ambiguity and nuances of meaning: so there maybe a 'rise in profits' and a 'strong sale prices', in the story about builder's McCarthy & Stone above, both phrases suggesting that this is a good news story, except for the last sentence suggesting that 'boom maybe over'. Nevertheless, many of the news items do not change the nuance of the story by such highly temperate notes.

2.2. Correlating Sentiment and Market Indices

We created a corpus of 1,539 English financial texts from one source (Reuters) on the World Wide Web, published during a 3 month period (Oct 2001-January 2002) comprising over 310,000 tokens. The corpus comprised a blend of both short news stories and financial reports. Most of the news is business news from Britain with thirty percent of the news is from Europe and from the United Stages.

We found over 70 terms each for conveying good news and bad news in the above corpus. The texts in our corpus were also time stamped, and by using our text and terminology management system, System Quirk, we computed the cumulative weekly frequency of *good* words and *bad* words during one month – November 2001. The 'week' is a working week comprising 5 days, Mondays-Fridays:

Time (5 day Week)	Good Word Frequency	Bad Word Frequency
1	58	40
2	71	***75***
3	***77***	66
4	73	59
5	72	28
Total	**351**	**268**

Table 2: Frequency of Good and Bad words in Nov 2001. The underlined figures in the 2nd and 3rd columns indicate the minimum value of the frequency and the numbers in italics are the maximum value.

Table 2 shows that in November the highest frequency of 'good' words was in week 3 (77 instances) and the 'bad' words was in week 2 (75 instances). How does this correlate with the movements of the London stocks and shares as expressed by the FTSE 100? Figure 3 provides an example of the correlation between the frequency of 'good' words from news in November in our corpus and close prices of FTSE100 Index for the whole month of November.

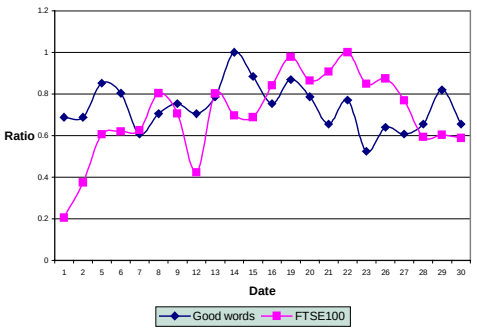


Figure 3: Market correlation between 'good' word frequency and FTSE index.

The highest value of the FTSE 100 index was on on 22 November 2001 (5345.94). There is a perhaps a correlation between the changes in the value of the index and the frequency of 'good' words: Positive gradient in the 'good' words time series correlates well with the positive gradient in the FTSE 100 index values. What will be interesting for the purposes of predicting the movement of the market, will be a correlation that suggests that a rise in the number of good words one day nudges the market . Correspondingly, that a decrease in the number the previous day will lead either to a static market or falling market the next day. The same can be said, perhaps in reverse, about the bad news words (see Figure 4).

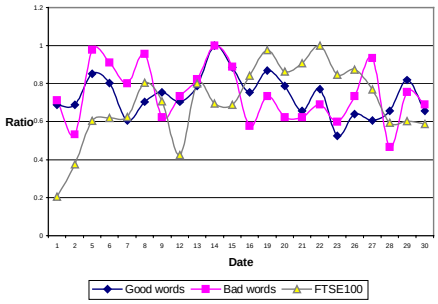


Figure 4: Good and bad word frequency correlated with FTSE 100.

Figure 4 shows 'good' word and 'bad' word frequency is perhaps correlated with FTSE100 values. For example, from 23rd to 29th November the frequency of 'bad' words increased while the FTSE100 went down over this period. After the 29th November, the FTSE100 value slightly increased as the 'good' word frequency also increased.

The above analysis and the concomitant results are of a tentative nature in that work is progressing in three major directions. First, one needs a bigger corpus, and a longer time series, to be more assertive about a correlation between an index and the corresponding sentiment-bearing terms. Second, further analysis is underway to note that the good news is sometimes tempered with bad news and vice versa – this will involve a phrasal or sentential analysis. Third, the notion of a 'time series' is a carefully defined concept for a series of cardinal numbers collected at discrete intervals of time or collected continuously; we are exploring the status of a time series made up of counts of lexical strings found in a news report that may have been produced over an approximate time. Nevertheless, Figures 3 and 4 show how a news stream, comprising subject specific texts, can be visualised especially in the context other indexical names.

3. Classifying News Wires Using Keywords

3.1. Categories of News Reports

A news stream comprises news stories that: (a) range over a whole range of subjects; (b) the news may emanate from or maybe about a nation state; and (c) the news may be focused on a certain specific area of human enterprise Reuters labels for items (a)-(c) are 'Topic', 'Country' and 'Industry'; these labels are used by Reuters' sub-editors to tag each news story with one or more Topic and Country tags, and in some cases with the Industry tags. These pre-assigned tags, about 1000 different tags in all, can, in principle, be used to categorise individual news stories in a news stream. However, the plurality of tags, that is the presence of one or more tags with either Topic, Country,

or Industry, makes such a categorisation more complex. Before we discuss how to deal with such a complex categorization task (see Section 3.5), which is possibly subjective in that the categories are based on an ontology which was created by Reuters themselves, we look at how to categorize texts based on (semi-)automatically extracted keywords.

One well-recognized way of describing news reports is to classify the texts as a distinct *register* or *genre* of writing. The term register is used to indicate that the language within a specialized field differs from that of *general language* or language of everyday use, at lexical, syntactic and semantic levels. A large collection of general language text may thus be contrasted with a set of specialist reports at various linguistic levels, including lexical and semantic.

An important use of this contrast is in a method of semi-automatically identifying the terms of a set of specialist domains. This method involves comparing the frequency of systematic terms in a collection of specialist texts sometimes called a *corpus*, with the frequency (or absence) of the terms in a carefully compiled corpus of general language texts. Each term can be construed as a dimension in a vector space and the presence or absence of a term within a text is then used to allocate the text its position within the vector space. There is some evidence from work in linguistics that word categories (nouns, verbs, adjectives, adverbs, prepositions, etc.) may be inferred from the statistical occurrences of words in different contexts. For Kohonen and his colleagues, '"*context patterns*" consist of groups of contiguous symbols'; the authors cite pairs or triplets of words in a sentence as an example of such patterns. Such *pairs* or *triplets* are then used as inputs in the training and testing of a neural network (the so-called self-organising feature maps or SOFM; details of this map is presented in the next Section 3.2). Kohonen has shown that a SOFM-trained word context pairs, derived from 10,000 random sentences, shows 'a meaningful geometric order of the various word categories'. A larger SOFM, the WEBSOM has been variously described by Kohonen as a *scheme, content-addressable memory, method* and *architecture*. WEBSOM is a two-level self-organising feature map comprising a word category map and a document category map, which has been used to classify newsgroup discussions, full-text data and articles in scientific journals (Kohonen 1997b, Kaski *et al*. 1996). Terms were pre-selected by the builders of WEBSOM. There are other neural network architectures that have been used in text categorisation, especially the widely-used supervised learning algorithms – SOFM is based on unsupervised learning algorithm – which have been discussed by Lewis (1995).

Consider a set of texts that may have been selected according to certain criteria: for instance, all texts streaming along a news wire over a short period of time comprising news related to specialist topics – like environmental news or economic news. Such a short news stream may contain may result in a text collection, or if collected systematically, a text corpus, that may be characterised the high frequency of environment – or economics – related terms. However, over a long period of time this may not be the case as the news stream may start to deliver texts in different specialist areas. So how do we extract terms from such a corpus?

Specialist texts can be distinguished from a general language text at the lexical level of linguistic descriptions by looking at the ratio of relative frequency of a linguistic token in a specialist text and its frequency in general language texts. This ratio has been termed *weirdness* to indicate how it measures the preponderance of words in specialist texts that would be unusual in general language, (see, for example, Ahmad 1995).

Typically, before text documents are represented as vectors in order to act as the input to a text categorisation system, pre-processing takes the form of filters to remove words 'low in content' from the text (see the WEBSOM method in Kaski et al 1996). We remove punctuation, numerical expressions and *closed-class words* as a precursor of generating the feature set. Vectors representing news texts were created on the basis of a lexical profile of the training set of texts. This lexical profile was determined by two measures: the frequency of a term; and, a weirdness coefficient describing the subject-specificity of a term.

The feature set was created by first selecting the top 5% most frequently occurring words, and from this set, by choosing the words with the highest weirdness coefficient. Subsequently, the 50 most frequent words are selected, excluding spelling mistakes, and numerical expressions and terms too infrequent to provide consistency within a domain are avoided. A high value for the weirdness coefficient is indicative of a word which is uncommon in general language but common in the specialist corpus under examination and is thus a good candidate for a domain term or other word specific to that genre. By disregarding words with a weirdness coefficient lower than a threshold, many *closed-class words* and other terms common in general language are automatically removed. Before we show texts can be categorised using the above method, we digress to briefly outline the Kohonen Self-organising Maps

3.2. Kohonen Self-organising Maps

A SOFM is a neural network and associated learning algorithm that is designed to produce a statistical approximation of the input space by mapping an input in to a two-dimensional output layer (see Kohonen 1997a for an extensive discussion). The approximation is achieved by selection of features that characterise the data, which are output in a topologically ordered map. The Kohonen Self-Organising Map has a close resonance with the *k-means clustering* method, with the additional constraint that cluster centres are located on a regular grid (or some other topographic structure). Furthermore their location on the grid is monotonically related to the pair-wise proximity (Murtagh & Hernández-Pajares, 1995).

The basic SOFM consists of a single layer of neurons formed into a two-dimensional lattice. Each neuron is connected to the input via a set of connections utilising connection weights, just as in a perceptron. There is no 'output' of the map, rather the values of each neuron's weight vector are used to visualise the formed topological ordering. The weight vectors form a cluster prototype that is measured against each input to determine how 'close' the vector is to a given cluster. Since the map is two-dimensional and the input typically has a high dimensionality, the SOFM acts as a dimensional squash

allowing the visualisation of features within multi-dimensional data.

Learning is achieved in the SOFM using a competitive algorithm. The Euclidean distance between each training input vector and all weight vectors is determined. The neuron with the weight vector that has the smallest Euclidean distance to the input pattern is termed the winner. To reward the winning neuron its weight vector is adjusted to be 'closer' to the input vector, with the amount of adjustment determined by the number of times the training patterns have been presented (via the learning rate). Additionally, all vectors within a defined neighbourhood of the winner are adjusted, essentially forming a cluster of similar values that are seen to be activated by the winner. The neighbourhood size decreases with the number of training cycles, typically using a bubble neighbourhood (a rectangular area) or a Gaussian neighbourhood, both centred on the winning neuron. The adjustment of the weight vector towards the input is achieved by effectively 'moving' the weight vector's direction towards that of the input. This simple process of adjusting ever-smaller neighbourhoods of winners allows the formation of clusters within the lattice. As the number of cycles increases, the clusters become more stable and can be viewed through probing to find winners using test data.

The principal way in which information about the clustering performed by the SOFM learning algorithm is visualised is through probing with a test set to find the winning neurons. The co-location of different winners from different categories highlights the similarity between clusters. The effectiveness of such clusters can be measured by comparing different versions of the map trained on the same data through a technique being developed by Ahmad et al (2001), where Fisher's Linear Discriminant Rule is used to quantify the discrimination ability of different clusters.

3.3. Limitations of a SOFM

The SOFMs strength lies in its ability to *statistically* summarise the input space. However, it has been shown that the basic SOFM does not always produce a *faithful* approximation (Ritter & Schulten, 1986). This faithful approximation is defined as the proportionality between the density of the weight vectors and the density of the input space. Lin et al (1997) has shown that the SOFM underrepresents high-density regions and overrepresents low-density regions.

3.4. Automatic Categorization of Texts Based on Keywords Using an SOFM

Our text corpus consisted of 100 Associated Press (AP) news wires selected from 10 pre-classified news categories shown in Table 3 together with their icons. The average length of the articles was 622 words.

Text Categories

1	Bioconversion		**6**	Exportation of Industry	
2	Pollution Recovery		**7**	Foreign Trade	
3	Alternative Fuels		**8**	Int. Drug Enforcement	
4	Fossil Fuels		**9**	Foreign Car Makers	
5	Rain Forests		**10**	Worldwide Tax Sources	

Table 3: Text categories used in the TIPSTER – SUMMARY program

The 100 AP news wires comprised over 56,000 words. System Quirk was used to compute frequency distribution of words in the AP News wire corpus. The System also has access to the frequency distribution of words in the British National Corpus (Aston and Burnard 1998) a carefully compiled general language corpus. Some of the high weirdness terms, e.g., *drug*, *taxes*, *pollution* and *environmental* are important keywords, but the same cannot be said for 'terms' like *billion*, *percent* and *federal*. Usually, proper nouns are also flagged as terms by this method. The feature words identified for the 100 AP News Wire texts are shown in Table 4 according to rank:

1	percent	15	congress	28	dioxide	41	corp
2	tax	16	mexico	29	marine	42	forests
3	billion	17	emissions	30	mazda	43	cocaine
4	drug	18	drugs	31	gases	44	enforcement
5	reagan	19	fuels	32	shale	45	warming
6	cars	20	senate	33	deficit	46	smog
7	taxes	21	auto	34	export	47	ozone
8	environmental	22	proposal	35	recycling	48	Massachu-setts
9	pollution	23	gasoline	36	epa	49	imports
10	fuel	24	exports	37	honda	50	automobile
12	federal	25	vehicles	38	methanol	51	trafficking
13	dukakis	26	ohio	39	automakers		
14	bush	27	green-house	40	panama		

Table 4: Feature words identified for the 100 AP News Wire Texts.

Having identified the feature set the training vectors for each of the texts could then be generated. Each vector consisted of binary values indicating the presence or not of each of the feature words determined above.

We have developed a system for creating Kohonen Feature Maps (SANC: Surrey Artificial Network Classifier). The system, after having trained an SOFM, is also capable of testing it. (There are facilities to vary the key parameters associated with the learning algorithm). The system can be used to test the trained. Furthermore, the system allows the storage of previously trained maps for reference purposes (Ahmad, Vrusias and Ledford 2001).

The results of the Kohonen classifications for full texts are shown in Figure 5. Using symbols to represent each of the locations of the 'winning node', the position of each text is indicated across the two-dimensional map (shown in Table 3). It can be seen that the quality of clustering for the full-texts is successful for a range of categories, but especially for categories 9 (FOREIGN CAR MAKERS) and 10

(WORLDWIDE TAX SOURCES). Patterns in categories 1 (BIOCONVERSION), 4 (FOSSIL FUELS), 6 (EXPORTATION OF INDUSTRY) and 8 (INTERNATIONAL DRUG ENFORCEMENT) are also effectively grouped together. The widespread distribution of Class 5 (RAIN FORESTS) shows it to be the worst class on the map.

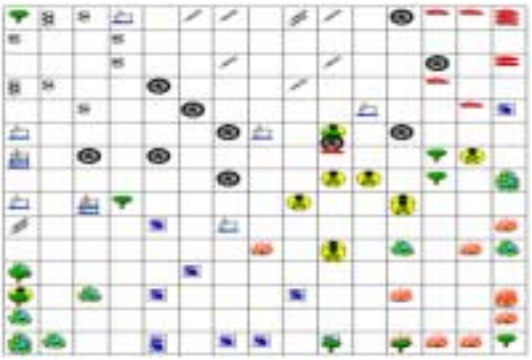


Figure 5: Results of a Full Text Map trained using exponentially decreased neighbourhood and learning rate.

These results for the trained Kohonen map were similar across a number of trials despite variations in training method and learning rate used. Some categories, for example 10 (WORLDWIDE TAX SOURCES), clustered consistently better than others for instance 5 (RAIN FORESTS). By simply counting the number of feature set words that appear in at least nine of the ten texts of each category, the best clustered categories are guaranteed to have some of these words. This reflects the tendency of these categories to cluster well. On the other hand, for a category in the 'best' case, only four of the texts share a common feature set word. This difference in classification difficulty was also seen in the TIPSTER results from two human assessors.

3.5. Multiple Categories and Text Categorization

Recall that Reuters News Agency has three categories: "Topic", "Country" and "Industry". The total number of different tags, or concepts, defined in these three categories is approximately 1000.

We have created a text corpus of 800 news stories streamed by Reuters in 1997. Each of the news stories is encoded in XML format and has clearly delineated headline, date, writer, text and code fields using XML tagset. The XML-based delineation helps in extracting keywords associated with the Topic, Country and Industry tags. The frequency of each concept was calculated within 800 documents; 80 of the keywords turned out be more frequent than other 920: the distribution of the keywords in the various fields was as follows:

Industry	**39**	***Topic***	**32**	***Country***	**19**

A SOFM was trained for categorising the 102 out of the 800 news stories. The input vector was created from the 80 most frequent keywords associated with the triple, Industry-Topic-Country: the absence and presence of a particular keyword was used to create the input vector for each of the texts. The neural network was trained 100 times. The vector thus created can, in principle, cope with upto 39 different categories of 'Industry', of 32 different 'Topics' and '19' different countries. The downside here is that documents comprising references to the 920 keywords may not get classified as well as those that may comprise the 80 categories used in the construction of the input vector.

After the training period, the pre-specified Reuters documents were visualised on the map. As can be seen in Figure 6, the documents associated with each neuron were represented by a blue square. The distribution and the similarity of the documents were based mostly on the "Topic". On the lower right side of the map, the topics related to "Government/Social" were clustered. The subtopics of "Government/Social", for example "Sports" and "Art", were also clustered near this area. The documents categorised as "Management" were found on the lower left corner of the map. "Strategy and Plans", "Comments/Forecast" and "Economy" follow this as we approach the upper left corner. "European Community" documents were found on the upper right corner of the map.

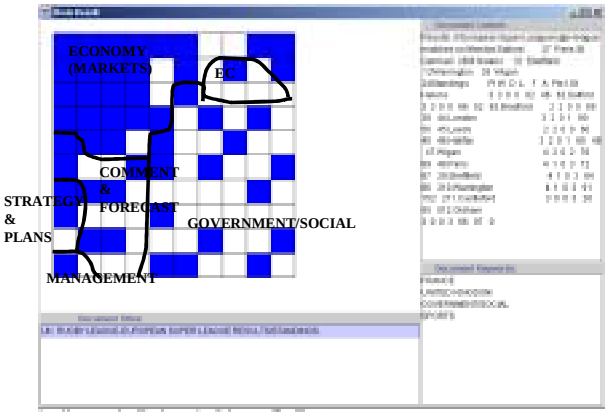


Figure 6: A Categorisation of Reuters news stories using pre-specified category information.

4. Afterword

Our current work involves evaluating the categorisation produced by the method that relies on different distribution of specialist terms in special and general language texts with that of using networks to classify texts that have pre-specified category information as was the case just described.

The pre-specified categories appear to be complex and, as mentioned above subjective in nature. We are currently examining whether a summary of text may give us some indication of the category. The reasoning is as follows: a full news story may contain extraneous material and a good summary will eliminate sentences within the text that are not directly related to the category or categories. Lexical cohesion studies have shown that keywords form the glue that helps to create a cohesive and coherent texts (Hoey 1991). In our previous work on AP news wires (Ahmad, Vrusias and Ledford 2001) we looked at three different types of text streams – headlines only, news summaries and full news items and categorised these texts using self-organising feature maps (SOFM). We found that an SOFM trained on vectors related to summaries

only provides a fairly accurate cluster when compared with vectors related to full text. This work is currently being carried out on the 102 Reuters texts mentioned above.

An analysis shows a vector for the 102 texts using our method based on the weirdness of the keywords within the news stories (Table 5).

Element	Description	Words	
1 – 25	*Single Words* Top 25 simple words with high weirdness and high frequency	inventories, yen analysts, merger cents, peso investors, exports quarterly, forecast pesos, shares dealers	analyst directive billion, soccer traded, allegations trading, stocks fiscal, tobacco nickel, earnings
26 – 30	*Compound Words:* 5 most frequent compound words	shareholder newsroom worldwide	online chairman
31 – 40	*Proper Nouns* 10 proper nouns with high weirdness and high frequency	dorfman compuserve novell aol microsoft	ec kimberly chrysler saudi netherlands
41 – 45	*Movement Indicators:* 5 most frequent downtrend words.	lost fall falling	risk losses
46 – 50	*Movement Indicators:* 5 most frequent up trend words.	up growth high	added strong

Table 5: Vector for the 102 Reuters news items (c.1997)

Note that in the above vector we have included movement indicators, proper nouns and compound words together with the single word terms. The 30 keywords and 10 proper nouns/indexical terms, together with 10 movement indicators will help us to define an event. Initial results of this analysis are encouraging in that we obtain the major clusters much like as found in Figure 6

We are currently exploring the notion that news streams will be filtered by using a trained Kohonen SOFM and the filtered text will be used to study market movement. The filter has to be 'cleaned' in that news stories are perishable items with constantly changing subjects – one idea is to re-train the network everyday, towards the end of the day perhaps, with a fixed number of stories which will exclude the very first day of the previous training set and include yesterday's news stories.

Event modelling, especially in noisy and dynamic environments, requires a careful consideration of the key concepts, expressed as keywords, and of indexicals like persons, places, things or ideas which play a crucial role in turning an occurrence, happening or phenomenon into a significant one.

References

Ahmad, K., Vrusias, L. & Ledford, A. (2001). Choosing Feature Sets for Training and Testing Self-organising Maps: A Case Study. Neural Computing and Applications, 10(1), 56-66.

Ahmad, K. Pragmatics of Specialist Terms and Terminology Management. (1995). In (Ed.) Petra Steffens. Machine Translation and the Lexicon. pp. 51-76. Heidelberg: Springer.

Aston, G. and Burnard, L. The BNC Handbook: Exploring the British National Corpus with SARA. 1998. Edinburgh: Edinburgh University Press.

Gaizauskas, R., Wakao, T., Humphreys, K., Cunningham, H. & Wilks, Y. (1995). Description of the LaSIE system as used for MUC-6. In Proceedings of the Sixth Message Understanding Conference (MUC-6). Morgan Kaufmann.

Hoey, M. (1991). Patterns of Lexis in Text. Oxford: Oxford University Press.

Kaski, S, Honkela, T, Lagus, K & Kohonen T. (1996). Creating an order in digital libraries with self-organising maps. In Proc. WCNN'96, World Congress on Neural Networks, 1996, pp 814-817. Lawrence Erlbaum and INNS Press.

Knowles, F. (1996) Lexicographical Aspects of Health Metaphors in Financial Texts. In (Eds.)Martin Gellerstam et al. Euralex'96 Proceedings (Part II). Göteborg, Sweden: Göteborg University. pp 789-796.

Kohonen, T. (1997a). Exploration of very large databases by self-organizing maps. In Proceedings of ICNN'97, 1997, pp. PL1-PL6, IEEE Service Center, Piscataway, NJ.

Kohonen, T. (1997b). Self-Organizing Maps. 2nd Ed. Berlin, Heidelberg, New York: Springer-Verlag.

Lewis, DD. (1995). Evaluating and optimising autonomous text classification systems. In SIGIR 95: Proc. of the 18th Annual ACM-SIGIR Conference on Research and Developments in Information Retrieval. pp 246-254.

Lin, J.K., Grier, D.G. & Cowan, J.D. (1997). Faithful Representation of Separable Distributions. Neural Computation, 9(6), 1305-1320.

Mani, I. (1998) The TIPSTER SUMMAC Text Summarization Evaluation. Mitre Technical Report: MTR 98W0000138, 1998.

Maybury (1995). Generating Summaries from Event Data. Information Processing and Management. 31(5), 733-751.

Murtagh F., Hernández-Pajares M. (1995). The Kohonen Self-Organizing Map Method: An Assesment. Journal of Classification, 12, 165-190.

Rhea, R. (1994). The Dow Theory. Burlington: Fraser Publishing Company.

Ritter, H. & Schulten, K. (1986). On the Stationary State of Kohonen's Self-Organizing Sensory Mapping. Biological Cybernetics, 54, 99-106.

Learning IE patterns: a terminology extraction perspective

Roberto Basili, Maria Teresa Pazienza, Fabio Massimo Zanzotto

University of Rome Tor Vergata,
Department of Computer Science, Systems and Production,
00133 Roma (Italy),
{basili, pazienza, zanzotto}@info.uniroma2.it

Abstract

The large-scale applicability of knowledge-based information access systems such as the ones based on Information Extraction techniques strongly depends on the possibility of automatically acquiring the large amount of knowledge required. However, the basic assumption of the IE paradigm, i.e. that the information need is known in advance, limits inherently its applicability since the resulting IE pattern learning algorithms are not generally conceived for the analysis of large corpora if not driven by a specific information need. Since in the terminological studies the corpora and not the information needs already drive the extraction of the knowledge, they offer many insights and mechanisms to automatically model the knowledge content of a coherent text collection. In this paper, we will present a terminological perspective to the acquisition of IE patterns based on a novel algorithm for estimating the domain relevance of the relations among domain concepts. The algorithm and the representation space will be presented. Before starting the discussion, however, we will describe the overall process of building a domain ontology out from a extensional domain model (i.e. the collected domain corpus). Finally, the results of the application of the algorithm over a large domain corpus will be presented and the resulting ontology is discussed.

1. Introduction

The large-scale applicability of knowledge-based information access systems such as the ones based on Information Extraction techniques strongly depends on the possibility of automatically acquiring the large amount of knowledge required. The applicability of these systems over large heterogeneous text collections (e.g. the World Wide Web) may be one of the keys of success of "emerging" information access paradigm such as the Question Answering (QA) and the Automatic Summarisation (AS). In fact, the major strength of the Information Retrieval engines (typically based on the "poor" abstraction of stem) is due more to their wide applicability than to their actual retrieval performances.

A very well assessed approach to Information Access is the paradigm of Information Extraction (MUC-7, 1997; Pazienza, 1997). This latter gave the fertile area where a number of techniques for the automatic acquisition of knowledge have been proposed. However, these learning approaches are focused on the extraction of knowledge needed for the satisfaction of a particular information need (i.e. the one expressed by the template) as the IE paradigm imposes. Therefore, the resulting learning approaches are biased by the fact that they can rely on two important hypothesis limiting their search space. From the one side, the target knowledge domain is generally small and, from the other side, the target information need is very narrow (such as missile launch event in one of the MUC conference). Therefore, the size of the resulting ontology can be kept controlled and the scope of the learning algorithms is a controlled (and small) corpus. In fact, in unsupervised learning techniques as in (Yangarber, 2001; Riloff and Jones, 1999), texts are firstly classified according to their relevance with respect to the particular information need and then particular surface forms somehow related are extracted and retained. The first step narrows the corpus that is given to the second.

However, the basic assumption, i.e. that the information need is known in advance, limits the applicability of the IE paradigm and of the resulting IE pattern learning algorithms. In fact, these latter are not generally conceived for the analysis of large corpora if not driven by a specific information need. If the goal to be achieved is the applicability in large, a different approach has to be undertaken. In such a perspective, the final information needs can not drive the learning phase that should totally rely on the corpus that has to be the source of this information, i.e. it is the final source of information that should suggest the information needs that can be satisfied. This is the typical case a information access system has to face when exposed to an uncontrolled information scenario (e.g. the Web).

Since in the terminological studies the corpus is already the major source of knowledge, they offer many insights and mechanisms to automatically model the knowledge content of a coherent text collection. Here, in fact, the corpus plays the central role of extensional model for the target domain where a domain ontology (i.e. a thesaurus) is extracted from. In this latter, terms and relations among them are generally described. The "operational" notion of *term*, i.e. that the term is the surface representation of a domain concept, allows to define two different levels of analysis: the notion of *admissible surface forms* and the notion of *domain relevance*. The target is generally the extraction of concepts conveyed by nominal phrases and the investigated relations are IS-A and PART-OF. Neverthless this terminological perspective to the extraction of IE patterns can be adopted for widening the applicability. IE patterns may be considered as domain relations among specific concepts, i.e. typical concepts of the domain and named entity classes that hold by definition the special status of domain concepts.

In this paper, we will present a novel algorithm for estimating the domain relevance of the relations among domain concepts. As for the term, the application of a terminolog-

ical approach to the problem of the discovering the domain relations among concept has to establish:

- which are the surface representations of the target relations;
- which is the estimator of the "domain importance" for the discovered relations.

The algorithm and the representation space will be presented in Sec. 4.. Before starting the discussion, however, we will describe the overall process of building a domain ontology out from a extensional domain model (i.e. the collected domain corpus) in Sec. 2. Finally, the results of the application of the algorithm over a large domain corpus will be presented and the resulting ontology discussed (Sec. 5.).

2. Building an ontology for a large-scale IE system

A large-scale IE system for a news agency should be able to scan news streams. The activity of building the needed knowledge base is therefore a huge task. However, in our opinion, this may be undertaken using some insight given by the terminology extraction practice. News streams are, in fact, coupled with a news classification scheme that can be more or less complex (cf. IPTC standards (IPTC,)). This rough or fine-grained classification over the news items allows the definition of coherent knowledge areas over which terminology extraction techniques can be helpful. Each collection of news items belonging to a class is in fact the extensional model for the underlying domain according to the classifiers.

The process of the knowledge modelling is sketched in the following. Given the corpus as model for the knowledge domain (or class) under investigation, the activities that have to be carried out for building the domain ontology are the following:

1. the definition of the named entity classes
2. a first analysis of the corpus for the acquisition of the most important concepts and relations among the concepts
3. the analysis of the extracted domain knowledge for the definition of the top "event" classes
4. the extraction of all the important concepts and relations among the concepts and their clustering under the defined event classes

For the activities 2 to 4, terminology extraction practice may be very useful with the notions of *admissible surface forms* and of *domain relevance*. The latter is a key notion that helps in showing to the ontology builder only the most relevant IE patterns (a combination of the domain concepts and domain relations). These patterns sorted according the domain relevance estimated by the importance function can drive the definition of the top event classes. The event classes elsewhere referred as "template types" will represent the knowledge the final IE system is able to make explicit over the particular domain. Finally, since IE patterns are ranked according to their importance, in the activity of clustering this guarantees that the most important events (and generally the most frequent) may be captured by the resulting IE system.

The attention on the clustering activity is somehow one of the major difference between the construction of a domain ontology for an IE system and the one of a terminological knowledge base (TKB) (or thesaurus). This is mainly because of the nature of the typical target knowledge domains. Terminology extraction is mainly conceived for giving a systematic representation of scientific or technological knowledge domains where certain terms are stable and a relatively small number of surface forms are used to convey a domain concept. On the other hand, in the news streams (the areas in which IE system has to find the information) domain concepts and, more often, domain relations are generally conveyed by more than one surface form. It is the equivalence between different event prototypes, i.e. prototypes that specifies the possible instances of the "Who? Where? What? When? Why?" events, that may make the difference.

3. Domain relations among concepts as event prototypes

Event prototypes (or IE patterns) used by IE systems to perform the activity of extracting information are very similar to what a domain relation among domain concepts may look like. Given for instance the financial domain, the prototype necessary to extract a *"sell event"* from the following news items:

Example 1 *Financial news excerpts*

(a) Eon, the German utility formed by the merger of Veba and Viag, is poised to sell its electronics arm to an Anglo-American consortium for about $2.3bn.

(b) It is understood to be near a deal to sell the Longview smelter for $150m to McCook Metals.

may have the following form:

Example 2 *Sell event prototype*

```
sell(  (agent:companyNE),
       (patient:object),
       (to:companyNE),
       (for:currencyNE))
```

i.e. a company typically sells something to a company for a certain amount of money (`currencyNE`). Here, the two named entity categories, `companyNE` and `currencyNE`, are typical concepts of the financial domain and the showed event prototype is a typical domain relation among these concepts.

Due to the difference on the perspective and on the application domain, some adjustments of the techniques developed in terminology extraction are mandatory in the IE pattern extraction problem. As suggested in the example, in IE, a major role is played by named entities. They are not important as surface forms but as generalised forms (i.e. their category). This is a major difference with the general terminology extraction where named entities are important as instances. For instance, *Newton's law* and *Zipf's law* convey very different meaning and are relevant as such and not in a generalised form `personNE's law`. The adoption of TE techniques on the IE tasks requires that named entity categories are considered as typical concepts of the domain. Admissible surface forms also consider the possibility of selecting forms with named entities (e.g. `companyNE_share` where `companyNE` is a named entity category that may be used for detecting *IBM shares* in target text).

Furthermore, in the IE perspective, the definition and the extraction of the domain relations plays a major role. Such a problem is generally neglected in the TE studies because major efforts are spent in the definition of algorithm for extracting and using catalogues for the general relations among terms such as IS-A or PART-OF (Morin, 1999; CON, 1998). The resulting methods are not suitable for the extraction of domain relations.

In order to adopt an TE perspective to the IE pattern learning these two issues have to be faced. In the following section we will present our approach to the extraction of domain relations over large collection of texts.

4. Learning domain relations from large textual collections

The approach to the extraction of domain relations should be completely corpus driven since information needs are not stated in advance. Therefore, given the corpus C, all the relations have to be analysed in order to detect the more important ones. Since the corpus should suggest the typical domain relations in the first phase of the construction of the domain model (cf. Sec. 2.), the target relations should then not to be too far from the admissible surface form as happens for the concept spotting in TE. As for the concept detection, we should then define the admissible surface forms and a function for estimating the domain importance of the given form. However, a minimal abstraction is needed to take into account the relatively free order of the participants when they appear in the actual text as in the above example (Ex. 1). In the following section (Sec. 4.1.), the admissible surface forms and their equivalence are stated and the size of the problem is estimated. On the other hand, an efficient algorithm for the estimation of the importance function based on the frequency of the relations in the target corpus is presented in Sec. 4.2.

4.1. Admissible surface forms: the size of the problem

A relation $r = (rv, (ra_1, ra_2, ..., ra_n))$ (as the one of the Ex. 2) may be represented in a number of different surface forms. Due to the fact that the corpus should suggest the important relations, we will only consider the realisation of r in verbal phrases. The corpus C s then seen as a collection of verb contexts $c = (v, (a_1, a_2, ...a_n))$ where v is the governing verb and each argument a_i is a couple (g_i, c_i) representing its grammatical role g_i (e.g. subject, object, pp(for), pp(to), etc.) and the concept c_i semantically governing it. A context $c \in C$ is a positive example of the target relation $r \in R$ if $rv = v$ and r partially cover c, i.e. the arguments of r should then appear in any order in the context c.

Given the domain corpus C represented as a collection of verb contexts, the objective is to evaluate the relevance of each possible relation $(r, (ra_1, ra_2, ..., ra_n))$. The first problem is to estimate how many different relations have to be analysed. This may be obtained after partitioning the corpus C according to the verb governing the contexts. For each verb v, a subset of the corpus is then defined as:

$$C(v) = \{(a_1, ..., a_n) | (v, (a_1, ..., a_n)) \in C\} \quad (3)$$

Notice that the notion of context that we use is open to two different 'views'. A lexicalized notion of context is obtained by relying on the full definition. A context $c = (v, ((g_1, c_1), (g_2, c_2), ..., (g_n, c_n)))$ expresses the governing verb v with the lexical (c_i) and its syntactic role (g_i) for each argument found within a given corpus fragment. c_i is usually a partially generalized surface form. c_i denote thus partially generalized surface forms like `companyNE` (for fragments like *IBM*, *Financial Times*, *Apple Ltd.*) or `companyNE_shares` for structures like *IBM's shares*. If we neglect this rich $lexical$ information, and make use a generic concept (e.g. `object`) for the arguments, the remaining information is purely syntactic, making explicit only the grammatical role in the context:

$$c = (v, ((g_1, object), (g_2, object), ..., (g_n, object)))$$

As a result the following two sets of arguments in contexts of $C(v)$ remain defined:

$$A_\Lambda(v) = \{a | \exists (a_1, ..., a_n) \in C(v) \wedge \exists i . a_i = a\} \quad (4)$$

$$A_\Sigma(v) = \{ \begin{array}{l} (s, object) | \exists i . g_i = s \wedge \\ \exists ((g_1, c_1), ..., (g_n, c_n)) \in C(v)\} \end{array} \quad (5)$$

Given the above sets, $A_\Lambda(v)$ and $A_\Sigma(v)$, the set $R(v)$ of the possible relations for a given v is the following:

$$R(v) = \bigcup_{i=1...MC(v)} R_i(v) \quad (6)$$

where $R_i(v)$ are the collection of individual combinations of exactly i arguments in the set $A(v) = A_\Lambda(v) \cup A_\Sigma(v)$ that are syntactically meaningful. The distinction between lexicalised and syntactic arguments is useful to take into account the fact that some relations may have a recurrent syntactic argument whose filler concept is not recurrent.

If $R(v)$ is the set of all the relations for the investigated verb v, the domain importance of each $r(v) \in R(v)$ should be assessed. Therefore, at least the evaluation of the frequency of the relation $r(v)$ over the corpus $C(v)$ has to be used.

Given the defined sets, the size of the $R(v)$ set is, in the worst case, the following:

$$|R(v)| = \sum_{i=1...MC(v)} \left(\begin{matrix} |A(v)|+\ i-1 \\ i \end{matrix} \right)^1 \qquad (7)$$

where $MC(v)$ is the maximum context size for the verb v in $C(v)$. It is worth noticing that $|R(v)|$ values lie in a very large range, due to the size of $A(v)$. In the next section we concentrate on a measure of relevance (for the target domain) that allows to systematically reduce the size of the space where pattern selection is applied for each verb v.

4.2. Estimating the importance: Counting efficiently instances of event prototypes

Given the corpus C, the space of the possible relations is huge. This inherent complexity is the result of tackling the argument order freedom that is neglected in (Yangarber, 2001). In order to tackle with the problem, an informed exploration strategy may be settled. This strategy can not take advantage on the biasing given by the awareness of the final information need that is typical of the IE pattern extraction algorithm. However, some observations may be useful for the purpose:

- the target of the analysis is to emphasize the more important relations arising from the domain corpus
- the frequency of a specific relation strictly depends on the frequency of a more general relation

A very simple but effective domain relevance estimator is represented by the frequency of the relation in the corpus. In this perspecitive, the more important relations are the more frequent. Therefore, the above considerations may reduce the complexity of the search algorithm if only promising relation are explored, i.e. patterns whose generalisations are over a frequency threshold.

The idea is then to drive the analysis using the pattern generalisation that may be obtained projecting the patterns on their "syntactic" counterpart. The projection $\widehat{\Sigma}(r)$ of the relation r over the syntactic space Σ is defined as follows:

$$\widehat{\Sigma}(r) = (\widehat{\Sigma}(ra_1), ..., \widehat{\Sigma}(ra_m))$$

where $\widehat{\Sigma}(ra_i) = ra_i$ if ra_i is a "syntactic" argument ($ra_i \in A_\Sigma(v)$) or $\widehat{\Sigma}(ra_i) = (s_i, object)$ if $ra_i = (g_i, c_i)$ is a lexicalised argument ($ra_i \in A_\Lambda(v)$). The resulting search space $R_\Sigma(v) = \{\widehat{\Sigma}(r) | r \in R(v)\}$ is greatly smaller than $R_\Sigma(v)$ since $|A_\Lambda(v)| >> |A_\Sigma(v)| = \#preposition + 2$. This search space can be used for the extraction of the more promising generalised relations. This subset $\overline{R_\Sigma}$ can be used for narrowing the search space of the following step. In fact, when the acceptance threshold is settled, the resultant admissible relations are confined in the following set:

$$\overline{R}(v) \widehat{=} \{r | \Sigma(r) \in \overline{R_\Sigma}(v)\} \qquad (8)$$

[1]Notice that, in syntactically meaningful contexts, arguments may appear with multiplicity higher than 1, so that the factorial expression is a useful approximation.

The overall domain importance estimation procedure may take also advantage from the fact that the order of the relation arguments may be fixed after the analysis of the promising syntactic patterns. The final counting activity can be thus performed with a simple sorting algorithm with the $O(n\ log(n))$ complexity. In this case n is directly related to the number of context samples in the corpus $C(v)$. The procedure is sketched in the following:

procedure SelectAndRankRelations($R(v)$,$C(v)$)
begin
Select $\overline{R_\Sigma}(v) = \{r \in R_\Sigma(v) | hits(r, C(v)) \geq K\ \}$;
Set $L = \emptyset$;
for each $r \in \overline{R_\Sigma}(v)$
$L := L \cup proj(\ C(v), r)$;
$RankedR\ \ (v)$:= CountEquals(L);
return $RankedR(v)\ \ ;$
end

where $hits(r, C(v))$ is the number of instances of the relation r in $C(v)$ e $prj(C(v), r)$ is the projection of the contexts in $C(v)$ on the syntactic relation r. The procedure $CountEqualsl(L)$ using a standard sorting algorithm counts the repetition of each element in L. Finally, $RankedR(v)$ is the set of couples (f, r) where f the frequency of the relation $r \in \overline{R}(v)$ on the corpus.

5. A case study: IE patterns for the financial domain

The above methodology has been applied for the definition of an ontology for a financial domain. The ontology construction steps have been followed. Firstly, an homogeneous collection of texts has been prepared as the model for the target domain, namely a collection of 13,000 news stories of the *Financial Time* over a period of time ranging from 2000 to 2001. The corpus will be hereafter called $FinTimeNews$. The analysis of the corpus has been carried out with the Chaos robust parser (Basili et al., 2000).

In the tables 1 and 2, excerpts of the lists related to the complex concepts and the relations governed by the verb *to make* are respectively shown. The lists are sorted according to their frequency in the *FinTimeNews* corpus (f in the tables). A manual assessed domain relevance is then reported (DR in the tables). The rate of the complex concepts retained as useful exceeds the 60% in the presented top 50 positions. It is worth noticing that many of the complex concepts that have not been judged important for the domain are in fact relevant time indicator. These are not useful for understanding the nature of the domain knowledge but they are precious in the perspective of a IE system for the characterisation of the time stamp of the event. Some of these expression such as `first_half` are in any case typical of the financial jargon, in particular they are used in the declaration of the companies' economic performance.

In the case of the relations governed by the verb *make*, the number of domain relevant relations in the top 50 is around 28%. The other presented relations are generally phraseological use of the same verb.

The sorted lists allows the definition of the top level hierarchy of the possible events in the financial domain.

f	Surface form	DR
2924	last_year	
1739	chief_executive	✓
1138	last_week	
1086	next_year	
956	percentNE_stake	✓
946	entityNE_share	✓
834	last_month	
737	oil_price	
687	joint_venture	✓
641	first_half	
631	pre-tax_profit	✓
618	interest_rate	✓
583	entityNE_yesterday	
575	entityNE_company	✓
551	stake_in_entityNE	✓
499	prime_minister	✓
453	first_time	
438	entityNE_market	✓
431	entityNE_index	✓
429	earnings_per_share	✓
413	share_in_entityNE	✓
412	mobile_phone	
396	profit_of_currencyNE	✓
374	next_month	
361	second_quarter	
358	entityNE_official	
348	second_half	
341	few_year	
341	same_time	
337	entityNE_government	✓
332	next_week	
318	last_night	
316	percentNE_rise	✓
316	end_of_the_year	
309	end_of_dateNE	
299	entityNE's_share	✓
291	economic_growth	✓
285	recent_year	
281	loss_of_currencyNE	✓
281	central_bank	✓
275	entityNE_deal	✓
269	percentNE_increase	✓
267	percentNE_stake_in_entityNE	✓
248	public_offering	✓
240	executive_of_entityNE	✓
237	net_profit	✓
234	past_year	
234	entityNE_economy	✓
230	acquisition_of_entityNE	✓
229	entityNE_shareholder	✓

Table 1: Complex concepts in $FinTimesNews$

f	Surface form	DR
150	*(make,[(dirobj,sense)])*	
132	*(make,[(dirobj,money)])*	✓
121	*(make,[(dirobj,profit)])*	✓
118	*(make,[(dirobj,decision)])*	
108	*(make,[(for,entityNE)])*	
106	*(make,[(dirobj,sense),(subj,null)])*	
102	*(make,[(in,locationNE)])*	
100	*(make,[(to,entityNE)])*	
100	*(make,[(dirobj,null),(for,entityNE)])*	
95	*(make,[(subj,company)])*	✓
87	*(make,[(dirobj,acquisition)])*	✓
83	*(make,[(for,null),(subj,entityNE)])*	
81	*(make,[(dirobj,null),(to,entityNE)])*	
80	*(make,[(dirobj,null),(in,locationNE)])*	
79	*(make,[(dirobj,progress)])*	✓
76	*(make,[(in,entityNE)])*	
75	*(make,[(dirobj,null),(subj,company)])*	✓
71	*(make,[(subj,locationNE)])*	
71	*(make,[(dirobj,use)])*	
71	*(make,[(dirobj,difference)])*	
66	*(make,[(dirobj,use),(of,null)])*	
65	*(make,[(subj,entityNE),(to,null)])*	
60	*(make,[(dirobj,offer)])*	✓
57	*(make,[(subj,null),(to,entityNE)])*	
57	*(make,[(dirobj,null),(in,entityNE)])*	
55	*(make,[(dirobj,profit),(subj,null)])*	✓
55	*(make,[(dirobj,null),(subj,locationNE)])*	
54	*(make,[(dirobj,effort)])*	
53	*(make,[(in,locationNE),(subj,null)])*	
53	*(make,[(dirobj,currencyNE)])*	✓
51	*(make,[(dirobj,mistake)])*	
50	*(make,[(dirobj,null),(subj,entityNE),(to,null)])*	
49	*(make,[(dirobj,debut)])*	✓
48	*(make,[(for,entityNE),(subj,null)])*	
48	*(make,[(dirobj,money),(subj,null)])*	✓
48	*(make,[(dirobj,bid)])*	✓
47	*(make,[(dirobj,locationNE)])*	
46	*(make,[(on,null),(subj,entityNE)])*	
45	*(make,[(dirobj,null),(for,entityNE),(subj,null)])*	
45	*(make,[(dirobj,entityNE),(dirobj2,null),(subj,null)])*	
45	*(make,[(dirobj,difference),(subj,null)])*	
44	*(make,[(dirobj,sense),(subj,it)])*	
42	*(make,[(dirobj,progress),(subj,null)])*	
42	*(make,[(dirobj,decision),(subj,null)])*	
41	*(make,[(dirobj,investment)])*	✓
40	*(make,[(dirobj,payment)])*	✓
39	*(make,[(dirobj,case)])*	
38	*(make,[(dirobj2,currencyNE)])*	
37	*(make,[(dirobj,contribution)])*	
35	*(make,[(with,entityNE)])*	
35	*(make,[(dirobj,loss)])*	✓

Table 2: Relations governed by the verb *to make* in $FinTimesNews$

These have been defined as follows:

1. Relationships among companies
   (a) Acquisition/Selling
   (b) Cooperation/Splitting
2. Industrial Activities
   (a) Funding/Capital
   (b) Company Assets (Financial Performances, Balance Sheet Analysis)
   (c) Staff Movement (e.g Management Succession)
   (d) External Communications
3. Company Positioning
   (a) Position vs. the competitors
   (b) Market Sector
   (c) Market Strategies
4. Governamental Activities
   (a) Tax Reduction/Increase
   (b) Anti-trust Control
5. Job Market - Mass Employment/Unemployment
6. Stock Market
   (a) Share Trends
   (b) Currencies Trends

Once the definition of the top level events has been completed, the discovered event prototypes have been manually clustered according to their class. To give the flavour of the information contained in the produced knowledge base, in the following an excerpt of the event prototypes of the *Company Assets* class are presented:

Company Assets Event Prototypes
(cut,[(subj,entityNE),(dirobj,cost)]))
(rise,[(subj,profit),(to,currencyNE)])
(rise,[(from,currencyNE),(subj,profit),(to,currencyNE)])
(issue,[(subj,entityNE),(dirobj,profit_warning)]))
(suffer,[(subj,entityNE),(dirobj,loss)])
(report,[(subj,entityNE),(dirobj,loss_of_currencyNE)])
(announce,[(subj,entityNE),(dirobj,loss_of_currencyNE)])

The analysis of $1{,}100$ patterns give rise to 229 patterns retained as useful for the definition of the event prototypes in one of the give class.

6. Conclusions and future work

In this paper we presented a terminological perspective to the extraction of IE patterns. This corpus driven method is more suitable for a wide application of IE-based systems with respect to learning methods driven by the specific information need. The presented method helps in performing the activities required for building a domain ontology since the concepts and the relations are presented according to their relevance for the target domain.

Many issues are still open and are objective of further research. First of all, a more complete evaluation of the method should be performed with respect to the task of

event recognition. The acquired ontology should be evaluated in order to understand if the level of detail of the event prototypes is deep enough for the experts to classify the event prototypes in the correct class. Therefore, we intend to study the possibility of automatically cluster the event prototypes once the domain top level hierarchy has been defined. We will try here to adopt a booting algorithm and we will study the size of the necessary booting data. Finally, domain relations (i.e. IE patterns) not headed by verbs may be an interesting area of research.

7. References

Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. 2000. Customizable modular lexicalized parsing. In *Proc. of the 6th International Workshop on Parsing Technology, IWPT2000*, Trento, Italy.

1998. In Didier Bourigault, Christian Jacquemin, and Marie-Claude L'Homme, editors, *Proceedings of the First Workshop on Computational Terminology COMPUTERM'98, held jointly with COLING-ACL'98*, Montreal, Quebec, Canada.

IPTC. Iptc standards. In *www.iptc.org*.

Emmanuel Morin. 1999. *Extraction de liens sémantiques entre termes à partir de corpus de textes techniques*. Ph.D. thesis, Univesité de Nantes, Faculté des Sciences et de Techniques.

MUC-7. 1997. Proceedings of the seventh message understanding conference(muc-7). In *Columbia, MD*. Morgan Kaufmann.

Maria Teresa Pazienza. 1997. *Information Extraction. A Multidisciplinary Approach to an Emerging Information Technology*. Number 1299 in LNAI. Springer-Verlag, Heidelberg, Germany.

Ellen Riloff and R. Jones. 1999. Learning dictionaries for information extraction by multi-level bootstrapping. In *Proceedings of the Sixteenth National Conference on Artificial Intelligence (AAAI-99)*.

Roman Yangarber. 2001. *Scenario Customization for Information Extraction*. Ph.D. thesis, Courant Institute of Mathematical Sciences, New York University.

Unsupervised Event Clustering in Multilingual News Streams

Martijn Spitters, Wessel Kraaij

Department of Multimedia Technology & Statistics
TNO TPD
P. O. Box 155, 2600 AD Delft
The Netherlands
{spitters, kraaij}@tpd.tno.nl

Abstract

The Topic Detection and Tracking (TDT) benchmark evaluation project embraces a variety of technical challenges for information retrieval research. The TDT topic detection task is concerned with the unsupervised grouping of news stories according to the events they discuss. A detection system must both discover new events as the incoming stories are processed and associate incoming stories with the story clusters created so far. The TNO topic detection system is based on a language modeling approach. The system has been evaluated on a multilingual corpus of approximately 80.000 stories from multiple new sources. For the grouping of stories we combined a simple single pass method to establish an initial clustering and a reallocation method to stabilize the clusters within a certain allowed deferral period. The similarity of an incoming story S_n to an existing cluster C is defined as the average of the similarities of S_n to each story $S_i \in C$. These individual similarities are computed by taking the sum of the generative probabilities $P(S_n|S_i)$ and $P(S_i|S_n)$ where S_i and S_n are modeled as unigram language models. Because these story language models are based on extremely sparse statistics, the word probabilities are smoothed using a background model.

1. Introduction

This paper describes the design and development of a system for the unsupervised grouping of news stories according to the events they discuss. The system has been evaluated on an augmented version of the TDT3 corpus which contains approximately 80.000 stories from multiple news sources, including both text and speech. These sources are newswires, radio and television broadcasts, and internet sites. The source languages are English and Mandarin. The TDT3 corpus is annotated for 120 events, each of which spans both English and Mandarin sources.

The TNO topic detection system is based on a language modeling approach. We had good experience with the application of language models for different IR-related tasks, like ad hoc, cross language, web and spoken document retrieval (Hiemstra and Kraaij, 1999; Kraaij et al., 2000; Hiemstra et al., 2001; Kraaij et al., 2002), filtering (Ekkelenkamp et al., 1999), and multi-document summarization (Kraaij et al., 2001). We also successfully applied language models for topic tracking (Spitters and Kraaij, 2001). However, due to the substantially higher computational complexity of topic detection, it was not trivial to convert our tracking approach into a detection algorithm. In the topic tracking task, events are to be followed individually. Each target event is defined by a small set of training stories that discuss it. Our tracking system estimates a single unigram language model based on the union of these on-topic stories and computes for each incoming story the likelihood according to this topic model. The computational complexity of this process is linear to the input. However, the topic detection task is a highly dynamic process. The topic models are constructed on the fly from the incoming stories. Each incoming story is added to a cluster, and thus changes the corresponding topic model. Experiments showed that reclustering the already processed stories (within the allowed deferral window) is important for a good performance. Reclustering is a computationally demanding process, since every change in cluster membership lists is reflected in changes in the cluster models, which form the basis for the similarity computation. Therefore we have chosen for a clustering approach which is independent of the (global) cluster models and instead is based on the similarities between individual stories. The advantage of this approach is that the inter-story similarities can be cached, resulting in a significant speed-up of the clustering process.

The remainder of this paper is organized as follows. To familiarize the reader with the TDT framework, section 2 elaborates on the TDT corpora, the TDT research tasks, and the TDT evaluation method. In section 3 we describe in detail our language model-based approach to topic detection. This section also contains a short study into the influence of two different smoothing methods for language models on the detection performance of our system. In section 4 we try to draw some conclusions.

2. The TDT benchmark test

The topic detection and tracking (TDT) benchmark evaluation project[1] was initiated by DARPA in 1996. After a pilot study in 1997, TDT has continued with annual evaluations conducted by the National Institute of Standards and Technology (NIST). Main purpose of the TDT project is to advance the state-of-the-art in determining the topical structure of multilingual news streams from various sources. See (Wayne, 2000) for a detailed overview of the TDT project.

2.1. TDT corpora

Currently, the Linguistic Data Consortium (LDC) has three corpora available to support TDT research[2] (Cieri et al., 2000). The TDT-Pilot corpus contains newswire and

[1] http://www.nist.gov/speech/tests/tdt

[2] http://www.ldc.upenn.edu/Projects/TDT

transcripts of news broadcasts, all in English, and is annotated for 25 news events. The TDT2 and TDT3 corpora are multilingual (Chinese and English) and contain both audio and text. ASR transcriptions and close captions of the audio data as well as Systran translations of the Chinese data are also provided. TDT2 and TDT3 are completely annotated for 100 and 120 events respectively. Currently, LDC is developing a new TDT corpus (TDT4) which will include Arabic news.

In the TDT evaluation, there are three alternative choices for the form of the audio sources to be processed, namely manual transcriptions, ASR transcriptions, or the sampled audio signal. Three story boundary conditions are supported: reference story boundaries (manually determined correct boundaries), automatic story boundaries (automatically determined errorful boundaries), or no story boundaries (the system must provide its own boundaries). Sites that participate in one of the TDT tasks are required to perform at least one evaluation under shared conditions. See (Doddington and Fiscus, 2001) for the TDT evaluation details.

2.2. TDT research tasks

The TDT benchmark evaluation project embraces a variety of technical challenges for information retrieval research. The goal of *story segmentation* is to segment a stream of data into homogeneous regions, discussing certain events. Given a small number of stories that discuss a certain event, a *tracking* system has the task to detect which stories in the data stream are related to this event and which are not. In *topic detection* there is no knowledge of the events to be detected. A detection system must both discover new events as the incoming stories are processed and associate incoming stories with the event-based story clusters created so far. A task which is very similar to topic detection is *first-story detection*. The goal of this task is to detect, in a chronologically ordered stream of stories, the first story that discusses a certain event. Finally, in *link detection*, the question to be answered is whether or not two stories discuss the same event.

2.3. TDT evaluation method

Topic detection systems are evaluated in terms of their ability to cluster together stories that discuss the same event (or events and activities that are directly connected to the cluster's seminal event). Detection performance is characterized in terms of the probability of miss and false alarm errors (P_{Miss} and P_{FA}). To speak in terms of the more established and well-known precision and recall measures: a low P_{Miss} corresponds to high recall, while a low P_{FA} corresponds to high precision.

These two error probabilities are combined into a single detection cost C_{Det}, by assigning costs to miss and false alarm errors (Doddington and Fiscus, 2001):

$$C_{Det} = C_{Miss} \cdot P_{Miss} \cdot P_{target} + C_{FA} \cdot P_{FA} \cdot P_{\neg target} \quad (1)$$

where C_{Miss} and C_{FA} are the costs of a miss and a false alarm respectively; P_{Miss} and P_{FA} are the conditional probabilities of a miss and a false alarm respectively; P_{target} and $P_{\neg target}$ are the a priori target probabilities ($P_{\neg target} = 1 - P_{target}$).

Then C_{Det} is normalized to:

$$(C_{Det})_{Norm} = \frac{C_{Det}}{min(C_{Miss} \cdot P_{target}, C_{FA} \cdot P_{\neg target})} \quad (2)$$

Detection error probability is estimated by accumulating errors seperately for each topic and by taking the average of the error probabilities over topics, with equal weight assigned to each topic. A set of predefined topics is automatically mapped to the system output topics by choosing for each reference topic the system output topic which produces the lowest evaluation cost.

3. Design of a probabilistic topic detection system

This section describes in detail the design of the TNO topic detection system. 3.1. describes our clustering approach. We combined a simple single pass method to establish an initial clustering and a reallocation method to stabilize the clusters within a certain allowed deferral period. In 3.2. we describe our story-cluster similarity measure. An incoming story is compared to an existing cluster by averaging the similarities of the new story S_n to each story in the cluster S_i. These individual similarities are defined as the sum of the generative probabilities $P(S_n|S_i)$ and $P(S_i|S_n)$ where S_i and S_n are modeled as unigram language models. Because these story language models are based on extremely sparse statistics, the word probabilities are smoothed using a background model. Section 3.3. reports on our experiments concerning the application of two different smoothing methods for language models and some contrastive tests with automatic versus manually determined story boundaries.

3.1. Clustering method

Our clustering procedure combines a simple single pass method and a reallocation method. Because the clusters formed by the single pass method are dependent of the order in which the stories are processed, they are merely used to initiate reallocation clustering. However, because in the TDT evaluation a topic detection system may defer its assignment of stories until a limited amount of subsequent source data (10 source files) is processed, the reallocation is restricted to the stories within that deferral period. More specifically, our clustering process involves the following steps:

1. For each new story within the deferral window, compute its similarity to each cluster the system has created so far. There are two options for a story:
   (a) if the similarity of the story to the closest cluster exceeds a certain threshold, assign the story to that cluster
   (b) else create a new cluster with the concerning story as its seed

2. When the end of the deferral window is reached, loop through the window stories again and compare each story to each existing cluster. There are three options for a story:

   (a) a story may switch to another cluster if the similarity to that cluster exceeds both the similarity to its current cluster and the threshold

   (b) if neither the similarity to its current cluster nor the similarity to any other cluster exceeds the threshold, create a new cluster with the concerning story as its seed

   (c) if the similarity to its current cluster exceeds the threshold as well as the similarities to all other clusters, the story stays in its current cluster

   Step 2 is repeated until all clusters are stable, that is, when 2c is true for each story.

The combination of a cluster initialization step and a reallocation step has previously (successfully) been used for topic detection by a.o. BBN (Walls et al., 1999) and Dragon (Yamron et al., 2000).

The reclustering step is important for a good performance of the detection system. However, the fact that every change in a cluster membership list means that the cluster language model would have to be reestimated, makes it a computationally demanding process. Therefore we have chosen for an approach which does not use the global cluster language models (contrary to our topic tracking approach) but instead is based on the similarities between individual stories. The similarity of an incoming story S_n to an existing cluster C is defined as the average of the similarities of S_n to each story $S_i \in C$. The advantage of this approach is that the inter-story similarities can be cached, resulting in a significant acceleration of the clustering process. These inter-story similarities are computed using a two-way language modeling approach, which is discussed in detail in the following section.

A cluster which has not changed for an uninterrupted period of fifteen days is frozen, which means that it is no longer considered an 'active event'. The cluster is removed from the list of candidate clusters for new stories. This cluster evolution monitoring has two advantages. First of all it limits the computational complexity, because the number of clusters a story has to be compared with stays within certain bounds. Second, it can be argued that restricting the temporal extent of an event is beneficial for detection performance because it prevents different events with similar vocabulary (like different attacks or political elections) to be grouped together (Yang et al., 1999).

3.2. Language model-based similarity

The basic idea behind the language modeling approach to information retrieval is to estimate a (usually unigram) language model for each document and to rank documents by the probability that the document model generated the query. Absolute probabilities are not important for ranking in the IR situation. For other applications, i.e. topic tracking and also topic detection, scores have to be comparable on an absolute scale. For tracking, we found that modeling similarity as a likelihood ratio and normalizing this likelihood ratio by the (test) story length was adequate (Spitters and Kraaij, 2001). This normalized likelihood ratio is presented in equation (3), where $\mathrm{LLR}_{Norm}(T_1, T_2, ..., T_n|S_k)$ denotes the normalized log likelihood ratio of a story consisting of the terms $T_1, ..T_n$ given the story S_k in comparison with background model $\mathcal{B}$.

$$\mathrm{LLR}_{Norm}(T_1, T_2, ..., T_n|S_k) = \frac{1}{n} \log \sum_{i=1}^{n} \frac{P(T_i|S_k)}{P(T_i|\mathcal{B})} \quad (3)$$

In our clustering approach, the similarity between two stories S_n and S_i is based on a combination of the probability that the language model representing S_n generated story S_i and the reverse: the probability that the language model representing S_i generated story S_n. This approach results in the symmetrical similarity measure, presented in the following equation:

$$Sim(S_n, S_i) = \mathrm{LLR}_{Norm}(S_n|S_i) + \mathrm{LLR}_{Norm}(S_i|S_n) \quad (4)$$

Because the language models are estimated based on very limited amounts of text (single stories), it is very important that the word probabilities are smoothed using some background model. We performed a short study into the influence of two different smoothing methods on the performance of our detection system: Bayesian smoothing using Dirichlet priors and Jelinek-Mercer smoothing. The details of these smoothing methods and the results of our experiments are described in the following section.

3.3. Smoothing

Recent experiments at CMU have shown that different smoothing methods have different characteristics (Zhai and Lafferty, 2001a). For title ad hoc queries, Zhai and Lafferty found Dirichlet smoothing to be more effective than linear interpolation (Jelinek-Mercer smoothing). Both methods start from the idea that the probability estimate for unseen terms: $P_u(T_i|S_k)$ is modeled as a coefficient α_s times the background collection based estimate: $P_u(T_i|S_k) = \alpha_s \cdot P(T_i|\mathcal{B})$. A crucial difference between Dirichlet and Jelinek-Mercer smoothing is that the smoothing coefficient is dependent on the story length for Dirichlet, reflecting the fact that probability estimates are more reliable for longer stories. Formula (5) shows the weighting formula for Dirichlet smoothing, where $c(T_i|S_k)$ is the term frequency of term T_i in story S_k , $\sum_w c(T_i; S_k)$ is the length of story S_k and μ is a constant. The smoothing coefficient α_s is in this case $\frac{\mu}{\sum_w c(T_i;S_k)+\mu}$, whereas the smoothing coefficient is λ in the Jelinek-Mercer based model (formula (6)).

$$P(T_1, T_2, \cdots, T_n|S_k) = \prod_{i=1}^{n} \frac{c(T_i; S_k) + \mu P(T_i|\mathcal{B})}{\sum_w c(T_i; S_k) + \mu} \quad (5)$$

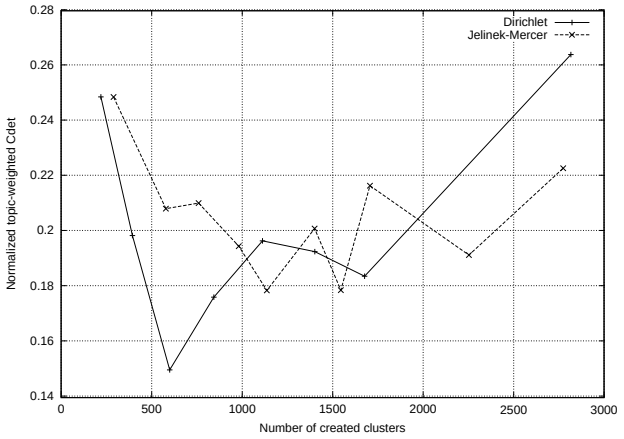


Figure 1: C_{Det} at different decision thresholds for two smoothing methods (Dirichlet and Jelinek-Mercer), performed on the TDT2 stories from April 1998, using automatic boundaries.

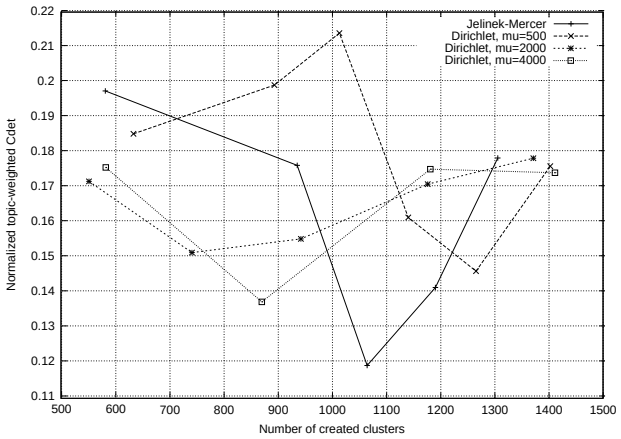


Figure 2: C_{Det} at different decision thresholds for two smoothing methods (Dirichlet and Jelinek-Mercer), performed on the TDT2 stories from April 1998, using reference boundaries.

$$P(T_1, T_2, \cdots, T_n|S_k) \;=\; \prod_{i=1}^{n} \lambda P(T_i|\mathcal{B}) + (1-\lambda) P(T_i|S_k) \tag{6}$$

For our official TDT2001 detection run, we applied Dirichlet smoothing with $\mu = 2000$. Our hypothesis was that Dirichlet smoothing would lead to improved performance, since story lengths vary considerably in the TDT corpus, and Dirichlet performed better than Jelinek-Mercer smoothing on a small test corpus (one month of stories from the TDT2 corpus) using the automatic story boundaries and ASR transcriptions of the audio (the primary topic detection evaluation requires these conditions). The results of this experiment are plotted in Figure (1).

We performed some post-hoc experiments on this same test set using reference story boundaries instead of automatic story boundaries and were surprised to find that Jelinek-Mercer performed better than Dirichlet under that condition, even when we varied μ (see equation (5)). Figure (2) shows the results. It is too early to draw conclusions from these experiments, since the test set was small and we did not explore the complete parameter space. However, one explanation could be the observation from Zhai and Lafferty (Zhai and Lafferty, 2001b; Zhai and Lafferty, 2001a) that smoothing has two functions: i) improving the maximum likelihood estimates ii) generate common words in the query. The latter function is especially important for longer queries since they contain more common words.

In the topic detection task we use language models to generate stories instead of queries. Since stories are considerably longer than TREC title queries, it is probably important that the smoothed model generates common words with proper "idf"-like probabilities. The TREC experiments show that the two roles of smoothing have an inverse interaction with the query length. Dirichlet is a good strategy for the first smoothing role (avoiding the assignment of a zero probability to an unseen word) while Jelinek-Mercer is better for the second role (weighting query terms in an idf-like fashion) (Zhai and Lafferty, 2001a). The longer the "queries" are, the more important the second function will become. This phenomenenon might be an explanation for the fact that Dirichlet performs best under the automatic story boundary condition, and Jelinek-Mercer under the reference story boundary condition, since the former has shorter stories than the latter (median: 62 versus 114). Further experiments are needed, including a validation of a combined Dirichlet/Jelinek-Mercer smoothing scheme for the TDT tasks.

4. Conclusions and future work

We think that the choice to use normalized likelihood ratios as the basis of a similarity measure was the key for the good performance of our system. Like in the tracking task, a proper normalized similarity measure is of utmost importance. Simply adding the generative probabilities $P(S_n|S_i)$ and $P(S_i|S_n)$ proved to work well to "symmetrize" the similarity measure. The accuracy of a language model-based clustering approach which is independent of the (global) cluster models and instead is based on the similarities between individual stories surpassed our expectations. However, we intend to check whether a similarity measure based on the global cluster model would enhance the results. The results of some initial post-hoc experiments indicate that the Jelinek-Mercer smoothing method works better than Dirichlet smoothing for manually segmented data, while the Dirichlet method yields better performance than Jelinek-Mercer on automatically segmented data. Further investigation is necessary to draw definite conclusions.

5. References

C. Cieri, D. Graff, M. Liberman, N. Martey, and S. Strassel. 2000. Large multilingual broadcast news corpora for cooperative research in topic detection and tracking: The TDT2 and TDT3 corpus efforts. *Proceedings of the Language Resources and Evaluation Conference (LREC2000)*.

G. Doddington and J. Fiscus. 2001. The year 2001 topic detection and tracking (TDT2001) task definition and evaluation plan. Technical Report v. 1.0, National Institute of Standards and Technology.

R. Ekkelenkamp, W. Kraaij, and D. van Leeuwen. 1999. TNO TREC-7 site report: SDR and filtering. *Proceedings of the Seventh Text Retrieval Conference (TREC-7)*, pages 519–526.

D. Hiemstra and W. Kraaij. 1999. Twenty-one at trec-7: Ad hoc and cross language track. *Proceedings of the Seventh Text Retrieval Conference (TREC-7)*, pages 227–238.

D. Hiemstra, W. Kraaij, R. Pohlmann, and T. Westerveld. 2001. Twenty-one at CLEF 2000: Translation resources, merging strategies and relevance feedback. *Proceedings of the CLEF 2000 Cross-Language Text Retrieval System Evaluation Campaign.*

W. Kraaij, R. Pohlmann, and D. Hiemstra. 2000. Twenty-one at TREC-8: using language technology for information retrieval. *Proceedings of the Eighth Text Retrieval Conference (TREC-8)*, pages 282–299.

W. Kraaij, M. Spitters, and M. van der Heijden. 2001. Combining a mixture language model and naive bayes for multi-document summarisation. *Notebook papers of the Document Understanding Conference (DUC 2001)*.

W. Kraaij, T. Westerveld, and D. Hiemstra. 2002. The importance of prior probabilities for entry page search. *Proceedings of the 25th Annual ACM SIGIR Conference on Research and Development in Information Retrieval*, To Appear.

M. Spitters and W. Kraaij. 2001. Using language models for tracking events of interest over time. *Proceedings of the Workshop on Language Models for Information Retrieval (LMIR 2001)*, pages 60–65.

F. Walls, H. Jin, S. Sista, and P. van Mulbregt. 1999. Topic detection in broadcast news. *Proceedings of the DARPA Broadcast News Workshop*.

C.L. Wayne. 2000. Multilingual topic detection and tracking: Successful research enabled by corpora and evaluation. *Proceedings of the Language Resources and Evaluation Conference (LREC2000)*, pages 1487–1494.

J.P. Yamron, S. Knecht, and P. van Mulbregt. 2000. Dragon's tracking and detection system for the TDT2000 evaluation. *Notebook papers of the Topic Detection and Tracking Workshop (TDT) 2000*.

Y. Yang, J. Carbonell, R. Brown, T. Pierce, B.T. Archibald, and X. Liu. 1999. Learning approaches for detecting and tracking news events. *IEEE Intelligent Systems: Special Issue on Applications of Intelligent Information Retrieval*, 14(4):32–43.

C. Zhai and J. Lafferty. 2001a. Dual role of smoothing in the language modeling approach. *Proceedings of the Workshop on Language Models for Information Retrieval (LMIR) 2001*, pages 31–36.

C. Zhai and J. Lafferty. 2001b. A study of smoothing methods for language models applied to ad hoc information retrieval. *Proceedings of SIGIR 2001.*

The Workshop Programme

9:15–9:30	Opening: Aims of the workshop
9:30–10:00	*Relational evaluation schemes* Ted Briscoe, John Carroll, Jonathan Graham, Ann Copestake
10:00–10:30	*Towards a dependency-oriented evaluation for partial parsing* Sandra Kübler, Heike Telljohann
10:30–11:00	*LinGO Redwoods—A rich and dynamic treebank for HPSG* Stephan Oepen, Ezra Callahan, Dan Flickinger, Christoper D. Manning
11:00–11:30	Coffee break
11:30–12:30	Panel: Parser evaluation in context John Carroll, Patrick Paroubek, Owen Rambow, Hans Uszkoreit
12:30–14:00	Lunch break
14:00–14:30	*A test of the leaf-ancestor metric for parse accuracy* Geoffrey Sampson, Anna Babarczy
14:30–15:00	*Evaluating parser accuracy using edit distance* Brian Roark
15:00–15:10	Short break
15:10–15:40	*Evaluating syllabification: One category shared by many grammars* Karin Müller
15:40–16:10	*Towards comparing parsers from different linguistic frameworks: An information theoretic approach* Gabriele Musillo, Khalil Sima'an
16:10–16:40	*Evaluation of the Gramotron parser for German* Franz Beil, Detlef Prescher, Helmut Schmid, Sabine Schulte im Walde
16:40–17:10	Coffee break
17:10–17:40	*Evaluating a wide-coverage CCG parser* Stephen Clark, Julia Hockenmaier
17:40–18:10	*A comparison of evaluation metrics for a broad-coverage stochastic parser* Richard Crouch, Ronald M. Kaplan, Tracy H. King, Stefan Riezler
18:10–20:00	Wrap up and kick-off: Initiatives and action plans (open end)

Workshop Organisers

John Carroll	University of Sussex, UK
Anette Frank	DFKI GmbH, Saarbrücken, Germany
Dekang Lin	University of Alberta, Canada
Detlef Prescher	DFKI GmbH, Saarbrücken, Germany
Hans Uszkoreit	DFKI GmbH and Saarland University, Saarbrücken, Germany

Workshop Programme Committee

Salah Ait-Mokhtar	XRCE Grenoble
Gosse Bouma	Rijksuniversiteit Groningen
Thorsten Brants	Palo Alto Research Center
Ted Briscoe	University of Cambridge
John Carroll	University of Sussex
Jean-Pierre Chanod	XRCE Grenoble
Michael Collins	AT&T Labs—Research
Anette Frank	DFKI Saarbrücken
Josef van Genabith	Dublin City University
Gregory Grefenstette	Clairvoyance, Pittsburgh
Julia Hockenmaier	University of Edinburgh
Dekang Lin	University of Alberta
Chris Manning	Stanford University
Detlef Prescher	DFKI Saarbrücken
Khalil Sima'an	University of Amsterdam
Hans Uszkoreit	DFKI Saarbrücken and Saarland University

Table of Contents

Author Index

— Beyond PARSEVAL —
Towards Improved Evaluation Measures for Parsing Systems

John Carroll[1], **Anette Frank**[2], **Dekang Lin**[3], **Detlef Prescher**[2], **Hans Uszkoreit**[2]

[1] Cognitive and Computing Sciences
University of Sussex
Falmer, Brighton BN1 9QH
UK

[2] Language Technology Lab
DFKI GmbH
Stuhlsatzenhausweg 3
66123 Saarbrücken
Germany

[3] Department of Computing Science
University of Alberta
Edmonton, Alberta
Canada, T6G 2H1

1. Current Situation in Stochastic Parsing

The earliest corpus-based approaches to stochastic parsing (e.g. Sampson et al. (1989), Fujisaki et al. (1989), Sharman et al. (1990), Black (1992)) used a variety of data resources and evaluation techniques. With the creation of the Penn Treebank of English (Marcus et al., 1993) and the parser evaluation measures established by the PARSEVAL initiative (Black, 1992), new approaches to stochastic parsing and uniform evaluation regimes emerged (Magerman (1995), Charniak (1996), Collins (1996)), leading to impressive improvements in parser accuracy (Collins (1997), Charniak (2000), Bod (2001)).

In the meantime, annotated corpora have been built for several other languages, most notably the Prague Dependency Treebank for Czech (Hajic, 1998), and the NEGRA corpus for German (Skut et al., 1997). Well-known, but smaller corpora for English are the ATIS Corpus and SUSANNE. Many more corpora are available or under construction, e.g. the Penn treebanks for Chinese and Korean, the TIGER corpus for German, as well as corpora for Bulgarian, French, Italian, Portugese, Spanish, Turkish, etc. Annotation schemes in these treebanks vary, often motivated by language-specific characteristics. For example, dependency-based annotation is generally preferred for languages with relatively free word order.

More recently, in line with increasing interest in more fine-grained syntactic and semantic representations, stochastic parsing has been applied to several higher-order syntactic frameworks, such as unification-based grammars (Johnson et al., 1999), tree-adjoining grammars (Chen et al., 1999) and combinatory categorial grammars (Hockenmaier, 2001). In parallel, due to the lack of appropriate large-scale annotated training corpora, unsupervised methods have been investigated, i.e. training of manually written (context-free or unification-based) grammars on free text (Beil et al. (1999), Riezler et al. (2000), Bouma et al. (2001)).

As opposed to the PARSEVAL measures — which are based on phrase structure tree match — most of these novel parsing approaches use other evaluation measures, such as dependency-based, valence-based, exact, or selective category match.

2. Challenges for Parser Evaluation

Despite the emergence of stochastic parsing approaches using alternative syntactic frameworks, the currently established paradigm for evaluating stochastic parsing still consists of the combination of Penn Treebank English (Section 23) with PARSEVAL measures.

However, in practice (especially if we count industrial labs) parsing systems using treebank grammars are not representative of the field. Moreover, a strong trend in stochastic parsing is away from treebank grammars and towards higher-level syntactic frameworks and hand-built grammars.

Research in stochastic parsing with higher-order syntactic frameworks is therefore confronted with a lack of a common evaluation metrics: neither do the PARSEVAL measures straightforwardly correspond to dependency structures or other valence-based representations, nor have these alternative approaches come up with a common, agreed-on standard for evaluation. Furthermore, no common evaluation corpora exist for many alternative languages. To some extent, this problem has been circumvented by building small theory-specific treebanks (with the obvious drawbacks for supervised training and inter-comparability). In sum, the growing field in stochastic parsing with alternative syntactic models or languages other than English faces problems in benchmarking against the established Gold Standard.

As a consequence, the best-known stochastic parsers are trained for Penn Treebank English. Yet, to validate these parsers on a broader basis, it has to be evaluated how well these stochastic models carry over to languages with e.g. free word order, intricate long-distance phenomena, pro-drop properties, and agglutinative or clitic languages. Again, this presupposes the availability of annotated corpora and evaluation schemes appropriate to cover a broad range of diverse language types.

3. Towards a New Gold Standard

The current situation in stochastic parsing, as well as prospects for its future development, calls for a new and uniform scheme for parser evaluation which covers both shallow and deep grammars, different syntactic frameworks, and different language types.

What is needed is an annotation scheme bridging structural differences across diverse languages and frameworks. In practice, many researchers have been using their own evaluation metrics which, despite divergences, bear some common ground, namely higher-level syntactic annotations such as grammatical relations, dependencies, or subcategorization frames (Beil et al. (1999), Carroll et al. (2000), Collins et al. (1999), Hockenmaier (2001), etc). Such basic syntactic relations build on crucial, but underlying structural constraints, yet provide more abstract, functional information.

This information is not only an appropriate level of abstraction to bridge structural differences between languages and higher-level syntactic theories, but moreover, provides a basis for evaluation of partial, more shallow analysis systems, at a higher level of representation. For example, if the evaluation is against grammatical relation rather than phrase structure information, partial parsers extracting functional relations can be evaluated within the same setup as full parsers.

Starting from this state of affairs, one of the aims of the workshop will be to provide a forum for researchers in the field to discuss (define and agree on) a new, uniform evaluation metric which provides a basis for comparison between different parsing systems, syntactic frameworks and stochastic models, and how well they extend to languages of different types.

Definition of a new evaluation standard could be restrictive and flexible at the same time: flexible in that training can exploit fine-grained annotations of richer syntactic frameworks; and restrictive in that diverging analyses are then to be mapped to uniform (more coarse-grained) annotations for standardized evaluation.

4. Starting an Initiative

A previous LREC-hosted workshop on parser evaluation in 1998 in Granada brought together a number of people advocating parser evaluation based on dependencies or grammatical relations (Carroll and Briscoe (1998), Lin (1998), Bangalore et al. (1998)). The consensus of the concluding discussion at that workshop was that there is much common ground between these approaches, and that they constitute a viable alternative to the PARSEVAL measures.

In the meantime, as described above, many more corpora are under construction and novel stochastic parsing schemes are being developed, which call for an initiative for establishing a new, agreed-on evaluation standard for parsing which allows for comparison and benchmarking across alternative models and different language types.

The workshop is intended to bring together four parties: researchers in stochastic parsing, builders of annotated corpora, representatives from different syntactic frameworks, and groups with interests in and proposals for parser evaluation. As a kick-off initiative, the workshop should lead to collaborative efforts to work out a new evaluation metric, and to start initiatives for building or deriving sufficiently large evaluation corpora, and possibly, large training corpora according to the new metric.

In conclusion, stochastic parsing has now developed to a stage where new methods are emerging, both in terms of underlying frameworks and languages covered. These need to be brought together by means of a new evaluation metric to prepare the new generation of stochastic parsing.

5. Workshop Programme

The workshop comprises thematic papers focussing on benchmarking of stochastic parsing, parser evaluation, design of annotation schemes covering different languages, and different frameworks, as well as creation of high-quality evaluation corpora.

Intended as a forum for discussion, the workshop programme consists of paper presentations with discussion sessions and a panel, where important results of the workshop are summarized and discussed.

In the final session we intend to wrap-up, and plan a kick-off initiative leading to concrete action plans and the creation of working groups, as well as planning for future coordination. To maintain the momentum of this initiative we will work towards setting up a parsing competition based on new standard evaluation corpora and evaluation metric.

References

Srinivas Bangalore, Anoop Sarkar, Christine Doran, and Beth Ann Hockey. 1998. Grammar and parser evaluation in the xtag project. In *Workshop on the Evaluation of Parsing Systems*, LREC, Granada.

Franz Beil, Glenn Carroll, Detlef Prescher, Stefan Riezler, and Mats Rooth. 1999. Inside-outside estimation of a lexicalized PCFG for German. In *Proceedings of ACL'99*, College Park, MD.

Ezra Black. 1992. Meeting of interest group on evaluation of broad-coverage grammars of English. LINGUIST List 3.587, http://www.linguistlist.org/issues/3/3-587.html.

Rens Bod. 2001. What is the minimal set of fragments that achieves maximal parse accuracy? In *Proceedings of ACL-2001*.

Gosse Bouma, Gertjan van Noord, and Robert Malouf. 2001. Alpino: Wide-coverage computational analysis of dutch. In *Computational Linguistics in The Netherlands 2000*.

John Carroll and Ted Briscoe. 1998. A survey of parser evaluation methods. In *Workshop on the Evaluation of Parsing Systems*, LREC, Granada.

Eugene Charniak. 1996. Tree-bank grammars. Technical Report CS-96-02, Brown University.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics (ANLP-NAACL 2000)*, Seattle, WA.

J. Chen, S. Bangalore, and K. Vijay-Shanker. 1999. New models for improving supertag disambiguation. In *Proceedings of the 9th Conference of the European Chapter of the Association for Computational Linguistics*.

M. Collins, J. Hajic, L. Ramshaw, and Ch. Tillman. 1999. A Statistical Parser for Czech. In *Proceedings of ACL 99*.

Michael Collins. 1996. A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 34th Annual Meeting of the Association for Computational Linguistics (ACL'96)*, Santa Cruz, CA.

Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics (ACL'97)*, Madrid.

T. Fujisaki, F. Jelinek, J. Cocke, E. Black, and T. Nishino. 1989. A probabilistic method for sentence disambiguation. In *Proceedings of the 1st International Workshop on Parsing Technologies*.

J. Hajic. 1998. Building a syntactically annotated corpus: The prague dependency treebank. Issues of Valency and Meaning. Studies in Honour of Jarmila Panevova.

Julia Hockenmaier. 2001. Statistical parsing for ccg with simple generative models. In *Student Research Workshop of the 39th ACL/10th EACL*.

Mark Johnson, Stuart Geman, Stephen Canon, Zhiyi Chi, and Stefan Riezler. 1999. Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99)*, College Park, MD.

D. Lin. 1998. Dependency-based evaluation of minipar. In *Workshop on the Evaluation of Parsing Systems*, LREC, Granada.

David M. Magerman. 1995. Statistical decision-tree models for parsing. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics (ACL'95)*, Cambridge, MA.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of english: The Penn treebank. *Computational Linguistics*, 19(2):313–330.

S. Riezler, D. Prescher, J. Kuhn, and M. Johnson. 2000. Lexicalized stochastic modeling of constraint-based grammars using log-linear measures and EM training. In *Proc. of ACL-2000*.

G. Sampson, R. Haigh, and E. Atwell. 1989. Natural language analysis by stochastic optimization: a progress report on project april. *Journal of Experimental and Theoretical Artificial Intelligence*.

R. Sharman, F. Jelinek, and R. Mercer. 1990. Generating a grammar for statistical training. In *Proceedings of the DARPA Speech and Natural Language Workshop*.

Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proceedings of the Fifth Conference on Applied Natural Language Processing ANLP-97*, Washington, DC.

Relational Evaluation Schemes

Ted Briscoe*, John Carroll†, Jonathan Graham*, Ann Copestake*

*Computer Laboratory
University of Cambridge
{Ted.Briscoe, Ann.Copestake}@cl.cam.ac.uk

†Cognitive and Computing Sciences
University of Sussex
John.Carroll@cogs.susx.ac.uk

Abstract

We describe extensions to a scheme for evaluating parse selection accuracy based on named grammatical relations between lemmatised lexical heads. The scheme is intended to directly reflect the task of recovering grammatical and logical relations, rather than more arbitrary details of tree topology. There is a manually annotated test suite of 500 sentences which has been used by several groups to perform evaluations. We are developing software to create larger test suites automatically from existing treebanks. We are considering alternative relational annotations which draw a clearer distinction between grammatical and logical relations in order to overcome limitations of the current proposal.

1. Introduction

We have developed a scheme for evaluating parse selection accuracy based on named grammatical relations between lemmatised lexical heads. The scheme is intended to directly reflect the task of recovering semantic relations, rather than more arbitrary details of tree topology—as with the PARSEVAL scheme, which has been criticised frequently for the opaque relationship between its measures and such relations (Carroll *et al.*, 1998; Magerman, 1995; Srinivas, 1997). Carroll *et al.* (1998) provide more detailed motivation and comparison with other extant schemes.

Carroll *et al.* (1999, 2002 in press) report the development of a test suite of 500 sentences annotated with grammatical relations, the specification of the relations, and their criteria of application. The set of named relations are organised as a subsumption hierarchy in which, for example, subj(ect) underspecifies n(on)c(lausal)subj(ect). There are a total of 15 fully specified relations, however, many of these can be further subclassified; for example, subj relations have an initial-gr slot used to encode whether the syntactic subject is logical object (as in passive) and for other marked subjects (such as in locative inversion). Thus a fully specified GR might look like (ncsubj marry couple obj) to encode the subj relation in *The couple were married in August*, and the GR annotation of each sentence of the test suite consists of a set of GR *n*-tuples. Figure 1 gives the full set of named relations represented as a subsumption hierarchy. The most generic relation between a head and a dependent is dependent. Where the relationship between the two is known more precisely, relations further down the hierarchy can be used, for example mod(ifier) or arg(ument). Relations mod, arg_mod, aux, clausal, and their descendants have slots filled by a type, a head, and its dependent; arg_mod has an additional fourth slot initial_gr. Descendants of subj, and also dobj have the three slots head, dependent, and initial_gr. Relation conj has a type slot and one or more head slots. The x and c prefixes to relation names differentiate clausal control alternatives.

When the proprietor dies, the establishment should become a corporation until it is either acquired by another proprietor or the government decides to drop it.

```
(ncsubj die proprietor _)
(ncsubj become establishment _)
(xcomp _ become corporation)
(ncsubj acquire it obj)
(arg_mod by acquire proprietor subj)
(ncmod _ acquire either)
(ncsubj decide government _)
(xcomp to decide drop)
(ncsubj drop government _)
(dobj drop it _)
(cmod when become die)
(cmod until become acquire)
(cmod until become decide)
(detmod _ proprietor the)
(detmod _ establishment the)
(detmod _ corporation a)
(detmod _ proprietor another)
(detmod _ government the)
(aux _ become shall)
(aux _ acquire be)
(conj or acquire decide)
```

Figure 2: Grammatical relation sample annotation.

Figure 2 shows the GR encoding of a sentence from the Susanne corpus.

The evaluation metric uses the standard precision and recall and F_α measures over sets of such GRs. Carroll and Briscoe (2001) also make use of weighted recall and precision (as implemented in the PARSEVAL software) to evaluate systems capable of returning *n*-best sets of weighted GRs. The software makes provision for both averaged scores over all relations as well as scores by named relation. It also supports partial scoring in terms of non-leaf named relations which underspecify leaf relations. The current specification of the

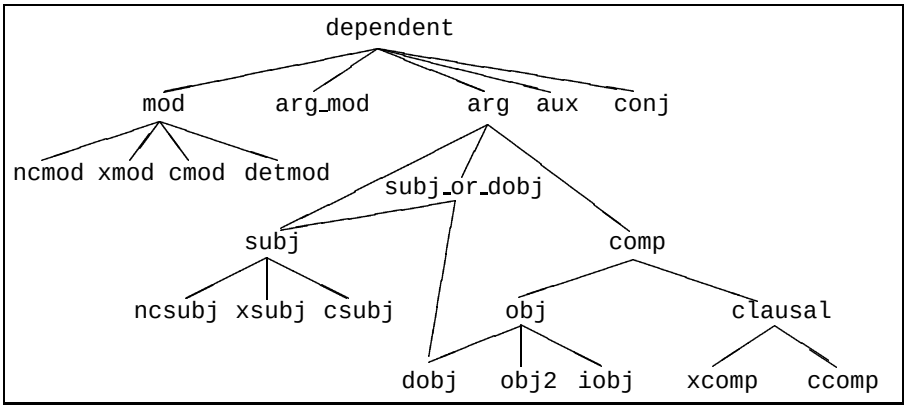


Figure 1: Grammatical relation hierarchy.

scheme along with the test suite and evaluation software (implemented in Common Lisp) is available from http://www.cogs.susx.ac.uk/lab/nlp/carroll/greval.html

Evaluation of stochastic parsers using relational schemes similar to our proposal is becoming more common (e.g. Collins, 1999; Lin, 1998; Srinivas, 2000). However, comparison across such results is hampered by the fact that the set of relations extracted is not standardised across these schemes, and it is clear that some relations (e.g. that between determiners and head nouns) are much easier to extract than others (e.g. control relations in predicative complements), as can be seen, for example, from the separate and divergent precision / recall results by named relation reported by Carroll *et al.* (1999). This makes meaningful comparison of 'headline results' such as mean overall F_1 measures very hard. Our scheme attempts to ameliorate these problems by supporting different levels of granularity within named relations (ncsubj / csub / xsubj $\subset$ subj) and encouraging not only the reporting of overall mean precision / recall scores, but also separate scores for each named relation.

In the rest of this paper we describe ongoing efforts to improve the evaluation scheme and enlarge the annotated test suite(s).

2. Divergent system output representations

There remain several infelicities in the current scheme that are a consequence of the method of factoring information into distinct relations which, in fact, still encode composites of information. For example, a system which clearly separates categorial constituency and functional information, such as one based on LFG, might choose to map F-structure SUBJ relations to subj in our scheme. A more constituency based parser might map NPs immediately dominated by S and preceding a VP to ncsubj, and Ss in the same configuration to csubj. Superficially the latter system is extracting more information because the relation name encodes categorial as well as relational information. The current scoring metric also assigns a penalty to systems that do not recover fully-specified (leaf) relations. However, for either system to score in the evaluation the subj relation most hold between lemmatised heads of the appropriate type, so the distinction between clausal and non-clausal subjects is maintained in both, since clausal subjects have verbal heads. On the other hand a system which systematically returned subj-or-dobj relations, as opposed to a leaf subj or obj one, would clearly be losing significant information pertinent to recovery of underlying logical relations.

There are many other cases of divergent encoding of aspects of categorial and functional information: for example, a LFG system will clearly distinguish clausal and predicative complements at F-structure corresponding directly to the xcomp / ccomp distinction in our relational scheme. However, a parser that represents such complements as clauses (S nodes) with or without an empty (PRO) NP subject, as in the Penn WSJ Treebank, would need to utilise a more complex (non-local) mapping from tree topology and node labels to named relations in order to maintain the xcomp / ccomp distinction. However, in this case, the easier underspecification to comp is genuinely significant since in either case the relation will hold between the same lexical (verbal) heads.

There are, in principle, two ways of dealing with such divergences. The first is to complicate the mapping from system output to named relations so that the specific set of leaf relations identified in the current scheme is recovered, if it is deducible from the total system output. The second is to modify the scoring metric so that informationally insignificant underspecification is not penalised. In some cases, such as the LFG system SUBJ case described above, the latter step will be much easier. In the new version of the specification and evaluation measure, we will attempt to identify such cases and parameterise the evaluation software to compute scores appropriately, as well as provide more specific guidance on mapping of named relations to the output of extant systems. This should improve the validity of cross-system evaluation. However, problems of this type are likely to emerge for each new system representation considered, so this is likely to be an ongoing process requiring judgement on the part of evaluators coupled with explicit description of decisions made alongside reported socres.

Provision of a flexible software system for mapping from parser output representations to factored relational ones may also ameliorate this class of problems (see section 5.). In particular, where a specific choice of system output representation necessitates a more complex mapping to leaf relations in our scheme, it would facilitate fair and

feasible cross-system comparison if the evaluation scheme provided software that would recover the named leaf relations from the system output. Once again, each new system representation is likely to throw up new problems of this type, so flexible and easily parameterisable software will be more useful.

3. Surface / logical form divergence

The current annotation scheme attempts to stay close to surface grammatical structure, while also encoding divergence from predicate-argument structure / logical form. Divergence is currently encoded using two distinct mechanisms for different types of cases. Extra slots in named relations are used to indicate surface / underlying logical relation divergences, as with subj discussed in section 1. An additional relation is used for coordination (conj) to indicate how the conjunction scopes over the individual conjuncts.

One conspicuous area where the current scheme is inadequate is with equative and comparative constructions, which occur quite frequently in the 500 sentence test suite. Semantically, it is standard to treat *more* and *as*, etc as generalised quantifiers over propositions so that an example like

> *GR evaluation is more / as attractive than / as PARSEVAL*

is represented (very crudely) as

> more′(is-attr′(GReval′), is-attr′(PARSEVAL′))

This example, however, is annotated by the GRs

> (ncmod _ attractive more)
> (ncmod than attractive PARSEVAL)

However, in general, the GR annotation of such constructions is variable because of the varied surface syntactic location of *more* and *as* and also because of the optionality of and degree of ellipsis in the *than / as* constituent. Furthermore, because of the divergence between surface form and logical form the current annotations give little indication of whether a system would be capable of outputting an appropriate logical form. Replacing the current annotation with one close to the target logical form would undermine the scheme, since most extant stochastic parsers would be unable to generate such a representation.

One alternative is to additionally annotate such constructions with construction-specific named relations. This could be based on the approach to coordination, where the named relation

> (conj conj-type conjunct-heads+)

is used in addition to distributing the conjunct heads over multiple occurrences of the relation over the coordinate construction. For comparatives and equatives, we could add a relation like

> (compequ as/more/... attractive GReval PARSEVAL)

encoding the type of comparison, the predicate of comparison, and the arguments to this predicate.

There are undoubtedly further constructions, beyond coordination and comparatives / equatives that merit some such treatment. The advantage of adding additional construction-specific named relations that encode the same phenomena from different perspectives is that the resulting annotation will support a graded and fine-grained evaluation of the extent to which a specific system can support recovery of underlying logical form / predicate-argument structure in addition to surface grammatical relations. The disadvantage of this approach is that the scheme is likely to become more complex, and thus its recovery from any specific parser representation more time-consuming. In addition, the encoding of the underlying logical relations in the GR scheme has already spawned two divergent mechanisms, and may well require more.

4. MRS-style annotation scheme

A second and more complex but potentially more thorough approach to the issue of surface / logical form divergence is to bleach the current GR scheme of all attempts to represent such mismatches and instead define a factored and underspecified semantic annotation scheme to be used in tandem with GR annotation. The approach to underspecified logical representation developed by Copestake *et al.* (2001) can be extended to allow semantics to be underspecified to a much greater degree. In this extension of minimal recursion semantics (MRS), a Parsons-style notation (Parsons, 1990) is used, with explicit equalities representing variable bindings. For instance, from

> *The couple were married.*

a particular parsing system might return

> (ARGN u1 u2)
> (marry u3)
> (couple u4)

However, the fully specified test suite annotation would be

> (ARG2 e1 x4)
> (marry e2)
> (couple x3)
> e1 = e2
> x3 = x4

where ARG2 is formally a specialisation of ARGN, and the equalities and variable sorts also add information.

Potentially, this would allow us to dispense with complications like init-gr fields in the GR annotation and provide a principled basis for a graded evaluation of the recovery of logical form. The disadvantage over the further extension of the existing scheme is that two stages of extraction from specific system output are now required, the matching operations and scoring metrics become more complex, and the ability to do a graded evaluation of recovery of both grammatical and logical relations may be somewhat undermined.

```
try
  {
   while (dd)
    {
     String s = readWord(W);
     setS += 1;

     if (c==0) dd = false;

     if (s.equals("S"))
      {
       if (domprecedes("S", "NP",
                       "VP", setS))
        { String head = mainverb(setvp);
          String dependent =
            righthead("NP", "N-", setnp);
          String objslot =
            ispassive(setvp);
          System.out.println(
                   "(ncsubj " + head + "
                   " + dependent + "
                   " + objslot + ")");
        }
       }
    }
  }
```

Figure 3: The ncsubj extraction class.

5. Enlarging and improving the test suite(s)

The current test suite of 500 sentences is too small, but was still labour-intensive to create semi-automatically. Consequently, it contains a number of inadequacies: tokenisation of multiwords is somewhat arbitrary, some relations which should be included are systematically omitted (e.g. predicative XP complements of *be* have not been annotated with their controlled subjects), quotation marks have been systematically removed, and so forth. The next release will attempt to remove these inadequacies. However, it is clear that we also need a method for annotating much more data efficiently. To this end we have been developing a generic system, implemented in JAVA, that can be applied to existing treebanks to extract relational information (Graham, 2002). This system can, in principle, extract GRs in the current or related schemes, or even (possibly underspecified) MRSs. It can be parameterised for different extant treebanks, such as Penn Treebank-II or Susanne, and requires a set of declarative rules expressed in terms of tree topology and node labels for each named relation. The system has been designed to process labelled trees looking for relations defined ultimately in terms of (immediate) dominance and (immediate) precedence efficiently. It has been tested on a subset of GRs, concentrating particularly on the subj sub-hierarchy. A fragment of the class for ncsubj encoding relevant constraints is shown in Figure 3, giving a sense of the degree of parameterisation required for different representations. Running a first prototype of the GR extractor on the 30 million word automatically annotated WSJ BLLIP corpus distributed by the LDC results in estimated recovery of 86% of ncsubj and dobj relations with a precision of 84%, taking around 3 hours CPU time on standard hardware.

This system will facilitate rapid automatic construction of relational annotation according to specified input and output scheme(s) up to the limit of what is currently represented in treebanks and system output. Our longer term plan is to make this software, and a number of rule sets implemented in it, available as part of the evaluation scheme. This should facilitate both the construction of test data and the mapping of system output to the required format.

6. Conclusions

Relational schemes for parser evaluation are gaining in popularity over the exclusive use of PARSEVAL or similar tree topology based measures. We hope that the ongoing work reported here will facilitate further cross-system and within-system relational evaluation. To this end, we are developing test suites and software to support flexible mapping from system and treebank output to relational encodings of grammatical and underlying logical relations, and actively seeking feedback from the community on weaknesses of our current encoding scheme and evaluation measures and errors in our current test set.

Acknowledgements

We would like to thank Ron Kaplan for carefully documenting many errors and inconsistencies in our semi-automatic annotation of the 500 word test suite. The GR encoding scheme was heavily influenced by the EAGLES encoding scheme, primarily developed by Antonio Sanfilippo. We would also like to thank Anne Abeillé and Srinivas Bangalore for useful discussions. This work was partially supported by the EPSRC-funded RASP project (grants GR/N36462 and GR/N36493).

References

Carroll, J. and E. Briscoe (2001) 'High precision extraction of grammatical relations', *Proceedings of the 7th ACL/SIGPARSE International Workshop on Parsing Technologies (IWPT'01)*, Beijing, China, pp. 78–89.

Carroll, J., E. Briscoe and A. Sanfilippo (1998) 'Parser evaluation: a survey and a new proposal', *Proceedings of the 1st International Conference on Language Resources and Evaluation*, Granada, pp. 447–454.

Carroll, J., G. Minnen and E. Briscoe (1999) 'Corpus annotation for parser evaluation', *Proceedings of the EACL-99 Post-Conference Workshop on Linguistically Interpreted Corpora (LINC'99)*, Bergen, Norway, pp. 35–41.

Carroll, J., G. Minnen and E. Briscoe (2002, in press) 'Parser evaluation using a grammatical relation annotation scheme' in Abeille, A. (ed.), *Treebanks: Building and Using Syntactically Annotated Corpora*, Dordrecht: Kluwer.

Collins, M. (1999) *Head-driven Statistical Models for Natural Language Parsing*, PhD Dissertation, University of Pennsylvania.

Copestake, A., A. Lascarides and D. Flickinger (2001) 'An algebra for semantic construction in constraint-based grammars', *Proceedings of the 39th Annual Meeting of the Association for Computational Linguistics*, Toulouse, France, pp. 132–139.

Graham, J. (2002, in preparation) *From Treebank to Lexicon*, DPhil Dissertation, University of Cambridge, Computer Laboratory.

Lin, D. (1998) 'Dependency-based evaluation of MINIPAR', *Proceedings of the The Evaluation of Parsing Systems: Workshop at the 1st International Conference on Language resources and Evaluation*, Granada, Spain.

Magerman, D. (1995) *Natural Language Parsing as Statistical Pattern Recognition*, PhD Dissertation, Stanford University.

Parsons, T. (1990) *Events in the Semantics of English*, MIT Press, Cambridge, MA.

Srinivas, B. (1997) *Complexity of Lexical Descriptions and its Relevance to Partial Parsing*, PhD Dissertation, University of Pennsylvania.

Srinivas, B. (2000) 'A lightweight dependency analyzer', *Natural Language Engineering, vol.6.2*, 113–138.

Towards a Dependency-Oriented Evaluation for Partial Parsing

Sandra Kübler, Heike Telljohann

Seminar für Sprachwissenschaft
Wilhelmstr. 113
D-72074 Tübingen
Germany
{kuebler,hschulz}@sfs.uni-tuebingen.de

Abstract

Quantitative evaluation of parsers has traditionally centered around the PARSEVAL measures of *crossing brackets, (labeled) precision,* and *(labeled) recall*. However, it is well known that these measures do not give an accurate picture of the quality of the parser's output. Furthermore, we will show that they are especially unsuited for partial parsers. In recent years, research has concentrated on dependency-based evaluation measures. We will show in this paper that such a dependency-based evaluation scheme is particularly suitable for partial parsers. TüBa-D, the treebank used here for evaluation, contains all the necessary dependency information so that the conversion of trees into a dependency structure does not have to rely on heuristics. Therefore, the dependency representations are not only reliable, they are also linguistically motivated and can be used for linguistic purposes.

1. Introduction

Quantitative evaluation of parsers has traditionally centered around the PARSEVAL measures of *crossing brackets, (labeled) precision,* and *(labeled) recall* (Black et al., 1991). However, it is well known that these measures do not give an accurate picture of the quality of the parser's output (cf. Manning and Schütze (1999)), e.g. in cases of attachment errors. Additionally, many phenomena like negation or unary branches are ignored in the original measures in order to allow a comparison between parsers that use incompatible grammars. For this reason, research in recent years has concentrated on dependency-based evaluation measures (cf. e.g. Lin (1995), Lin (1998)). We will show in this paper that such a dependency-based evaluation scheme is particularly suitable for partial parsers since it does not lead to disproportionately high losses in precision and recall for partial parses. Furthermore, the dependency representations are not only reliable, they are also linguistically motivated and can be used for linguistic purposes since the treebank used here for evaluation contains all the necessary dependency information.

2. Deficiencies of Constituency-Based Precision and Recall

It is a well known fact that the PARSEVAL measures do not always give an accurate picture of the quality of a parser's output. Carroll and Brisoce (1996), for example, note that the *crossing brackets* measure is too lenient in case of errors involving the disambiguation of arguments and adjuncts, which in some cases are not recognized as errors. The failure to attach a constituent which should be embedded n levels deep leads to n crossing errors, while this constituent may not be very important to the overall structure. Manning and Schütze (1999) show that this behavior is mirrored in *precision* and *recall*: If a constituent is attached very high in a complex right branching structure, but the parser attached it at a lower point in the structure, both precision and recall will be greatly diminished. An example of such a parsing error for the sentence "ich nehme den Zug nach Frankfurt an der Oder" (I will take the train to Frankfort on the Oder) is shown in Figure 1[1]. There the prepositional phrase "an der Oder" is erroneously grouped as an adjunct of the verb instead of being attached as a postmodifier to the noun phrase "nach Frankfurt" (cf. the following section for a description of the annotation scheme). The correct tree is shown in Figure 2. When using the PARSEVAL measures, the output of the parser shown in Figure 1 results in $10/13 = 76.92\%$ recall[2] and $10/12 = 83.33\%$ precision, the only error being the wrong attachment of the last prepositional phrase.

The same behavior can be observed when the parser attaches a constituent very high in a complex right branching structure instead of very low, or if the constituent is not attached at all. The latter is often the case for chunk parsers (Abney, 1991; Abney, 1996) or partial parsers (cf. e.g. Aït-Mokhtar and Chanod (1997)). These parsers generally aim at annotating only partial, reliably discoverable tree structures, i.e. base phrases and clausal structures. Postmodifications are generally not attached since this decision cannot be taken reliably based on very limited local context. TüSBL (Kübler and Hinrichs, 2001a; Kübler and Hinrichs, 2001b), e.g., a similarity-based parser for German, annotates syntactic structures including function-argument structure in a two-level architecture: in the first phase, a deterministic chunk parser (Abney, 1996) is used to anal-

[1] All syntactic trees shown in this paper follow the data format for trees defined by the NEGRA project of the Sonderforschungsbereich 378 at the University of the Saarland, Saarbrücken. They were printed by the NEGRA graphical annotation tool *Annotate* (Brants and Skut, 1998; Plaehn, 1998).

[2] Contrary to the original PARSEVAL measures, we do count the root node as well since there exist different root nodes in the annotation scheme, and there are cases when a sentence in the treebank is annotated with more than one tree (e.g. interjective utterances).

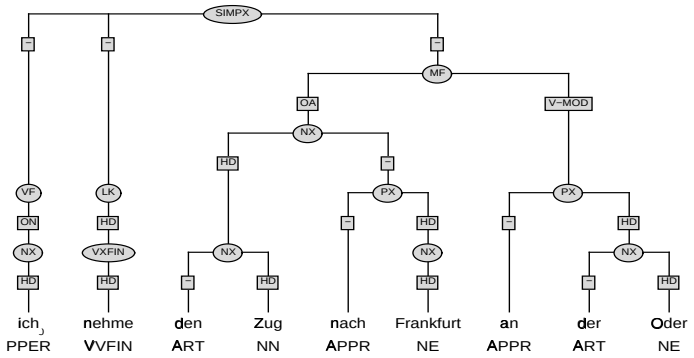


Figure 1: Wrong attachment of the prepositional phrase “an der Oder”.

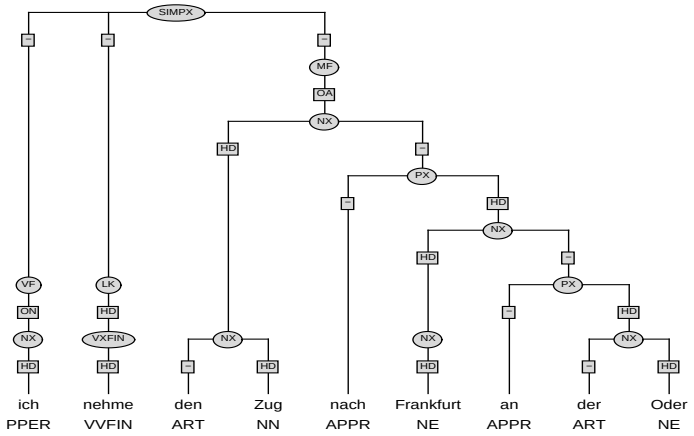


Figure 2: Correct attachment of the prepositional phrase “an der Oder”.

yse major syntactic constituents such as non-recursive base phrases and simplex clauses. As a consequence, dependency relations between individual chunks, such as grammatical functions or modification relations, within a clause remain unspecified. In the second step, the attachment ambiguities are resolved, and the partial annotation of the first step are enriched by dependency information. A typical output of this phase is shown in Figure 3. The second phase of analysis is based on a similarity-based machine learning approach, which uses a similarity metric to retrieve the most similar sentence to the input sentence from the instance base and adapts the respective tree to the input sentence. (For a more detailed description of the algorithm cf. Kübler and Hinrichs (2001a) and Kübler and Hinrichs (2001b).) The parser is designed to prefer partial analyses over uncertain ones. In some cases, this strategy leads to unattached phrases, mostly at the end of sentences, which results in high losses in precision and recall. We therefore propose to use a dependency-based evaluation as described by Lin (1995) and Lin (1998), in which both the gold standard and the parser’s output are transformed into dependencies and then compared on the basis of dependencies rather than on the basis of the constituent structure.

3. The TüBA-D Treebank

The dependency-based evaluation was based on the German corpus TüBa-D (Stegmann et al., 2000; Hinrichs et al., 2000a; Hinrichs et al., 2000b), which consists of approximately 38,000 syntactically annotated sentences. For this treebank, a theory-neutral and surface-oriented annotation scheme has been adopted that is inspired by the notion of topological fields – in the sense of Herling (1821), Erdmann (1886), Drach (1937), Reis (1980), and Höhle (1985) – and enriched by a level of predicate-argument structure, which guides the conversion into dependencies. The linguistic annotations pertain to the levels of morpho-syntax (part-of-speech tagging) (Schiller et al., 1995), syntactic phrase structure, and function-argument structure.

The tree structure contains different types of syntactic information in the following way: As the primary clustering principle the theory of topological fields (Höhle, 1985) is adopted, which captures the fundamental word order regularities of German sentence structure. In verb-second sentences, the finite verb constitutes the left sentence bracket (LK) and the verb complex the right sentence bracket (VC). This sentence bracket divides the sentence into the following topological order of fields: initial field (VF), LK, middle field (MF), VC, final field (NF). This structuring concept in addition favors bracketings that do not rely on crossing branches and traces to describe discontinuous dependencies.

Below this level of annotation, i.e. strictly within the bounds of topological fields, a phrase level of predicate-argument structure is established with its own descriptive

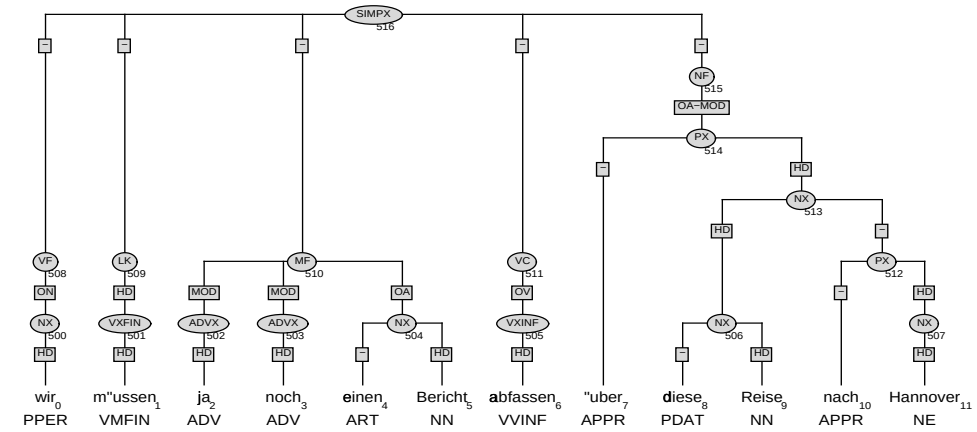


Figure 3: A tree annotated according to the TüBa-D treebank annotation scheme.

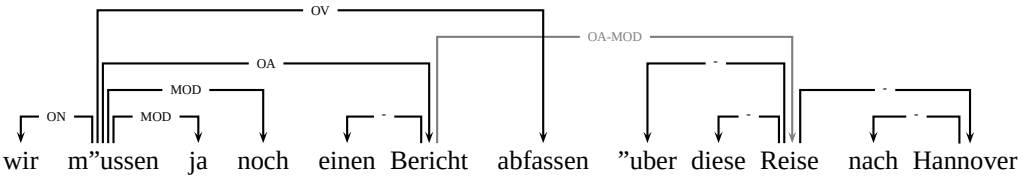


Figure 4: The dependency structure of the tree in Figure 3. The crossing dependency is shown in gray.

inventory based on a minimal set of assumptions concerning constituenthood, phrase attachment, and grammatical functions that have to be captured by any syntactic theory: nodes are labeled with syntactic categories on four different levels of annotation (sentence level, field level, phrase level, and lexical level), edges denote grammatical functions on the phrase level (i.e. immediately below the topological fields) and head/non-head distinctions within phrases. The integrated constituent analysis with its information about grammatical functions ensures that the resulting dependency structures are linguistically motivated and can also be used for linguistic purposes.

An example of such a tree for the sentence "wir müssen ja noch einen Bericht abfassen über diese Reise nach Hannover" (we still need to write a report on this journey to Hanover) is shown in Figure 3 (for more information about the annotation scheme cf. Stegmann et al. (2000)).

Two specific edge labels denote whether a constituent has the function of a head (HD), e.g. a phrase (NX, PX, ADJX, ADVX, VXFIN, VXINF), or a non-head (-), e.g. a determiner or a modifier attached to a phrase. On any annotation level, there is at most one head. The head of a sentence structure (e.g. SIMPX) is always the finite verb, which can be found in the left sentence bracket (LK). If there is no LK, the head is represented by the finite verb in the verb complex (VC). In coordinations, each conjunct depends on the head of the whole construction. Therefore, conjuncts are denoted with the non-head edge label.

The constituents below the topological fields are assigned grammatical functions. A subset of the edge label set consists of labels denoting the grammatical function of complements and modifiers, which depend on the head of the sentence. Another subset consists of labels determining long distance dependencies among these complements or modifiers as well as between conjuncts of split-up coordinations.

In Figure 3, e.g., the first constituent is marked as subject (ON), the finite verb is the head (HD), the two adverbs are modifiers (MOD), and the second noun phrase represents the direct object (OA). The constituent following the verb complex modifies the direct object (OA-MOD). Since the annotation scheme for the TüBa-D treebank facilitates a theory-neutral and surface-oriented representation of syntactic trees, this long distance relation is marked by the label OA-MOD (modifier of the accusative object) which refers to OA (accusative object) in the same tree; instead of using crossing branches and traces. This shows that long distance dependencies, which can even go beyond the border of topological fields, are encoded by special naming conventions for edge labels. Unambiguous edge labels, referring to exactly one non-adjacent constituent in the same tree, are used either for long distance modifications (*X*-MOD) like in the example above or for the rightmost conjunct of split-up coordinations (*X*K) (for an example cf. Figure 5). In both patterns, *X* is a variable for the grammatical function of the constituent to which it refers.

4. Converting TüBa-D into Dependencies

For TüBa-D, the conversion of the constituent structure into dependencies is in general determined by the head/non-head distinction in the tree. The dependency relations are labeled with the functional labels of the governed constituents. Using these strategies, the tree shown in Figure 3 is converted into the dependency structure in Figure 4. Here, the noun phrase "einen Bericht" is converted into one dependency relation, which denotes that the noun "Bericht"

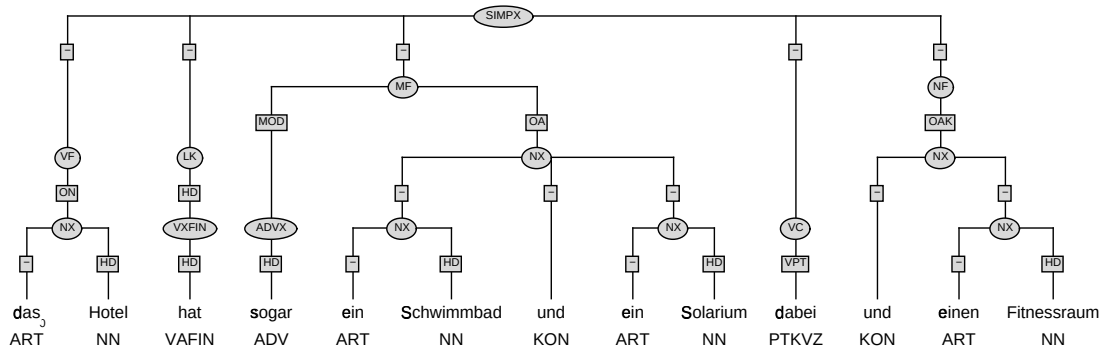


Figure 5: A complex coordination of noun phrases.

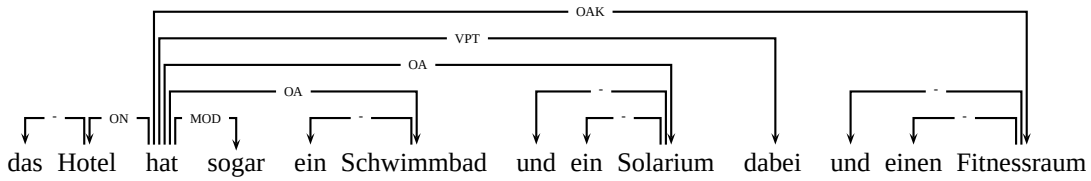


Figure 6: The dependency structure of the tree in Figure 5.

governs the article “den”.

It is evident that the dependency structure contains two different types of dependencies: head/non-head dependencies within phrases (-) and dependencies from the finite verb, i.e. from the head of the clause, to its complements and adjuncts, which are labeled by the grammatical functions of the governed constituents (ON, MOD, OA, OV). This is why e.g. the direct object “einen Bericht” is represented as a dependent of the modal verb “müssen” although it constitutes an argument of the embedded main verb “abfassen”. However, the dependency relations among the finite verb and the (possibly multiple) infinite verbs is explicitly annotated in the syntactic and therefore in the dependency structure. And since information about clausal boundaries is present in the trees, even in this surface-oriented structure, the predicate-argument structure can be recovered.

The long-distance dependency between the direct object and its modifying prepositional phrase was modeled in the syntactic tree by the function label “OA-MOD” instead of by the attachment of the prepositional phrase to the direct object because the latter would have resulted in a *crossing branch*. In the dependency structure, this restriction is suspended, and the dependency is explicitly marked and has now resulted in crossing dependencies. Note that this is the only type of phrase-internal dependency that is not labeled by the head/non-head distinction but by unambiguous labels which denote their specific reference.

Since head information is present on all levels for the majority of constituents, specific decisions for determining dependency have to be taken only in the few cases when dependency relations are not clearly defined in the tree structure, i.e. for the following syntactic phenomena:

1. Conjunctions within coordinations do not depend on the head of the whole construction. Therefore, they are attached to the conjunct on their right hand side. An example of such a coordination is shown in Figure 5, the corresponding dependency structure in Figure 6. Here, the third conjunct is positioned after the verb complex and thus is assigned the label “OAK”.
   Similar constructions with a preposition instead of a conjunction like “der achte bis neunte” (the eighth until the ninth) are treated in the same way. In order to stress the identical syntactic status of conjuncts, all conjuncts depend on the head governing the coordination. This analysis is in contrast to Lin (1998), who relies on the *Single Head Assumption* and proposes a dependency relation between the first and the second conjunct.

2. Sentence-initial coordinative particles such as “und” (and) or “oder” (or) in the KOORD-field depend on the head of the sentence.

3. The annotation of prepositional phrases in the syntactic trees is based on the principles of Dependency Grammar (Heringer, 1996); therefore, the noun phrase constitutes the head. For an example of the dependency structure of a prepositional phrase cf. the phrase “nach Hannover” in Figure 4. Circumpositions and postpositions are treated similarly.

4. The single elements of proper names, split cardinal numbers, the spelling of words, and complex conjunctions in the C-field, e.g. “so daß” (so that), are attached on the same level carrying a non-head edge label to indicate that there is no obvious dependency relation between them. Therefore, they are treated like conjuncts in coordinations.

5. A heuristic analysis has to be applied when long distance relations are underspecified – a MOD-MOD la-

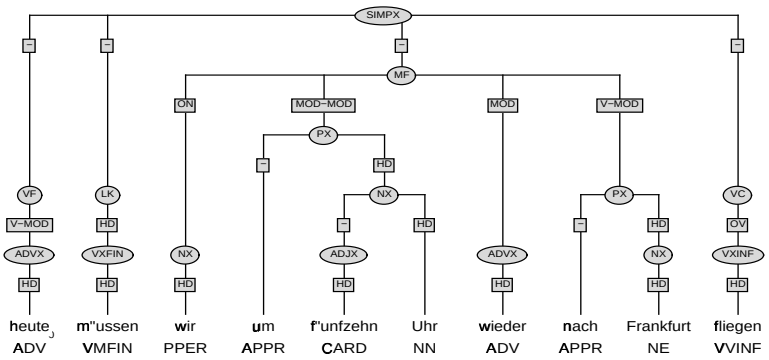


Figure 7: An ambiguous long-distance modifier: MOD-MOD.

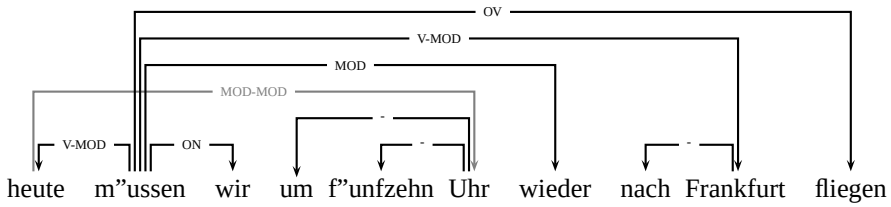


Figure 8: Resolved dependencies for ambiguous long-distance modifiers. The crossing dependency is shown in gray.

bel (modifier of a modifier), e.g., may refer to one of several modifiers in the sentence, such as for the sentence “heute müssen wir um fünfzehn Uhr wieder nach Frankfurt fliegen” (today we need to fly again to Frankfort) in Figure 7. Here, the long-distance modifier MOD-MOD might modify the V-MOD “heute” or the V-MOD “nach Franfurt”. A close inspection of such ambiguous sentences in TüBa-D revealed that in a majority of all cases, the MOD-MOD label refers to the first V-MOD in the clause, or the first MOD if there is no V-MOD present. Exceptions to this rule are MOD-MODs in resumptive constructions, which generally refer to the modifier in the VF. Ambiguous OA-MODs generally refer to the closest OA in the clause. By applying these heuristics, the ambiguities are resolved in the dependency structure, as shown in Figure 8 for the syntactic tree in Figure 7.

5. Dependency-Based Parser Evaluation

Lin (1998) proposed a procedure for converting syntactic trees from the gold standard and from the parser into dependency structures. From these structures, precision and recall are calculated.

Another similar evaluation procedure was suggested by Srinivas et al. (1996), they first convert hierarchical phrasal constituents into chunks, and then compute the dependencies between these chunks. This is a valid approach for the Penn treebank annotation style, which assumes a complete flat annotation of complex noun phrases such as noun compounds. Parsers based on manually developed rules tend to assign more internal structure to such noun phrases, which leads to decreased precision. Reducing such phrases to flat chunks alleviates this problem of comparing these different structures. The TüBa-D annotations, however, assign more complex, non-trivial structures to complex noun phrases. Using the method of Srinivas et al. (1996) would therefore lead to a significant loss in information. Additionally, the flattening of phrases into chunks might introduce errors in the data in such cases, in which the conversion into chunks is not obvious, such as for the noun phrase “wichtige Konferenzen und Besprechungen” in the sentence “da haben wir noch wichtige Konferenzen und Besprechungen” (we still have important conferences and business meetings) shown in Figure 9.

Basili et al. (1998) developed a similar approach for the Italian language. But instead of parsing a sentence completely and then reducing this parse to chunks and dependencies between chunks, Basili et al. apply a chunk parser combined with a module that calculates dependencies between these chunks. For this approach, the same restrictions hold as for the evaluation procedure of Srinivas et al. (1996).

The evaluation method presented here is based on Lin’s (Lin, 1998) approach. Following Lin’s procedure, we first convert both the gold standard tree and the parser’s output into dependency structures and compare these by applying (labeled) precision and (labeled) recall to these dependency structures.

TüSBL’s analyses depend heavily on the syntactically annotated sentences contained in the instance base. It is therefore difficult to give examples of errors for specific sentences or linguistic phenomena. It is, however, possible to characterize the typical behavior of the parser and give typical examples of errors.

Attachment errors. Attachment errors as described in Section 1. are not very common for TüSBL. Since TüSBL uses the complete sentence as context to retrieve the most similar tree, it either finds the correct spanning analysis or it does not attach all constituents. In the few cases

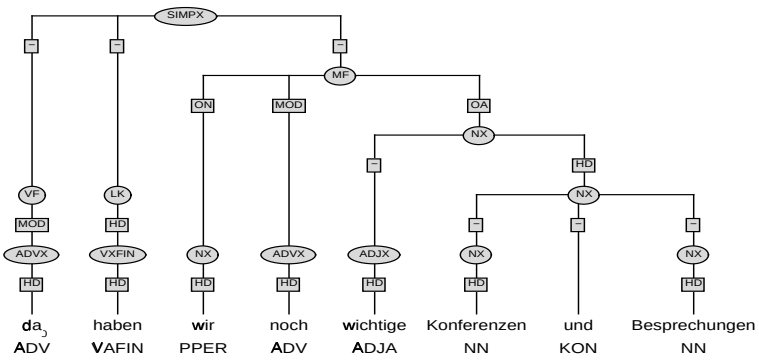


Figure 9: A complex noun phrase in the TüBa-D annotation scheme.

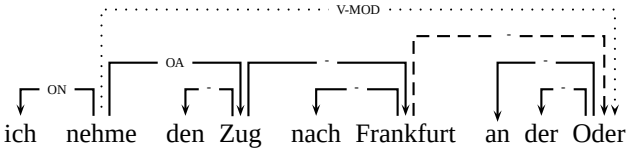


Figure 10: The dependency structure of the trees in Figure 1 and 2. The wrong attachment is shown as a dotted arc whereas the correct attachment is shown as a dashed arc.

where attachment errors are introduced by incorrect adaptations of the retrieved trees or in cases when a wrong tree is found as the most similar one, the parsers evaluation based on constituents suffers from the same problems as decribed in Section 2. above. The parser's output containing the wrong attachment in Figure 1 would result in $10/13 = 76.92\%$ recall and $10/12 = 83.33\%$ precision when using a constituent-based evaluation scheme. The dependency structure of the wrong and the correct attachment is shown in Figure 10. With the dependency-based evaluation, both precision and recall would be calculated as $7/8 = 87.50\%$.

Coordination. Coordination phenomena are in general very difficult to treat with deterministic partial parsers since this type of parsers needs to make the decision on the scope of a coordination early on when there is not enough information available. Two examples of coordination can be found in Figure 11. For both cases, TüSBL would typically retrieve these trees but not be able to attach the conjunction and the second conjunct, as shown in Figure 12 for the second example. For the first example, "am siebten und achten" (on the seventh and the eighth), this would lead to $2/4 = 50.00\%$ recall and $2/3 = 66.67\%$ precision. For the second example, "das wäre Mittwoch der dritte und Donnerstag der vierte August" (that would be Wednesday the third and Thursday the fourth of August), recall would be $9/12 = 75.00\%$ and precision $9/11 = 81.82\%$. If the evaluation is based on dependencies, TüSBL's analysis would deviate from the gold standard by the missing dependencies of the conjunction and the second conjunct. Therefore, recall would be $1/3 = 33.33\%$, for the first example, and $7/9 = 77.78\%$ for the second example. Precision would be $1/1 = 100\%$ for the first example and $7/7 = 100\%$ for the second example.

Another problematic coordination phenomenon constitute split-up coordinations such as in the sentence "das Hotel hat sogar ein Schwimmbad und ein Solarium dabei und einen Fitnessraum" (the hotel even has a swimming pool and a tanning booth – and a fitness room) in Figure 5. A typical error that might occur when parsing such sentences with TüSBL is that the split-up conjunct "und einen Fitnessraum" would not be attached. This would result in $12/14 = 85.71\%$ recall and $12/13 = 92.31\%$ precision. The evaluation based on the dependency structure shown in Figure 6 leads to $11/12 = 91.67\%$ recall and $11/11 = 100\%$ precision.

The comparison shows that dependency-based recall tends to suffer less than constituent-based recall since the unattached part of the coordination does not contribute to errors on higher levels, such as the MF and SIMPX in the second example, which are in principle correct. Dependency-based precision, on the other hand, does not depend on the level of embedding of the coordinations but only on the number of conjuncts that were correctly attached.

Unattached phrases. The failure to attach constituents at the end of an input sentence is the most common error type when evaluating partial parsers. It is generally part of the design decisions to prefer partial analyses which can be gained with a small amount of effort but which will be correct in a majority of cases to complete analyses which involve a high degree of manual labor and a higher error rate for attachment decisions. A typical analysis of TüSBL for the input sentence "wir müssen ja noch einen Bericht abfassen über diese Reise nach Hannover" would be similar to the tree in Figure 3; one possible error might be that the last PX ("nach Hannover") could not be attached to the NX ("diese Reise"). Thus, the NX node 513 would be missing, and the PX node 514 would then immediately dominate the NX node 506. Using the PARSEVAL measures, this

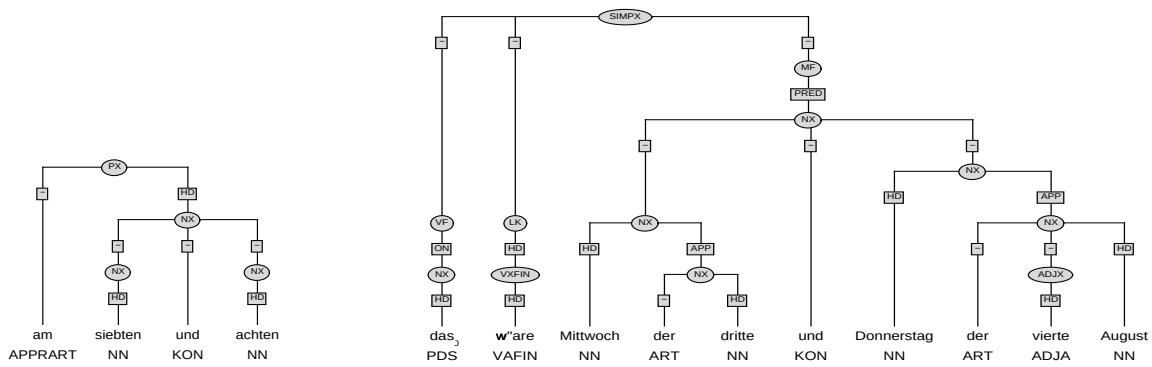


Figure 11: Two trees containing coordination.

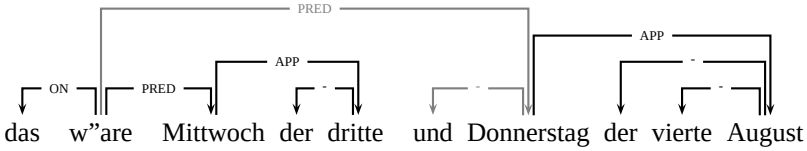


Figure 12: The dependency-based representation of the second example in Figure 11. TüSBL's analysis is shown in black, the missing dependencies in gray.

error would result in $13/17 = 76.47\%$ labeled recall and $13/16 = 81.25\%$ labeled precision. The evaluation based on the dependency structure would give $10/11 = 90.90\%$ labeled recall and $10/10 = 100\%$ labeled precision. Considering that only the attachment of the final PX is missing and that the analysis of the sentence is otherwise correct and complete, the latter figures give a better picture of the quality of the partial parse.

6. Conclusion

We have shown that the PARSEVAL measures do not allow a suitable evaluation of partial parsers. If the evaluation is based on constituency, missing information in the partial parses leads to precision and recall errors in several constituents, and the losses in both measures are disproportionately high. We therefore proposed a dependency-based evaluation. TüBa-D, the treebank used here, contains all the necessary dependency information so that the conversion of trees into a dependency structure does not have to rely on heuristics. Therefore, the dependency representations are not only reliable, they are also linguistically motivated and can be used for linguistic purposes. Using these structures for evaluation ensures that missing information will not decrease the evaluation measures disproportionately, which allows a more suitable evaluation of partial information.

7. Acknowledgments

The research reported here was supported by the German Research Council (DFG) as part of the Sonderforschungsbereich 441 "Linguistische Datenstrukturen" (Linguistic Data Structures). The authors are grateful to Steven Abney, who made the chunk parser available, and to Prof. Hans Uszkoreit and his colleagues at the Universität des Saarlandes, who kindly provided us with the graphical annotation tool *Annotate*. The authors are also grateful to the anonymous reviewers for their helpful comments.

8. References

Steven Abney. 1991. Parsing by chunks. In Robert Berwick, Steven Abney, and Caroll Tenney, editors, *Principle-Based Parsing*. Kluwer Academic Publishers.

Steven Abney. 1996. Partial parsing via finite-state cascades. In John Carroll, editor, *Workshop on Robust Parsing (ESSLLI '96)*, pages 8 – 15, Prague, Czech Republic.

Salah Aït-Mokhtar and Jean-Pierre Chanod. 1997. Incremental finite-state parsing. In *Proceedings of ANLP'97*, pages 72 – 79, Washington, D.C.

Roberto Basili, Maria Teresa Pazienza, and Fabio Massimo Zanzotto. 1998. Evaluating a robust parser for Italian language. In *Proceedings of the International Conference on Language Resources and Evaluation, LREC 1998*, Granada, Spain.

E. Black, S. Abney, D. Flickinger, C. Gdaniec, R. Grishman, P. Harrison, D. Hindle, R. Ingria, F. Jelinek, J. Klavans, M. Liberman, M. Marcus, S. Roukos, B. Santorini, and T. Strzalkowski. 1991. A procedure for quantitatively comparing the syntactic coverage of English grammars. In *Proceedings of the DARPA Speech and Natural Language Workshop 1991*, Pacific Grove, CA.

Thorsten Brants and Wojciech Skut. 1998. Automation of treebank annotation. In *Proceedings of NeMLaP-3/CoNLL98*, pages 49 – 57, Sydney, Australia.

John Carroll and Ted Brisoce. 1996. Apportioning development effort in a probabilistic LR parsing system through evaluation. In *Proceedings of the ACL/SIGDAT Conference on Empirical Methods in Natural Language Processing*, pages 92 – 100, University of Pennsylvania, Philadelphia, PA.

Erich Drach. 1937. *Grundgedanken der Deutschen Satzlehre*. Frankfurt/M.

Oskar Erdmann. 1886. *Grundzüge der deutschen Syntax nach ihrer geschichtlichen Entwicklung dargestellt.* Stuttgart. Erste Abteilung.

Hans Jürgen Heringer. 1996. *Deutsche Syntax Dependentiell*. Stauffenburg-Verlag, Tübingen.

Simon Heinrich Adolf Herling. 1821. Über die Topik der deutschen Sprache. In *Abhandlungen des frankfurterischen Gelehrtenvereins für deutsche Sprache*, pages 296–362, 394. Frankfurt/M. Drittes Stück.

Erhard W. Hinrichs, Julia Bartels, Yasuhiro Kawata, Valia Kordoni, and Heike Telljohann. 2000a. The Tübingen treebanks for spoken German, English, and Japanese. In Wolfgang Wahlster, editor, *Verbmobil: Foundations of Speech-to-Speech Translation*. Springer, Berlin.

Erhard W. Hinrichs, Julia Bartels, Yasuhiro Kawata, Valia Kordoni, and Heike Telljohann. 2000b. The Verbmobil treebanks. In *5. Konferenz zur Verarbeitung natürlicher Sprache (KONVENS 2000)*, pages 107 – 112, Ilmenau, Germany.

Tilman Höhle. 1985. Der Begriff "Mittelfeld, Anmerkungen über die Theorie der topologischen Felder. In *Akten des Siebten Internationalen Germanistenkongresses*, pages 329–340, Göttingen.

Sandra Kübler and Erhard W. Hinrichs. 2001a. From chunks to function-argument structure: A similarity-based approach. In *Proceedings of ACL-EACL 2001*, pages 338 – 345, Toulouse, France.

Sandra Kübler and Erhard W. Hinrichs. 2001b. TüSBL: A similarity-based chunk parser for robust syntactic processing. In *Proceedings of the First International Human Language Technology Conference, HLT-2001*, San Diego, CA, March.

Dekang Lin. 1995. A dependency-based method for evaluating broad-coverage parsers. In *Proceedings of IJCAI-95*, pages 1420 – 1425, Montreal, Canada.

Dekang Lin. 1998. A depedency-based method for evaluating broad-coverage parsers. *Natural Language Engineering*, 4(2):97 – 114.

Christopher D. Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. MIT Press, Cambridge, MA.

Oliver Plaehn, 1998. *Annotate Bedienungsanleitung*. Universität des Saarlandes, Sonderforschungsbereich 378, Projekt C3, Saarbrücken, Germany, April.

Marga Reis. 1980. On justifying topological frames: 'Positional field' and the order of nonverbal constituents in German. *DRLAV: Revue de Linguistique*, 22/23:59 – 85.

Anne Schiller, Simone Teufel, and Christine Thielen. 1995. Guidelines für das Tagging deutscher Textkorpora mit STTS. Technical report, Universität Stuttgart and Universität Tübingen, September.

B. Srinivas, Christine Doran, Beth Ann Hockey, and Aravind Joshi. 1996. An approach to robust partial parsing and evaluation metrics. In John Carrol, editor, *Eight European Summer School in Language, Logic, and Information*, pages 70 – 82, Prague, Czech Republic.

Rosmary Stegmann, Heike Telljohann, and Erhard W. Hinrichs. 2000. Stylebook for the German Treebank in VERBMOBIL. Technical Report 239, Verbmobil.

LinGO Redwoods

A Rich and Dynamic Treebank for HPSG

**Stephan Oepen, Ezra Callahan, Dan Flickinger,
Christoper D. Manning, Kristina Toutanova**

Center for the Study of Language and Information
Stanford University
Ventura Hall, Stanford, CA 94305 (USA)

{oe | ezra99 | dan | manning | kristina}@csli.stanford.edu

Abstract

The LinGO Redwoods initiative is a seed activity in the design and development of a new type of treebank. A treebank is a (typically hand-built) collection of natural language utterances and associated linguistic analyses; typical treebanks—as for example the widely recognized Penn Treebank (Marcus, Santorini, & Marcinkiewicz, 1993), the Prague Dependency Treebank (Hajic, 1998), or the German TiGer Corpus (Skut, Krenn, Brants, & Uszkoreit, 1997)—assign syntactic phrase structure or tectogrammatical dependency trees over sentences taken from a naturally-occuring source, often newspaper text. Applications of existing treebanks fall into two broad categories: (i) use of an annotated corpus in empirical linguistics as a source of structured language data and distributional patterns and (ii) use of the treebank for the acquisition (e.g. using stochastic or machine learning approaches) and evaluation of parsing systems.

While several medium- to large-scale treebanks exist for English (and some for other major languages), all pre-existing publicly available resources exhibit the following limitations: (i) the depth of linguistic information recorded in these treebanks is comparatively shallow, (ii) the design and format of linguistic representation in the treebank hard-wires a small, predefined range of ways in which information can be extracted from the treebank, and (iii) representations in existing treebanks are static and over the (often year- or decade-long) evolution of a large-scale treebank tend to fall behind theoretical advances in formal linguistics and grammatical representation.

LinGO Redwoods aims at the development of a novel treebanking methodology, (i) *rich* in nature and *dynamic* in both (ii) the ways linguistic data can be retrieved from the treebank in varying granularity and (iii) the constant evolution and regular updating of the treebank itself, synchronized to the development of ideas in syntactic theory. Starting in October 2001, the project is aiming to build the foundations for this new type of treebank, develop a basic set of tools required for treebank construction and maintenance, and construct an initial set of 10,000 annotated trees to be distributed together with the tools under an open-source license. Building a large-scale treebank, disseminating it, and positioning the corpus as a widely-accepted resource is a multi-year effort; the results of this seeding activity will serve as a proof of concept for the novel approach that is expected to enable the LinGO group at CSLI both to disseminate the approach to the wider academic and industrial audience and to secure appropriate funding for the realization and exploitation of a larger treebank. The purpose of publication at this early stage is three-fold: (i) to encourage feedback on the Redwoods approach from a broader academic audience, (ii) to facilitate exchange with related work at other sites, and (iii) to invite additional collaborators to contribute to the construction of the Redwoods treebank or start its exploitation as early-access versions become available.

1. Why Another (Type of) Treebank?

For the past decade or more, symbolic, linguistically oriented methods (like those pursued within the HPSG framework; see below) and statistical or machine learning approaches to NLP have typically been perceived as incompatible or even competing paradigms; the former, more traditional approaches are often referred to as 'deep' NLP, in contrast to the comparatively recent branch of language technology focussing on 'shallow' (text) processing methods. Shallow processing techniques have produced useful results in many classes of applications, but have not met the full range of needs for NLP, particularly where precise interpretation is important, or where the variety of linguistic expression is large relative to the amount of training data available. On the other hand, deep approaches to NLP have only recently achieved broad enough grammatical coverage *and* sufficient processing efficiency to allow the use of HPSG-type systems in certain types of real-world applications. Fully-automated, deep grammatical analysis of unrestricted text remains an unresolved challenge.

In particular, applications of analytical grammars for natural language parsing or generation require the use of sophisticated statistical techniques for resolving ambiguities. We observe general consensus on the necessity for bridging activities, combining symbolic and stochastic approaches to NLP; also, the transfer of HPSG resources into industry has amplified the need for general parse ranking, disambiguation, and robust recovery techniques which all require suitable stochastic models for HPSG processing. While we find promising research in stochastic parsing in an number of frameworks, there is a lack of appropriately rich and dynamic language corpora for HPSG. Likewise, stochastic parsing has so far been focussed on IE-type applications and lacks any depth of semantic interpretation. The Redwoods initiative is designed to fill in this gap.

Most probabilistic parsing research—including, for example, work by by Collins (1997), Charniak (1997), and Manning and Carpenter (2000)—is based on branching process models (Harris, 1963). An important recent advance in this area has been the application of log-linear models (Agresti, 1990) to modeling linguistic systems. These models can deal with the many interacting dependencies and the structural complexity found in constraint-based or unification-based theories of syntax (Johnson, Geman, Canon, Chi, & Riezler, 1999). The availability of even a medium-size treebank would allow us to begin exploring

the use of these models for probabilistic disambiguation of HPSG grammars. At the same time, other researchers have started work on stochastic HPSG (or are about to), some pursuing unsupervised approaches, but in many cases using the same grammar or at least the same descriptive formalism and grammar engineering environment. The availability of a reasonably large, hand-disambiguated HPSG treebank is expected to greatly facilitate comparability of results and models obtained by various groups and, eventually, to help define a common evaluation metric.

2. Background

The LinGO Project at CSLI has been conducting research and development in Head-Driven Phrase Structure Grammar (HPSG; Pollard & Sag, 1994) since 1994. In close collaboration with international partners—primarily from Saarbrücken (Germany), Cambridge, Edinburgh, and Sussex (UK), and Tokyo (Japan)—the LinGO Project has developed a broad-coverage, precise HPSG implementation of English (the LinGO English Resource Grammar, ERG; Flickinger, 2000), a framework for semantic composition in large-scale computational grammars (Minimal Recursion Semantics, MRS; Copestake, Lascarides, & Flickinger, 2001), and an advanced grammar development environment (the LKB system; Copestake, 1992, 1999). Through contributions from collaborating partners, a pool of open-source HPSG resources has developed that now includes broad-coverage grammars for several languages, a common profiling and benchmarking environment (Oepen & Callmeier, 2000), and an industrial-strength C^{++} run-time engine for HPSG grammars (Callmeier, 2000). LinGO resources are in use world-wide for teaching, research, and application building. Because of the wide distribution and common acceptance, the HPSG framework and LinGO resources present an excellent anchor point for the Stanford treebanking initiative.

3. A Rich and Dynamic Treebank

The key innovative aspect of the Redwoods approach to treebanking is the anchoring of all linguistic data captured in the treebank to the HPSG framework and a generally-available broad-coverage grammar of English, viz. the LinGO English Resource Grammar, combined with tools for the extraction of various, user-defined representations and a software environment to continuously update the treebank as part of the on-going grammar maintenance and extension. Unlike existing treebanks, there will be no need to define a (new) form of grammatical representation specific to the treebank. Instead, the treebank will record complete syntacto-semantic analyses as defined by the LinGO ERG and provide tools to extract many different types of linguistic information at greatly varying granularity.

In particular, the project centrally draws on the [incr tsdb()] profiling environment (essentially a specialized database recording fine-grained parsing results obtained from diverse HPSG systems; Oepen & Carroll, 2000), constructing the treebank as an extension of the existing data model and tools. In turn building on a pre-existing tree comparison tool in the LKB (similar in kind to the SRI Cambridge TreeBanker; Carter, 1997), the treebanking environment presents annotators, one sentence at a time, with the full set of analyses produced by the grammar. Using the tree comparison tool, annotators can quickly navigate through the parse forest and identify the correct or preferred analysis in the current context (or, in rare cases, reject all analyses proposed by the grammar). The tree selection tools persents users, who need little expert knowledge of the underlying grammar, with a range of properties that distinguish competing analyses and that are relatively easy to judge. Each such property corresponds to the usage of a particular lexical item, semantic relation, or grammar rule applied to a specific substring to form a constituent; unlike the LFG packed f-structure representations discussed by King, Dipper, Frank, Kuhn, and Maxwell (2000), the set of basic discriminating properties reduces the information presented to annotators to the minimal amount of structure required to completely disambiguate a sentence. All disambiguating decisions made by annotators are recorded in the [incr tsdb()] database and thus become available for (i) later dynamic extraction from the annotated profile or (ii) dynamic propagation into a more recent profile obtained from re-running an extended version of the grammar on the same corpus.

Important innovative research aspects pertaining to this approach to treebanking are (i) enabling users of the treebank to extract information of the type they need and to transform the available representation into a form suited for their needs and (ii) updating the treebank for an enhanced version of the grammar underlying the recorded analyses in an automated fashion, viz. by re-applying the disambiguating decisions to an updated version of the corpus.

Depth of Representation and Transformation of Information Internally, the [incr tsdb()] database records analyses in three different formats, viz. (i) as a derivation tree composed of identifiers of lexical items and constructions used to construct the analysis, (ii) as a traditional phrase structure tree labeled with an inventory of some fifty atomic labels (of the type 'S', 'NP', 'VP' et al.), and (iii) as an underspecified MRS meaning representation. While (ii) will in many cases be similar to the representation found in the Penn Treebank, (iii) subsumses the functor – argument (or tectogrammatical) structure as it is advocated in the Prague Dependency Treebank or the German TiGer corpus. Most importantly, however, representation (i) provides all the information required to replay the full HPSG analysis (e.g. using the original HPSG grammar and one of the open-source HPSG processing environments, e.g. the LKB or PET, which already have been interfaced to [incr tsdb()]). Using the latter approach, users of the treebank are enabled to extract information in whatever representation they require, simply by reconstructing the full analysis and adapting the existing mappings (e.g. the inventory of node labels used for phrase structure trees) to their needs. Figure 1 depicts the internal Redwoods encoding and two export representations derived from existing conversion routines. Labeled phrase structure trees result from reconstructing a derivation (using the original grammar) and matching a user-defined set of underspecified feature structure 'templates'

Table 1: Redwoods development status as of February 2002: four sets of transcribed and hand-segmented VerbMobil dialogues have been annotated. The columns are, from left to right, the total number of sentences (excluding fragments) for which the LinGO grammar has at least one analysis ('♯'), average length ('‖'), lexical and structural ambiguity ('⊙' and '×', respectively), followed by the last four metrics broken down for the following subsets: sentences (i) for which the annotator rejected all analyses (no active trees), (ii) where annotation resulted in exactly one preferred analysis (one active tree), (iii) those where full disambiguation was not accomplished through the first round of annotation (more than one active tree), and (iv) massively ambiguous sentences that have yet to be annotated.

corpus	total				active = 0				active = 1				active > 1				unannotated			
	♯	‖	⊙	×	♯	‖	⊙	×	♯	‖	⊙	×	♯	‖	⊙	×	♯	‖	⊙	×
VM6	2422	7·7	4·2	32·9	218	8·0	4·4	9·7	1910	7·0	4·0	7·5	80	10·0	4·8	23·8	214	14·9	4·3	287·5
VM13	1984	8·5	4·0	37·9	175	8·5	4·1	9·9	1491	7·2	3·9	7·5	85	9·9	4·5	22·1	233	14·1	4·2	212·0
VM31	1726	6·2	4·5	22·4	164	7·9	4·6	8·0	1360	6·6	4·5	5·9	61	10·1	4·2	14·5	141	13·5	4·7	201·5
VM32	608	7·4	4·3	25·6	46	9·8	4·1	18·3	516	7·5	4·4	9·2	21	10·4	3·9	29·6	25	16·6	4·8	375·4

against the HPSG feature structure at each node in the tree. The elementary dependency graph, on the other hand, is an abstraction from the full MRS meaning representation associated to each full analysis; informally, elementary dependencies correspond to the type of tectogrammatical representations found in the Prague Dependency Treebank or the German TiGer corpus and, likewise, resemble the basic relations suggested for parser evaluation by Carroll, Briscoe, and Sanfilippo (1998). Given a rich body of MRS manipuation and conversion software, it is relatively straightforward to adapt the type and form of elementary dependencies to user needs.

For evaluation purposes, the existing [incr tsdb()] facilities for comparing across competence and performance profiles can be deployed to gauge results of a (stochastic) parse disambiguation system, essentially using the preferences recorded in the treebank as a 'gold standard' target for comparison. While the concept of a meta-treebank of the type proposed here has been explored in earlier research (e.g. the AMALGAM project at Leeds University in the UK; Atwell, 1996), previous approaches to the dynamic mapping of treebank representations have built on a static, finite set of hand-constructed mappings.

Automating Treebank Construction Although a precise HPSG grammar like the LinGO ERG will typically assign a small number of analyses to a given sentence, choosing among a handful or sometimes a few dozens of readings is time-consuming and error-prone. The project will explore two approaches to automating the disambigutation task, viz. (i) seeding lexical selection from a part-of-speech (POS) tagger and (ii) automated inter-annotator comparison and assisted resolution of conflicts. Ranking lexical ambiguity on the basis of tagger-assigned POS probabilities requires research into generalizations over the rather fine-grained hierarchy of HPSG lexical types and identifying many-to-many correspondences in a standard POS tagset. Conversely, detecting mismatches (i.e. conflicts) between disambiguating decisions made for the same input sentence by two independent annotators will facilitate research into the linguistic nature of the discriminating properties used and existing logical relations (inclusion, implication, inconsistency et al.) among subsets of discriminators. To exemplify the nature of these properties, consider the sentence

(1) *Have her report on my desk by Friday!*

which is (correctly) assigned thirty two readings by the HPSG grammar; while human language users (and correspondingly human annotators) will typically not note most of the alternative analyses, one can contextualize the sentence to emphasize either one of the following ambiguities: the causative vs. possessive *have*, the determiner vs. personal pronoun *her*, the noun vs. verb *report*, the temporal vs. locative preposition *by*, and *Friday* as a day of the week vs. as a proper noun (e.g. the name of a bar). Using the tree comparison tool and our notion of elementary discriminators, annotators can reduce the set of analyses quickly (where full disambiguation requires minimally four decisions for this example); yet, a POS tagger will reliably assign high probability to the pairings ⟨*her* , determiner⟩ and ⟨*report* , noun⟩ which could be used to bias the presentation to annotators.

Treebank Maintenance and Evolution Perhaps the most challenging research aspect of the Redwoods initiative is about developing a methodology for automated updates of the treebank to reflect the continuous evolution of the underlying linguistic framework and of the LinGO grammar. Again building on the notion of elementary linguistic discriminators, it is expected to explore the semi-automatic propagation of recorded disambiguating decisions into newer versions of the parsed corpus. While it can be assumed that the basic phrase structure inventory and granularity of lexical distinctions have stabilized to a certain degree, it is not guaranteed that one set of discriminators will always fully disambiguate a more recent set of analyses for the same utterance (as the grammar may introduce additional distinctions), nor that re-playing a history of disambiguating decisions will necessarily identify the correct, preferred analysis for all sentences. Once more, a better understanding into the nature of discriminators and relations holding among them is expected to provide the foundations for an update procedure that, ultimately, should be fully automated or at least require minimal manual inspection.

Scope and Current State of Seeding Initiative The first 10,000 trees to be hand-annotated as part of the kick-off initiative are taken from a domain for which the English Resource Grammar is known to exhibit broad and accurate

coverage, viz. transcribed face-to-face dialogues in an appointment scheduling and travel arrangement domain. Corpora of some 50,000 such utterances are readily available from the VerbMobil project (Wahlster, 2000) and have already been studied extensively among researchers worldwide in the field. For the follow-up phase of the project, it is expected to move into a second domain and text genre, presumably more formal, edited text taken from newspaper text or another widely available on-line source. As of April 2002, the seeding initiative is well underway. The integrated treebanking environment, combining [incr tsdb()] and the LKB tree selection tool, has been established and has been deployed in a first iteration of annotating a corpus of 10,000 VerbMobil utterances. For a second-year Stanford undergraduate in linguistics, the approach to parse selection through minimal discriminators turned out to be not at all hard to learn and required less training in specifics of the grammatical analyses delivered by the LinGO grammar than could have been expected.

Table 1 summarizes the current Redwoods development status; while annotation of a residual fraction of highly ambiguous sentences and inter-annotator cross-validation continue, the current development snapshot of the treebank can be made available upon request. We have just started work on stochastic parse selection models for the Redwoods treebank, so far obtaining a parse selection accuracy of around eighty per cent from a combination of existing methods applied to the Redwoods derivation trees and elementary dependency graphs (see Figure 1); details on Redwoods parse selection results will be reported in separate publications.

4. Related Work

To our best knowledge, no prior research has been conducted exploring both the linguistic depth, flexibility in available information, and dynamic nature of treebanks as proposed presently. Earlier work on building corpora of hand-selected analyses relative to an existing broad-coverage grammar was carried out at Xerox PARC, SRI Cambridge, and Microsoft Research; as all these resources are tuned to proprietary grammars and analysis engines, the resulting treebanks are not publicly available, nor have research results reported been reproducible. Yet, especially in the light of the successful LinGO open-source repository, it seems vital that both the treebank and associated processing schemes and stochastic models be made available to the general (academic) public.

An on-going initiative at Rijksuniversiteit Groningen (NL) is developing a treebank of dependency structures (Mullen, Malouf, & Noord, 2001), as they are derived from an HPSG-like grammar of Dutch (Bouma, Noord, & Malouf, 2001). While the general approach resembles the Redwoods initiative (specifically the discriminator-based method used in selecting trees from the set of analyses proposed by the grammar; the LKB tree selection tool was originally developed by Malouf, after all), there are three important differences. Firstly, the Groningen decision to compose the treebank from dependency structures commits the resulting resource to a single stratum of representation, tectogrammatical structure essentially, and thus eliminates some of the flexibility in extracting various types of linguistic structure that the Stanford initiative foresees. Secondly, and in a similar vein, recording dependency structures means that the (stochastic) disambiguation component has to consider two syntactically different analyses equivalent whenever they project identical dependency structures; hence, there is a mismatch of granularity between the disambiguated treebank structures and the primary structures (i.e. derivation trees) constructed by the grammar. Finally, the Groningen initiative is making the assumption that the dependency structures, once they are stored in the treebank, are correct and do not change over time (or as an effect of grammar evolution); from the available publications, at least, there is no evidence that the disambiguating decisions made by annotators are recorded in the treebank or that the project expects to dynamically update the treebank with future revisions of the underlying grammar.

Another closely related approach is the work reported by Dipper (2000), essentially the application of a broad-coverage LFG grammar for German to constructing tectogrammatical structures for the TiGer corpus. While many of the basic assumptions about the value of a systematic, broad-coverage grammar for the treebank construction are shared, the strategy followed by Dipper (2000) exhibits the same limitations as the Groningen initiative: the TiGer target representation, still, is mono-stratal and the approach to hand-disambiguation and subsequent transfer of result structures into the TiGer corpus looses the linkage to the original analyses and basic properties used in the disambiugation, hence the potential for dynamic adaptation of the data or automatic updates.

Acknowledgements

The Redwoods initiative is part of the LinGO Laboratory at CSLI and many people, both at Stanford and at partner sites, have contributed to its design and (given small amounts of resources) relative success so far. Ivan A. Sag, Tom Wasow, Emily M. Bender, Tim Baldwin, John Beavers, and Kathryn Campbell-Kibler all have participated in our regular tree conferences, helping annotators select parses and offering productive critiques on analyses provided by the LinGO grammar. Ann Copestake, John Carroll, Rob Malouf, and Stephan Oepen are the main developers of the LKB and [incr tsdb()] software packages from which the Redwoods treebanking environment has been built and, in various capacities, have influenced the Redwoods approach significantly. During a three-month visit to Stanford, Stuart Shieber has been among the driving forces for applications of the existing development version of the treebank, helping us develop and fine-tune suitable stochastic parse selection models. The Redwoods initiative has been partially funded by an internal opportunity grant from CSLI Stanford and by a donation from YY Technologies.

References

Agresti, A. (1990). *Categorical data analysis.* John Wiley & Sons.

Atwell, E. (1996). Comparative evaluation of grammatical annotation models. In R. Sutcliffe, H.-D. Koch, & A. McElligott (Eds.), *Proceedings of the Workshop on Industrial Parsing of Software Manuals* (pp. 25 – 46). Amsterdam, The Netherlands: Rodopi.

Bouma, G., Noord, G. van, & Malouf, R. (2001). Alpino. Wide-coverage computational analysis of Dutch. In W. Daelemans, K. Sima-an, J. Veenstra, & J. Zavrel (Eds.), *Computational linguistics in the netherlands* (pp. 45 – 59). Amsterdam, The Netherlands: Rodopi.

Callmeier, U. (2000). PET — A platform for experimentation with efficient HPSG processing techniques. *Natural Language Engineering, 6 (1) (Special Issue on Efficient Processing with HPSG)*, 99 – 108.

Carroll, J., Briscoe, E., & Sanfilippo, A. (1998). Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st International Conference on Language Resources and Evaluation* (pp. 447 – 454). Granada, Spain.

Carter, D. (1997). The TreeBanker. A tool for supervised training of parsed corpora. In *Proceedings of the Workshop on Computational Environments for Grammar Development and Linguistic Engineering*. Madrid, Spain.

Charniak, E. (1997). Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the Fourteenth National Conference on Artificial Intelligence* (pp. 598 – 603). Providence, RI.

Collins, M. J. (1997). Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Meeting of the Association for Computational Linguistics and the 7th Conference of the European Chapter of the ACL* (pp. 16 – 23). Madrid, Spain.

Copestake, A. (1992). The ACQUILEX LKB. Representation issues in semi-automatic acquisition of large lexicons. In *Proceedings of the 3rd ACL Conference on Applied Natural Language Processing* (pp. 88 – 96). Trento, Italy.

Copestake, A. (1999). *The (new) LKB system.* (CSLI, Stanford University: `http://www-csli.stanford.edu/~aac/doc5-2.pdf`)

Copestake, A., Lascarides, A., & Flickinger, D. (2001). An algebra for semantic construction in constraint-based grammars. In *Proceedings of the 39th Meeting of the Association for Computational Linguistics.* Toulouse, France.

Dipper, S. (2000). Grammar-based corpus annotation. In *Workshop on linguistically interpreted corpora linc-2000* (pp. 56 – 64). Luxembourg.

Flickinger, D. (2000). On building a more efficient grammar by exploiting types. *Natural Language Engineering, 6 (1) (Special Issue on Efficient Processing with HPSG)*, 15 – 28.

Hajic, J. (1998). Building a syntactically annotated corpus. the Prague dependency treebank. In *Issues of valency and meaning* (pp. 106 – 132). Prague, Czech Republic: Karolinum.

Harris, T. E. (1963). *The theory of branching processes.* Berlin, Germany: Springer.

Johnson, M., Geman, S., Canon, S., Chi, Z., & Riezler, S. (1999). Estimators for stochastic 'unification-based' grammars. In *Proceedings of the 37th Meeting of the Association for Computational Linguistics* (pp. 535 – 541). College Park, MD.

King, T. H., Dipper, S., Frank, A., Kuhn, J., & Maxwell, J. (2000). Ambiguity management in grammar writing. In *Workshop on linguistic theory and grammar implementation* (pp. 5 – 19). Birmingham, UK.

Manning, C. D., & Carpenter, B. (2000). Probabilistic parsing using left corner language models. In H. Bunt & A. Nijholt (Eds.), *Advances in probabilistic and other parsing technologies* (pp. 105 – 124). Kluwer Academic Publishers.

Marcus, M. P., Santorini, B., & Marcinkiewicz, M. A. (1993). Building a large annotated corpus of English. The Penn Treebank. *Computational Linguistics, 19*, 313 – 330.

Mullen, T., Malouf, R., & Noord, G. van. (2001). Statistical parsing of Dutch using Maximum Entropy models with feature merging. In *Proceedings of the Natural Language Processing Pacific Rim Symposium.* Tokyo, Japan.

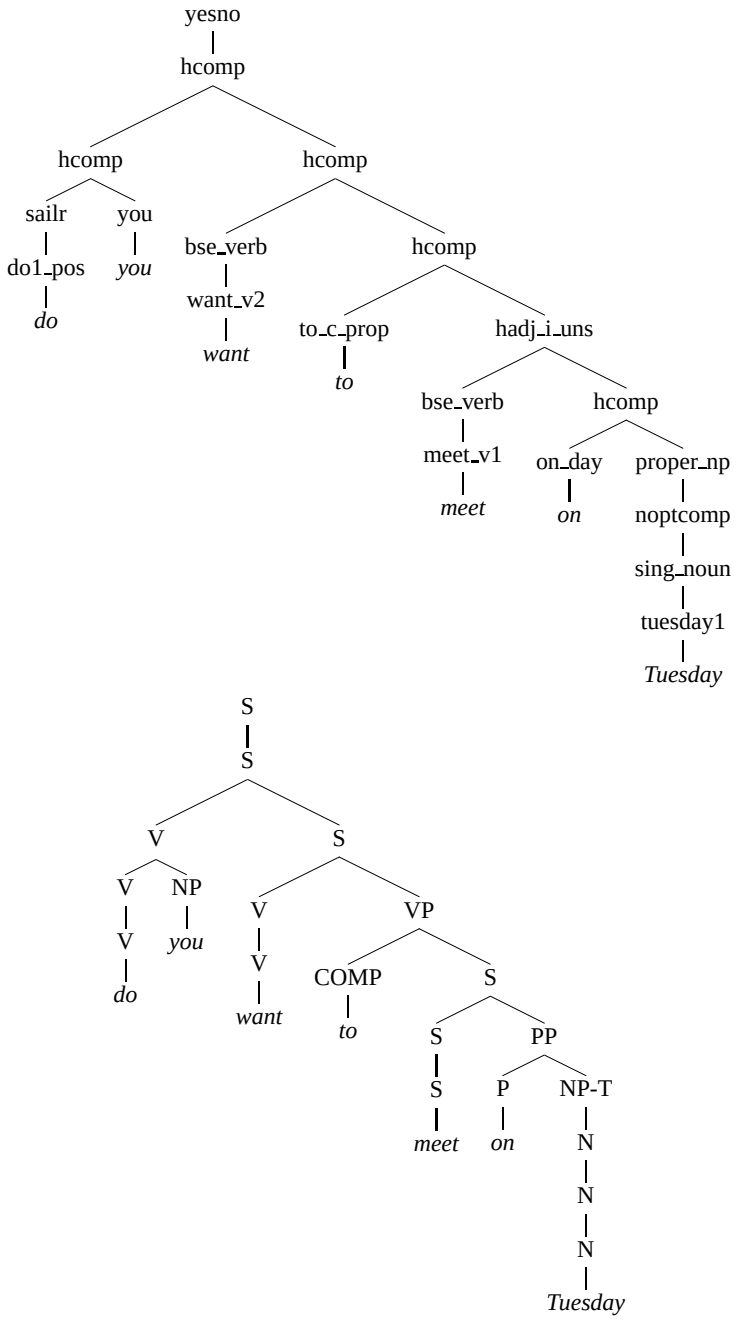
Oepen, S., & Callmeier, U. (2000). Measure for measure: Parser cross-fertilization. Towards increased component comparability and exchange. In *Proceedings of the 6th International Workshop on Parsing Technologies* (pp. 183 – 194). Trento, Italy.

Oepen, S., & Carroll, J. (2000). Performance profiling for parser engineering. *Natural Language Engineering, 6 (1) (Special Issue on Efficient Processing with HPSG)*, 81 – 97.

Pollard, C., & Sag, I. A. (1994). *Head-Driven Phrase Structure Grammar.* Chicago, IL and Stanford, CA: The Univeristy of Chicago Press and CSLI Publications.

Skut, W., Krenn, B., Brants, T., & Uszkoreit, H. (1997). An annotation scheme for free word order languages. In *Proceedings of the 5th ACL Conference on Applied Natural Language Processing.* Washington, DC.

Wahlster, W. (Ed.). (2000). *Verbmobil. Foundations of speech-to-speech translation.* Berlin, Germany: Springer.




```
_4:{
    _4:int_rel[SOA e2:_want2_rel]
   e2:_want2_rel[ARG1 x4:pron_rel, ARG4 _2:hypo_rel]
   _1:def_rel[BV x4:pron_rel]
   _2:hypo_rel[SOA e18:_meet_v_rel]
   e18:_meet_v_rel[ARG1 x4:pron_rel]
   e19:_on_temp_rel[ARG e18:_meet_v_rel, ARG3 x21:dofw_rel]
   x21:dofw_rel[NAMED :tue]
   _3:def_np_rel[BV x21:dofw_rel]
  }
```

Figure 1: Native and derived Redwoods representations for the sentence *Do you want to meet on Tuesday?* — (a) derivation tree using unique rule and lexical item identifiers of the source grammar (top), (b) phrase structure tree labelled with user-defined, parameterizable category abbreviations (center), and (c) elementary dependency graph extracted from MRS meaning representation (bottom).

A Test of the Leaf-Ancestor Metric for Parse Accuracy

Geoffrey Sampson and Anna Babarczy

School of Cognitive and Computing Sciences
University of Sussex
Falmer, Brighton BN1 9QH, England
{geoffs, annab}@cogs.susx.ac.uk

Abstract

The GEIG metric for quantifying accuracy of parsing became influential through the Parseval programme, but many researchers have seen it as unsatisfactory. The LA metric, first developed in the 1980s, arguably comes closer to formalizing our intuitive concept of relative parse accuracy. We support this claim via an experiment which contrasts the performance of alternative metrics on the same body of automatically-parsed examples. The LA metric has the further virtue of providing straightforward indications of the location of parsing errors.

1. Introduction

One of us (Sampson 2000) has argued that what we call the "leaf-ancestor" (LA) metric is better than the Grammar Evaluation Interest Group (GEIG) metric used in the Parseval competition series (e.g. Black et al. 1991) as a way of quantifying the accuracy of automatic parses, in a context where gold-standard parses using a known scheme of node labelling are available. This paper presents an experiment comparing the performance of the two metrics on a sample of automatic-parser output.

The GEIG metric, which counts the numbers of tagmas (multi-word grammatical units) correctly and incorrectly identified, from our point of view lays excessive weight on locating the exact boundaries of constructions.

As originally defined by Black et al. and as it is often applied, the GEIG metric takes no account of node labels at all: it only considers the location of brackets. And in consequence, this metric includes no concept of approximate correctness in identifying tagmas: a pair of brackets either enclose a sequence of words (or other terminal elements) exactly corresponding to a sequence bracketed off in the gold-standard parse, or not. The result is that "it is unclear as to how the score on [the GEIG] metric relates to success in parsing" (Bangalore et al. 1998).

More recently (Magerman 1995, Collins 1997) a refined variant of the GEIG metric has been used which does check label identity as well as wordspan identity in matching tagmas between gold-standard and candidate parses. We shall argue that even this variant of the GEIG metric is inferior to the LA metric. We shall refer to the Black et al. (1991) and Collins (1997) variants of the GEIG metric as GEIG/unlabelled and GEIG/labelled respectively.

We think of "parsing" as determining what kind of larger elements are constituted by the small elements of a string that are open to direct observation. Identifying the exact boundaries of the larger elements is a part, but only one part, of that task. If, for instance, in the gold standard, words 5 to 14 are identified as a noun phrase, then a candidate parse which identifies a noun phrase as beginning at word 5 but ending at word 13, or word 15, should in our view be given substantial though not full credit; under the GEIG metric it is given no credit. The LA metric quantifies accuracy of parsing in this sense.

Incidentally, we believe that the LA metric was the earliest parse-assessment metric in the field, having been used, and briefly described in print, in the 1980s (Sampson, Haigh, & Atwell 1989: 278), though it was later eclipsed by the influential Parseval programme.

2. The Essence of Leaf-Ancestor Assessment

The LA metric evaluates the parsing of an individual terminal element in terms of the similarity of the "lineages" of that element in candidate and gold-standard parse trees, where a lineage is essentially the sequence of node-labels for nodes on the unique path between the terminal element and the root node. The LA value for the parsing of an entire sentence or other many-word unit is simply the average of the values for the individual words. Apart from (we claim) yielding figures for parsing accuracy of complete sentences which succeed better than the GEIG metric in quantifying our intuitions about parse accuracy, the LA metric has the further practical virtue of identifying the location of parsing errors in a straightforward way.

We illustrate the general concept of LA assessment using one of the shortest sentences in our experimental data-set. (The nature of that data-set, and the gold-standard parsing scheme, will be discussed below.) The sentence runs *two tax revision bills were passed.* (Certain typographic details, including capitalization, inverted commas, and sentence-final punctuation marks, have been eliminated from the examples.) The gold-standard analysis, and the candidate analysis produced by an automatic parser, are respectively:

1G [S [N1 *two* [N1 *tax revision*] *bills*] *were passed*]
1C [S [NP *two tax revision bills*] *were passed*]

(Here and below, "nG" and "nC" label gold-standard and candidate analyses for an example *n*.)

The automatic parser has failed to identify *tax revision* as a unit within the tagma headed by *bills*, and it has labelled that tagma NP rather than N1. Lineages for these tree structures are as follows, where for each terminal element the gold-standard lineage is shown to the left and the candidate lineage to the right of the colon, and within each of the paired lineages the Leaf end is to the Left and the Root end to the Right:

two	N1 [S : NP [S
tax	[N1 N1 S : NP S
revision	N1] N1 S : NP S
bills	N1] S : NP] S
were	S : S
passed	S] : S]

The only aspect of the relationship between this notation and the tree structures which is not self-explanatory is the inclusion of boundary markers (left and right square brackets) in many of the lineages. These are included in accordance with the following rules:

- a left-boundary symbol is inserted in the lineage of a terminal element immediately before the label of the highest nonterminal beginning with that element, if there is such a nonterminal

- a right-boundary symbol is inserted in the lineage of a terminal element immediately after the label of the highest nonterminal ending with that element, if there is such a nonterminal

The reason for including these elements in lineages is that, without them, a set of lineages would not always uniquely determine a tree structure; for instance the structures ÒP [Q *a b*] [Q *c*]]Óand ÒP [Q *a b c*]]Ó would not be distinguishable, since the lineage for each terminal element in both cases would consist of the sequence Q P. A set of lineages in which boundary markers have been inserted by these rules uniquely determines the tree structure from which it is derived.

Thus, in the example above, the LA metric equates the accuracy with which the word *two* has been parsed with the degree of similarity between the two strings NP˚[S and N1 [S, it equates the parse-accuracy for *tax* with the degree of similarity between NP S and [˚N1˚N1˚S, and so forth; for the last two words the lineages are identical, so the metric says that they have been parsed perfectly. We postpone discussion of our method for calculating string similarity until after we have discussed our experimental material.

3. The Experimental Material

Our experiment used a set of sentences from genre sections A and G of the SUSANNE Treebank (Sampson 1992) parsed by an automatic treebanker developed at the Universities of Cambridge and Sussex by Ted Briscoe and John Carroll; of those sentences for which the treebanker was able to produce a structure, a set of 500 was randomly chosen. For the purposes of illustrating the performance of the LA metric and comparing it with the GEIG metric, we wanted material parsed by a system that used a simple parsing scheme with a smallish vocabulary of nonterminal labels, and which made plenty of mistakes in applying the scheme to real-life data; there is no suggestion that the parses in our experimental data-set represent the Ò*state of the art*Ófor automatic parsing.

The parsing scheme which the automatic parser was intended to apply used seven nonterminal labels, which we gloss with our own rather than Briscoe and Carroll's labels:

S	finite clause
VP	nonfinite clause
NP	noun phrase containing specifier
N1	noun phrase without specifier
PP	prepositional phrase
AP	adjectival or adverbial phrase
T	Òtextual constituentÓ, defined by Briscoe and Carroll as a tagma enclosing Òa sequence of sub-constituents whose relationship is syntactically indeterminate due to the presence of intervening, delimiting punctuationÓ

The use of these seven categories is defined in greater detail in documentation supplied to us, but for present purposes it is unnecessary to burden the reader with this material. The automatic-parser output occasionally included node-labels not on the above list (e.g. V, N2), but these were always regarded by the developers of the parser as mistakes.

Briscoe and Carroll's original data included GEIG/unlabelled precision and recall scores for each automatic parse, assessed against the SUSANNE bracketing as gold standard. For the purposes of this experiment, the Evalb program (Sekine & Collins 1997) was used to produce GEIG/labelled precision and recall figures for the same data. In order to be able to compare our LA scores with single GEIG/labelled and GEIG/unlabelled scores for each sentence, we converted pairs of precision (P) and recall (R) figures to F-scores (van Rijsbergen 1979) by the formula $F = 2PR/(P + R)$, there being no reason to include a weighting factor to make precision accuracy count more than recall accuracy or *vice versa*.

One of us (Babarczy) constructed a set of gold-standard trees that could be compared with the trees output by the automatic parser, by manually adding labels from the seven-element Briscoe and Carroll label vocabulary to the SUSANNE bracketing, in conformity with the documentation on that seven-element vocabulary. Because the parsing scheme which the automatic parser was intended to apply was very different from the SUSANNE scheme, to a degree this was an artificial exercise. In some cases, none of the seven labels was genuinely suitable for a particular SUSANNE tagma; but one of them was assigned anyway, and such assignments were made as consistently as possible across the 500-sentence data-set. The admitted artificiality of this procedure did not seem unduly harmful, in the context of an investigation into the performance of a metric (as opposed to an investigation into the substantive problem of automatic parsing).

4. Calculation of Lineage Similarity

Leaf-ancestor assessment depends on quantifying the similarity between pairs of strings, which is done in terms of a variant of Levenshtein distance (Levenshtein 1966), also called edit distance. The Levenshtein distance between two strings is the minimum cost for a set of insert, delete, and replace operations to transform one string into the other, where each individual operation has a cost of one. For instance, the Levenshtein distance between A B C B D and A D C B is two: the latter string can obtained from the former by replacing the second character with D and deleting the last character.

We define similarity between candidate and gold-standard lineages in terms of a variant of Levenshtein distance, in which the cost of a replace operation is not fixed but varies over the interval (0, 2) depending on an application-defined concept of similarity between the two symbols involved in a replacement. In the present experiment, replacement of a symbol by an unrelated symbol costs 2; replacement of a symbol by a different symbol sharing the same first character (e.g. NP for N1 or *vice versa*) costs 1.5. The intuition here is that if two grammatical categories are entirely dissimilar, then for a parser to mistake one for the other amounts to two separate errors of failing to recognize the existence of a tagma of one kind, and falsely positing the existence of another type of tagma (a delete and an insert); but partial credit ought to be given for mistaking, say, a noun phrase for an N-bar. (When LA assessment is deployed in practice, the symbol-replacement cost function should be chosen by reference to the nature of the particular scheme of parsing categories, and to the goals of the application for which parsing is needed.)

If len(*s*) is the length of a string *s*, and ML(*s*, *t*) is the modified Levenshtein distance (under some chosen symbol-replacement cost function) between string *s* and string *t*, then the similarity between candidate and gold-standard lineages *c*, *g* for a given terminal element is computed as 1 — ML(*c*, *g*)/(len(*c*) + len(*g*)), which for any *c*, *g* must fall on the interval (0, 1). The accuracy of a candidate parse is defined as the mean similarities of the lineage-pairs for the various words or other terminal elements of the string.

Applied to our short example sentence, this metric gives the scores for successive terminal elements shown in the left-hand column below:

0.917	*two*	`N1 [ S : NP [ S`
0.583	*tax*	`[ N1 N1 S : NP S`
0.583	*revision*	`N1 ] N1 S : NP S`
0.917	*bills*	`N1 ] S : NP ] S`
1.000	*were*	`S : S`
1.000	*passed*	`S ] : S ]`

The average for the whole sentence is 0.833. For comparison, the GEIG/unlabelled and GEIG/labelled *F*-scores are 0.800 and 0.400.

5. Are the Metrics Equivalent?

The figure for the LA metric just quoted happens to be very similar to one of the two GEIG figures. An obvious initial question about the performance of the metrics over the data-set as a whole is whether, although the metrics are calculated differently, they perhaps turn out to impose much the same ranking on the candidate parses.

To this the answer is a clear no. An extreme case is (2):

2G [S *it is not* [NP *a mess* [S *you can make sense of*]]] G12:0520.27
2C [S *it is not* [NP *a mess* [S *you can make sense*]] *of*] (0.333, 0.333, 0.952)

(Here and below, gold-standard analyses are followed by the SUSANNE location code of the first word — omitting the last three characters, this is the same as the Brown Corpus text and line number. The location for example (1) was A02:0790.51. Candidate parses are followed in brackets by their GEIG/unlabelled, GEIG/labelled, and LA scores, in that order.)

The LA and GEIG numerical scores are very different; but more important than the raw scores are the rankings assigned by the respective metrics, relative to the range of 500 examples. For both GEIG/labelled and GEIG/unlabelled, the parse of (2) is in the tenth (i.e. the lowest) decile; for LA, it is in the second decile. (The ònthÓ decile for any of the metrics, refers to the set of examples occuping ranks 50(*n* — 1) + 1 to 50*n*, when the 500 candidate parses are ordered from best to worst in terms of score on that metric; for instance the second decile is the set of examples ranked 51 to 100. Where a group of examples share identical scores on a metric, for purposes of dividing the examples into deciles an arbitrary order was imposed on members of the group.)

The difference between the low GEIG scores and the high LA score for example (2) is a classic illustration of one of the standard objections to the GEIG metric. The parser has correctly discovered that the sentence consists of an S within NP within S structure, and almost every word has been given its proper place within that structure; just one word has been attached at the wrong level, but because this leads to the right-hand boundary of two of the three tagmas being slightly misplaced, the GEIG score is very low. We believe that most linguistsÕ intuitive assessment of example (2) would treat it as a largely-correct parse with one smallish mistake, not as one of the worst parses in the data-set — that is, the intuition would agree much better with the LA metric than with the GEIG metrics in this case.

An extreme case in the opposite direction (LA score lower than GEIG score) is (3):

3G [S *yes* , [S *for they deal* [PP *with distress*]]] G12:1340.42
3C [T *yes* , [PP *for they deal* [PP *with distress*]]] (1.0, 0.333, 0.262)

The GEIG/unlabelled metric gives 3C a perfect mark — all brackets are in the right place; but its LA score is very low, because two of the three tagmas are wrongly labelled — 0.262 is in fact by a clear margin the lowest LA score for any of the 500 examples. One might of course debate about whether, in terms of the Briscoe/Carroll labelling scheme, the root tagma *should* be labelled S rather than T, but that is not to the point here. The relevant point is that, *if* the target is the gold-standard parse shown, then a metric which gives a poor score to 3C is performing better than a metric which gives it a perfect score.

For example (3), GEIG/labelled performs much better than GEIG/unlabelled. Where parsing errors relate wholly or mainly to labelling rather than to structure, that will be so. But we have seen in the case of (2), and shall see again, that there are other kinds of parsing error where GEIG/labelled is no more, or little more, satisfactory than GEIG/unlabelled.

6. Performance Systematically Compared

In order systematically to contrast the performance of the metrics, we need to focus on examples for which the ranking of the candidate parse is very different under the

different metrics, which implies checking cases whose parses are among the lowest-ranked by one of the metrics. It would be little use to check the highest-ranked parses by either metric. Many candidates are given perfect marks by the LA metric, because they are completely accurate, in which case they will also receive perfect GEIG marks. Some candidates receive perfect GEIG/unlabelled marks but lower LA (and GEIG/labelled) marks, however this merely reflects the fact that GEIG/unlabelled ignores labelling errors.

We have checked how many examples from the lowest GEIG/unlabelled and GEIG/labelled deciles fall into the various LA deciles, and how many examples from the lowest LA decile fall into the various GEIG/unlabelled and GEIG/labelled deciles. These are the results:

LA deciles for GEIG/unlabelled 10th decile:

1st	0
2nd	1
3rd	3
4th	2
5th	2
6th	4
7th	8
8th	7
9th	11
10th	12

LA deciles for GEIG/labelled 10th decile:

1st	0
2nd	1
3rd	1
4th	0
5th	0
6th	1
7th	7
8th	8
9th	13
10th	19

GEIG/unlabelled deciles for LA 10th decile:

1st	1
2nd	0
3rd	4
4th	9
5th	3
6th	8
7th	4
8th	5
9th	4
10th	12

GEIG/labelled deciles for LA 10th decile:

1st	0
2nd	0
3rd	1
4th	1
5th	3
6th	8
7th	5
8th	5
9th	8
10th	19

Clearly there is a tendency for parses assigned poor LA scores also to be assigned poor GEIG scores, and *vice versa*. If there were not, at least one of the metrics could never have been taken seriously by anyone. But there are many exceptions.

The GEIG/unlabelled and GEIG/labelled 10th-decile, LA 2nd-decile example is (2), already discussed above. The GEIG/labelled 10th-decile, LA 3rd-decile example (and this is also one of the GEIG/unlabelled 10th-decile, LA 3rd-decile examples) is (4):

4G [S *then he began* [VP *to speak* [PP *about* [NP *the tension* [PP *in art*] [PP *between* [NP *the mess* [N1 *and form*]]]]]]] G12:0610.27
4C [S *then he began* [VP *to speak* [PP *about* [NP *the tension*]] [PP *in* [N1 *art* [PP *between* [NP *the mess*]] [N1 *and form*]]]]] (0.353, 0.353, 0.921)

The two further GEIG/unlabelled 10th-decile, LA 3rd-decile cases are:

5G [S *Alusik then moved Cooke across* [PP *with* [NP *a line drive* [PP *to left*]]]] A13:0150.30
5C [S *Alusik then moved Cooke across* [PP *with* [NP *a line drive*]] [PP *to left*]] (0.500, 0.500, 0.942)

6G [S [NP *their heads*] *were* [PP *in* [NP *the air*]] *sniffing*] G04:0030.18
6C [S [NP *their heads*] *were* [PP *in* [NP *the air sniffing*]]] (0.500, 0.500, 0.932)

Examples (5) and (6) are essentially similar to (2) above, since all three concern errors about the level at which a sentence-final element should be attached. The LA scores are marginally lower than for (2), because the misattached elements comprise a higher proportion of total words in the respective examples. In (5) a two-word phrase rather than a single word is misattached, and in (6) the misattached element is a single word but the sentence as a whole is only seven words long while example (2) contains ten words. Intuitively it is surely appropriate for a misattachment error involving a higher proportion of total words to be given a lower mark, but for these candidates nevertheless to be treated as largely correct. (Notice that the candidate parse for (5) verges on being a plausible alternative interpretation of the sentence, i.e. not mistaken at all. It is only the absence of *the* before *left* which, to a human analyst, makes it rather certain that our gold-standard parse is the structure corresponding to the writerÕs intended meaning.)

The candidate parse for (4) contains a greater variety of errors, and we would not claim that in this case it is so

intuitively clear that the candidate should be ranked among the above-average parses. Notice, though, that although several words and tagmas have been misattached, nothing has been identified as a quite different kind of tagma from what it really is (as *for they deal with distress* in (3) above was identified as a prepositional phrase rather than subordinate clause). Therefore our intuitions do not clearly suggest that the candidate should be ranked as worse than average, either; our intuitions are rather indecisive in this case. In other cases, where we have clear intuitions, the LA ranking agrees with them much better than the GEIG ranking.

Turning to cases where LA gives a much lower ranking than GEIG: the most extreme case is (3), already discussed. The LA 10th-decile, GEIG/labelled 3rd decile case (which is also one of the GEIG/unlabelled 3rd-decile cases) is (7):

7G [S [NP *its ribs*] *showed* , [S *it was* [NP *a yellow nondescript color*]] , [S *it suffered* [PP *from* [NP *a variety* [PP *of sores*]]]] , [S *hair had scabbed* [PP *off* [NP *its body*]] [PP *in patches*]]] G04:1030.15

7C [T [S [NP *its ribs*] *showed*] , [S *it was* [NP *a yellow nondescript color*]] , [S *it suffered* [PP *from* [NP *a variety* [PP *of sores*]]]] , [S *hair had scabbed off* [NP *its body*] [PP *in patches*]]] (0.917, 0.833, 0.589)

There are large conflicts between candidate and gold-structure parses here: the candidate treats the successive clauses as sisters below a T node, whereas under the SUSANNE analytic scheme the gold-standard treats the second and subsequent clauses as subordinate to the first, with no T node above it; and the candidate fails to recognize *off* as introducing a PP. The presence v. absence of the T node, because it occurs at the very top of the tree, affects the lineages of all words and hence has a particularly large impact on LA score (Sampson 2000: 66 discussed this property of the LA metric).

The three other LA 10th-decile, GEIG/unlabelled 3rd-decile cases are:

8G [S [S *when* [NP *the crowd*] *was asked* [S *whether it wanted* [VP *to wait* [NP *one more term*] [VP *to make* [NP *the race*]]]]] , *it voted no* —[S *and there were* [NP *no dissents*]]] A01:0980.06

8C [T [S [PP *when* [NP *the crowd*] *was asked* [PP *whether it wanted* [VP *to wait* [NP *one more term*] [VP *to make* [NP *the race*]]]]] , *it voted no*] — [S *and there were* [NP *no dissents*]]] (0.952, 0.667, 0.543)

9G [S [S *we wo +n't know* [NP *the full amount*] [S *until we get* [NP *a full report*]]] , *Wagner said*] A09:0520.12

9C [T [S *we wo +n't know* [NP *the full amount*] [PP *until we get* [NP *a full report*]]] , [S *Wagner said*]] (0.909, 0.545, 0.531)

10G [S [S [NP *her husband*] *was lying* [PP *on* [NP *the kitchen floor*]]] , *police said*] A19:1270.48

10C [T [S [NP *her husband*] *was lying* [PP *on* [NP *the kitchen floor*]]] , [S *police said*]] (0.909, 0.727, 0.627)

Each of these involves the same problems as (7) of presence v. absence of a root T node and co-ordinate v. subordinate relationships between successive clauses. Example (8) includes further large discrepancies: *when* and *whether* are both treated as initiating prepositional phrases rather than clauses (though neither word has a standard prepositional use). Example (9) has a similar error involving *until* (this is more understandable, since *until* can be a preposition). Intuitively, to the present authors, the LA metric seems correct in characterizing (8) and (9) as two of the cases where the candidate parse deviates furthest from the gold standard, rather than as two of the best parses. The case of (10) is admittedly less clearcut.

Some readers may find this section unduly concerned with analysts' intuitions as opposed to objective facts. But, if the question is which of two metrics better quantifies parsing success, the only basis for comparison is people's intuitive concept of what ought to count as good or bad parsing. Both metrics give perfect scores to perfect matches between candidate and gold-standard structure; which departures from perfect matching ought to be penalized heavily can only be decided in terms of "educated intuition", that is intuition supported by knowledge and discussion of the issues. It would not be appropriate to lend such intuitions the appearance of objectivity and theory-independence by "counting votes" from a large number of subjects. (Since the GEIG metric is widely known, raw numbers from an exercise like that could have as much to do with the extent to which individual informants were aware of that metric as with pre-theoretical responses to parse errors.) Deciding such an issue in terms of averaged subject responses would be as inappropriate as choosing between alternative scientific theories by democratic voting. Rather, the discussion should proceed as we have conducted it here, by appealing to readers' and writers' individual intuitions with discussion of particular examples.

7. Local Error Information

Different accuracy rankings assigned to complete parses are not the only reason for preferring the LA to the GEIG metric. Another important difference is the ability of the LA metric to give detailed information about the location and nature of parse errors.

Consider, for instance, example (11), whose gold-standard and candidate analyses are:

11G [S *however* , [NP *the jury*] *said* [S *it believes* [S [NP *these two offices*] *should be combined* [VP *to achieve* [N1 *greater efficiency*] [VP *and reduce* [NP *the cost* [PP *of administration*]]]]]]] A01:0210.15

11C [S *however* , [NP *the jury*] *said* [S *it believes* [NP *these two*] [S *offices should be combined* [VP *to* [VP *achieve* [N1 *greater efficiency*] [VP *and reduce* [NP *the cost* [PP *of administration*]]]]]]]] (0.762, 0.667, 0.889)

The score of 0.889 assigned by the LA metric to this candidate analysis, like any LA score, is the mean of scores for the individual words. For this example, those are as follows:

```
1.000  however [ S : [ S
1.000  ,       S : S
1.000  the     [ NP S : [ NP S
1.000  jury    NP ] S : NP ] S
1.000  said    S : S
1.000  it      [ S S : [ S S
1.000  believes S S : S S
0.667  these   NP [ S S S : [ NP S S
0.750  two     NP S S S : NP ] S S
0.667  offices NP ] S S S : [ S S S
1.000  should  S S S : S S S
1.000  be      S S S : S S S
1.000  combined S S S : S S S
1.000  to      [ VP S S S : [ VP S S S
0.800  achieve VP S S S : [ VP VP S S S
0.923  greater [ N1 VP S S S : [ N1 VP VP
   S S S
0.923  efficiency N1 ] VP S S S : N1 ] VP VP
   S S S
0.769  and     [ S VP S S S : [ VP VP VP
0.727  reduce  S VP S S S : VP VP VP S S S
0.800  the     [ NP S VP S S S : [ NP VP
   VP VP S S S
0.769  cost    NP S VP S S S : NP VP VP VP
   S S S
0.824  of      [ PP NP S VP S S S : [ PP
   NP VP VP VP S S S
0.824  administration    PP NP S VP S S S ] :
   PP NP VP VP VP S S S ]
```

The display shows that, according to the LA metric, the early part of the sentence is parsed perfectly, and that the worst-parsed part is *these two offices*. That seems exactly right; in the correct analysis, these three words are a NP, subject of the clause which forms the object of *believe*, but the automatic parser has interpreted *believe* as a ditransitive verb with *these two* as indirect object, and has treated *offices* as grammatically unrelated to *these two*. Intuitively, these are gross mistakes. The next stretch of erroneous parsing is from *achieve* to the end, where each wordÕs mark is pulled down by the error of taking *achieve* to open a subordinate VP within the VP initiated by the preceding *to*. Relative to the SUSANNE scheme used here as gold standard, this also seems a bad error. It is an understandable consequence of the fact that the Briscoe/Carroll automatic parser was based on a parsing scheme that made different theoretical assumptions about grammar, but in the present target scheme no English construction ought to be analysed as ÒVP *to* [VP ...Ó

We would not pretend that our intuitions are so refined that they positively justify a score for *reduce* which is marginally lower than those for the surrounding words, or positively justify the small difference between 0.727 as the lowest score in the second bad stretch and 0.667 in the earlier stretch; the present authorsÕ intuitions are vaguer than that. But notice that, as it stands, the GEIG metric offers no comparable technique for identifying the locations of bad parsing performance within parsed units; it deals in global scores, not local scores. True, one can envisage ways in which the GEIG metric might be developed to yield similar data; it remains to be seen how well that would work. (Likewise, the GEIG/labelled metric could be further developed to incorporate the LA concept of partial matching between label pairs. On the other hand, there is no way that GEIG could be adapted to avoid the unsatisfactory performance exemplified by (2) above.)

Using the LA metric, researchers developing an automatic parser in a situation where large quantities of gold-standard analyses are available for testing should easily be able to identify configurations (e.g. particular grammatical words, or particular structures) which are regularly associated with low scores, in order to focus parser development on areas where further work is most needed. If the parsing scheme used a larger variety of nonterminal labels, one would expect that individual nonterminals might regularly be associated with low scores, though with the very austere nonterminal vocabulary of the Briscoe/Carroll scheme that is perhaps less likely to be so. Even with a small-vocabulary scheme like this, though, one might expect to find generalizations such as ÒWhen a subordinate S begins with a multi-word NP, the parser tends to failÓ Note that we are not saying that this is a true generalization about the particular automatic parser used to generate the data under discussion; one such mistake occurs in the example above, but we do not know whether this is a frequent error or a one-off. Any automatic parser is likely to make errors in *some* recurring patterns, though; the LA metric is potentially an efficient tool for identifying these, whatever they happen to be.

8. Conclusion

From these results, we would argue that the leaf-ancestor metric comes much closer than the GEIG metric to operationalizing our intuitive concepts of accurate and inaccurate parsing.

Someone looking for reasons to reject the LA metric might complain, correctly, that the algorithm for calculating it is much more computation-intensive than that for the GEIG metric. But intensive computation is not a problem in modern circumstances.

More important: there is no virtue in a metric that is easy to calculate, if it measures the wrong thing.

9. Acknowledgment

The research reported here was supported in part by the Economic and Social Research Council (UK) under grant R000 23 8746. We are very grateful to John Carroll for supplying us with the experimental material, generating the GEIG/labelled data, and discussing the material with us.

10. References

Bangalore, S., et al. (1998) ÒGrammar and parser evaluation in the XTAG projectÓ In J.A. Carroll et al., eds., *Proceedings of the Workshop on the Evaluation of Parsing Systems, First International Conference on Language Resources and Evaluation, Granada, Spain, 26 May 1998.*

Black, E., et al. (1991) ÒA procedure for quantitatively comparing the syntactic coverage of English grammarsÓ In *Proceedings of the Speech and Natural Language Workshop, DARPA, February 1991, Pacific Grove, Calif.*, pp. 306-11. Morgan Kaufmann.

Collins, M. (1997) ÒThree generative, lexicalised models for statistical parsingÓ In *Proceedings of the 35th Annual Meeting of the ACL and 8th Conference of the*

European Chapter of the ACL, 7-12 July 1997, Madrid. Morgan Kaufmann.

Levenshtein, V.I. (1966) "Binary codes capable of correcting deletions, insertions, and reversals", *Soviet Physics — Doklady* 10.707-10 (translation of Russian original published in 1965).

Magerman, D.M. (1995) "Statistical decision-tree models for parsing". In *Proceedings of the 33rd Annual Meeting of the ACL, 26-30 June 1995, Cambridge, Mass.*, pp. 276-83. Morgan Kaufmann.

van Rijsbergen, C.J. (1979) *Information Retrieval*, 2nd ed. Butterworths (London).

Sampson, G.R. (1992) The SUSANNE Treebank. Up-to-date release available from `www.grsampson.net` "downloadable research resources"

Sampson, G.R. (2000) "A proposal for improving the measurement of parse accuracy", *International Journal of Corpus Linguistics* 5.53-68.

Sampson, G.R., R. Haigh, & E.S. Atwell (1989) "Natural language analysis by stochastic optimization", *Journal of Experimental and Theoretical Artificial Intelligence* 1.271-87.

Sekine, S. & M. Collins (1997) Evalb software at `www.cs.nyu.edu/cs/projects/proteus/evalb/`.

Evaluating parser accuracy using edit distance

Brian Roark

AT&T Shannon Labs
180 Park Avenue
Building 103, Room E145
Florham Park, NJ 07932-0971
email: roark@research.att.com

Abstract

This paper examines parser evaluation in terms of edit distance, rather than precision and recall. We demonstrate the potential efficacy of edit distance as an evaluation metric with an example, and compare several edit distance scores to labeled precision and recall scores for several statistical parsers from the literature. We suggest a simple schema that has some nice properties. Finally, we discuss the applicability of edit distance metrics for comparing heterogeneous output.

1. Introduction

The dominant metrics in parsing accuracy evaluation are precision and recall, which might suggest that parsing is a classification task, since these metrics are intended to measure classification accuracy. Another way to view parsing is as a recognition task, more analogous to something like speech recognition – recovering hidden structure over an unlabeled, temporally sequenced signal – than classification. From this perspective, we propose adopting the evaluation metric most widely used for speech recognition, namely string edit distance (Wagner and Fischer, 1974). While there is a literature on tree edit distance (Tai, 1979; Zhang and Shasha, 1989), which involves expanding and contracting labeled edges in a tree structure, we will not approach parser evaluation from this perspective. Rather, we will use the simpler string edit distance between string representations of parses. String edit distance as a general framework is very flexible, which will allow us to score parses on more than simply constituent label and span, and to tailor the edit costs to particular needs. We will use it to evaluate how well a parse matches to gold standard, in terms of (i) a sequence of context-free rules corresponding to the tree; (ii) a sequence of constituents with span information; or (iii) the labeled bracketing itself. We will show that the specific distance metrics we will present have some nice properties: they make similar distinctions between parsers to those of standard labeled precision and recall, when the new measures are applied to statistical parsers trained to produce the same kind of annotation; and they provide a conceptually clean way to encode both partial matches and matches across labeling schemas. This paper is not intended to present a specific approach to evaluating specific parsers; rather a general approach that, in our opinion, has the best chance to provide a unified evaluation framework for the parsing community.

Before continuing further, let us define the terms and metrics that we are going to be using in the course of this paper. A parse can be represented as a labeled bracketing[1], e.g.

```
(S  (NP (DT The) (NN dog) )
    (VP (VBD barked) ) )
```

Each left parenthesis, which we will also refer to as an open bracket, is associated with a label and a right parenthesis, which we will also call a close bracket. Each open and close bracket pair denotes a constituent in the parse tree. Each constituent has a span, which is the part of the string that is at the leaves of the sub-tree rooted at the constituent. For example, the NP constituent in the example spans the words "The" and "dog". Each local tree in the structure, i.e. the constituent label and the labels of its children, constitutes a context-free rule instance. For example, the highest local tree in the above labeled bracketing is an instance of the rule S $\rightarrow$ NP VP.

A parser returns a labeled bracketing of a string, which is evaluated with respect to a hand-labeled "gold standard" annotation. The most widely used metrics to evaluate the accuracy of the parser come from the PARSEVAL recommendations (Black et al., 1991); among these, labeled precision and recall are principal. The labeled measures do not count part-of-speech (POS, also known as pre-terminal) labels, which are those non-terminals with one and only one terminal child. In the above example, the accuracy of the DT, NN, and VBD labels would not be included in the labeled precision and recall scores. Of the other constituents in the labeled bracketing (S, NP, and VP), each would be compared with the constituents in the gold standard parse. If they match the label and the span of an otherwise unmatched constituent in the gold standard parse, then they are counted as correct, otherwise incorrect. Labeled precision (LP) is the number of correct constituents divided by the total number of constituents in the parser's labeled bracketing (excluding, of course, POS). Labeled recall (LR) is the number of correct constituents divided by the total number of constituents in the gold standard labeled bracketing. Often these are combined into a single measure, called the F-measure, according to the following formula:

$$2(\mathrm{LR})(\mathrm{LP})/(\mathrm{LR} + \mathrm{LP}) \quad (1)$$

[1] The labeled bracketing convention that will be used for examples in this paper are those from the Penn Treebank (Marcus et al., 1993), but that is not intended to restrict the scope of this paper to these annotations.

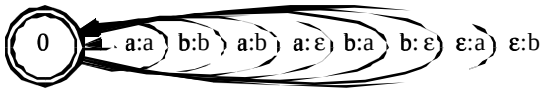


Figure 1: Example edit distance transducer.

The E-measure is 100 minus the F-measure. This will come in handy when comparing precision and recall based measures with error rates based on edit distance. There are other measures in the PARSEVAL recommendations, such as the crossing brackets scores, but for the purpose of this paper, we will focus on labeled precision and recall.

We will present string edit distance as weighted finite-state transduction[2]. Let Σ_0 be the alphabet of the input string $S_0 \in \Sigma_0$*, and Σ_1 be the alphabet of the output string $S_1 \in \Sigma_1$*. Let T be a finite state transducer that maps (i) all $a \in \Sigma_0 \bigcap \Sigma_1$ to themselves at no cost; (ii) all $a \in \Sigma_0$ to $b \in \Sigma_1$ at cost 1 (substitution); (iii) all $a \in \Sigma_0$ to the empty string at cost 1 (deletion); and (iv) the empty string to all $b \in \Sigma_1$ at cost 1 (insertion). Then the edit distance between S_0 and S_1 is the least cost path in $S_0 \circ T \circ S_1$.

Figure 1 presents a transducer for (a+b)*, which, when used according to the previous paragraph, returns the edit distance between two strings. Each arc is labeled with an input symbol i, and output symbol o and a cost c, as follows: $i{:}o/c$. The empty string has label ϵ. Only those arcs which map characters to themselves are cost free. All other arcs – substitution (a:b), deletion (a:ϵ), and insertion (ϵ:b) – have cost 1. Thus, for example, the edit distance between an input string "abab" and an output string "aabb" is two[3].

Of course, it is up to us what the alphabets and strings are. In the case of speech recognition, where edit distance is used to calculate the word error rate (WER), the strings are strings of words in the lexicon, and the transduction maps words to words or the empty string. For the case of parser evaluation, we might consider strings of rules, strings of constituents with span, or strings consisting of pieces of the labeled bracketing. It is also up to us what cost to put on the arcs. In the above figure, all edit costs are one, but there may be cases where the cost of an edit might be judged to be less than one.

The rest of the paper will be structured as follows. First we will motivate the use of edit distance with an example where precision and recall give scores that fail to correspond to intuitive notions of similarity, which an edit distance score is closer to. Then we will present three edit distance measures, and how they perform on four parser implementations from the statistical parsing literature. Finally, we will discuss how this approach can handle more complicated schemas that might address some of the cross-system issues.

[2] For an introduction to finite-state transducers, see, e.g. Jurafsky and Martin (2000).

[3] Note that there is more than one derivation here with the same cost, e.g. a:a b:a a:b b:b (two substitutions) and a:a b:ϵ a:a b:b ϵ:b (one deletion and one insertion).

2. Motivation

One perceived failure of precision and recall measures was presented in Bangalore et al. (1998), namely that shallow parses that do not make strong attachment choices are penalized less than parses that are quite close to the gold standard, but that make some wrong attachment decisions. Their example is repeated here, with node labels on the bracketing.

```
1. (S (NP (PRN She))
      (VP (VBD bought)
          (NP (NP (DT an) (JJ incredib ly)
                  (JJ expensive ) (NN coat))
            (PP (IN with)
              (NP (NP (JJ gold) (NNS buttons))
                  (CC and)
                  (NP (NN fur) (NN lining)))) )
          (PP (IN at)
              (NP (DT the) (NN store)))))

2. (S (NP (PRN She))
      (VP (VBD bought)
          (NP (NP (DT an) (JJ incredib ly)
                  (JJ expensive ) (NN coat))
            (PP (IN with)
              (NP (NP (JJ gold) (NNS buttons))
                  (CC and)
                  (NP (NP (NN fur) (NN lining) )
                      (PP (IN at)
                         (NP (DT the)
                            (NN store)))) )))))

3. (S (NP (PRN She))
      (VP (VBD bought)
          (NP (DT an) (JJ incredibly)
              (JJ expensive ) (NN coat))
          (IN with)
          (NP (JJ gold) (NNS buttons) )
          (CC and)
          (NP (NN fur) (NN lining))
          (PP (IN at)
              (NP (DT the) (NN store)))))
```

The gold standard parse (1) has the final prepositional phrase modifying the verb phrase, while parse 2 mistakenly attaches the PP to the lowest noun phrase. Other than this, though, parse 2 is quite close to the gold standard. Parse 3 is a much shallower parse, with much of the hierarchy in the gold standard left out. Using standard labeled precision and recall against the gold standard parse (1), parse 2 achieves 72.73 labeled recall and 66.67 labeled precision, or 69.56 F-measure. In contrast, parse 3, which is much farther from the gold standard, achieves 72.73 labeled recall and 100.0 labeled precision, or 84.21 F-measure. The clear reason for this is that judging correctness based on span ignores dominance relationships, so that parse 3 identifies many correct constituents, despite getting the immediate dominance relationships quite wrong (according to the gold standard).

One way to include the dominance relationships is to encode the parse as a sequence of rules (top-down, left-most). This uniquely identifies the tree, and serves as a string for input into the simple edit distance approach outlined in the previous section. We will uniformly give a cost of 1 for each insertion, deletion, and substitution, without giving credit for substitutions between more or less similar

Gold parse rules	parse (2) rules	(2) edits	edit type	parse (3) rules	(3) edits	edit type
S → NP VP	S→NP VP			S→NP VP		
→ PRN	→ PRN			→ PRN		
→ VBD NP PP	→ VBD NP	1	substitution	→ VBD NP IN NP CC NP PP	1	substitution
→ NP PP	→ NP PP				1	deletion
→ DT JJ JJ NN	→ DT JJ JJ NN			→ DT JJ JJ NN		
→ IN NP	→ IN NP				1	deletion
→ NP CC NP	→ NP CC NP				1	deletion
→ JJ NNS	→ JJ NNS			→ JJ NNS		
	→ NP PP	1	insertion			
→ NN NN	→ NN NN			→ NN NN		
→ IN NP	→ IN NP			→ IN NP		
→ DT NN	→ DT NN			→ DT NN		
Total:		2			4	

Table 1: Edits from gold standard for aligned rule sequences.

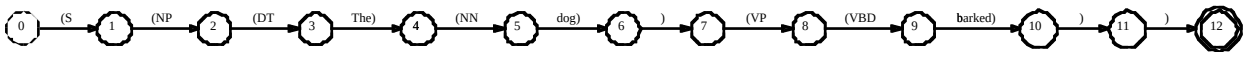


Figure 2: Labeled bracketing encoded as a string

rules. This may appear somewhat coarse, but it is merely a starting point. If it is felt that certain rules are, in some sense, closer than others, then this can be represented in the amount of cost that is associated with that arc in the transducer. The approach is general enough to allow for different costs. To begin, we will count each deletion, substitution, and insertion as one edit, and present the error rate, i.e. the total edit cost per 100 rules of gold standard. This is the same measure that is used for accuracy in the speech recognition community (word error rate).

One note about the encoding: because every left-hand side category (except S at the root) also occurs on a right-hand side, we omit the left-hand side category (except the initial S), so as to avoid doubling errors. Because of the rule ordering, there is no ambiguity with respect to the left-hand sides.

Table 1 shows the aligned sequences of rules, and the edit cost accumulated by both parse 2 and parse 3 under the standard cost. Recall that the symbols that are being compared are entire rules, not individual non-terminals within the rules. Hence each candidate parse has a single substitution for the verb phase expansion, despite the fact that the parse 2 VP expansion is in some sense closer to the original than that of parse 3. In this sense, as was stated above, the metric is somewhat coarse. Nevertheless, we can see that parse 2 accrues one substitution and one insertion, for an error rate of 18.2 (2 edits for 11 rules), whereas parse 3 accrues one substitution and three deletions for a 36.4 error rate. These scores better correspond to our intuition, and the one stated in Bangalore et al. (1998), that parse 2 is closer to the gold standard than parse 3.

What this example demonstrates is that there are circumstances where precision and recall can unduly penalize parses, and others where they do not penalize parses enough. Edit distance between sequences of rules does the right thing, at least in these cases. Note, however, that edit distance does not need to be over sequences of rules. In fact, we can calculate edit distance over constituent spans, by replacing the rules in the table above with the left-hand side and word span positions. In the next section, we will present a third edit distance approach, which treats the labeled bracketing itself as a string. This will be followed by some parser evaluations using all three of our proposed edit distance approaches.

3. Labeled bracketing edit distance

Before presenting our method for calculating edit distance between labeled bracketings, let us first discuss how to encode them as strings, i.e. what are the tokens? We will adopt what was used in Roark (2001b) for storing automatically generated treebanks. Every open bracket has a label, so an open bracket plus its label is one token. Terminal items (i.e. lexical items) are always followed by a close bracket (at least in Penn Treebank annotation), so terminal plus close bracket is one token. Finally, close brackets for non-POS constituents, which do not have an associated terminal item, are each tokens. In this way, the labeled bracketing can be encoded as a string. Figure 2 shows our original labeled bracketing encoded as a string with this tokenization.

To calculate the edit distance between labeled bracketings of the same string, we are going to exploit some characteristics of the bracketing. First, we know that the terminal items are the same in the input and the gold standard. Hence there is no need to provide any arcs for terminal items other than those mapping them to themselves. Next, we assume that both the input and the gold standard are balanced, i.e. there are the same number of right and left parentheses. What this means is that, for every successful mapping between one parse to the other, every deleted open bracket must be matched with either an inserted open bracket or a deleted close bracket, to maintain balanced bracketing. Indeed, any bracket insertion or deletion must be paired with another edit.

	Open Bracket	Close Bracket
Delete	(X:ϵ/0.5	):ϵ/0.5
Insert	ϵ:(X/0.5	ϵ:)/0.5

Table 2: Labeled bracketing edit cost table

If we used a standard edit cost of 1, then we would get a double count for every deleted or inserted constituent. If, however, we make the cost of deleting or inserting open (labeled) brackets or close (non-terminal) brackets one half, we can rely on the balanced brackets property to ensure that we ultimately get a cost of one for the error. Table 2 presents this cost schema. One nice side effect of this is that substitution can be considered simply as a deletion plus an insertion, and hence automatically given an edit cost of 1, without having to explicitly put substitution arcs into the transducer, since deleting a labeled bracket and inserting an alternatively labeled bracket has cost 1 under this schema. We can, however, if we wish, allow substitution at no cost. For example, if we do not wish to include POS tags in the evaluation, we can allow any POS tag to rewrite as any other POS tag with no cost. Since the close brackets for POS tags are terminals, which cannot be inserted or deleted, and since the yield for the two parses are identical, this free substitution means that no POS tagger error will result in any cost for the least cost transduction. Let us refer to as the *basic labeled bracketing edit transducer* that which maps (i) all symbols to themselves at no cost, (ii) all non-POS open bracket non-terminals to epsilon and vice versa at cost 0.5, (iii) the close bracket symbol to epsilon and vice versa at cost 0.5, and (iv) all POS open bracket non-terminals to all other POS open bracket non-terminals at no cost.

There does remain one serious problem with this approach, however. Referring back to our motivating example in the previous section, one can see by inspection that the least cost path under this approach between the gold standard (parse 1) and parse 2 will involve deleting one NP constituent (one open and one close bracket) and inserting and deleting three close brackets. In other words, the misattachment of the PP is costing just as much as it would under precision and recall evaluation. A better solution is to allow certain kinds of constituent movement to count as just one edit. Deleting and inserting consecutive close brackets in effect allows constituents to be moved higher or lower in the hierarchy. Thus, in parse 2, the deletion of the three close brackets from their current location and their insertion into the correct location – how the movement of the constituent is effected – should count as a single edit. We can do this with a simple additional pass over the least cost path through the composition with our basic labeled bracketing edit transducer.

This basic transducer cannot simply be changed to give less cost to deletion and insertion of multiple consecutive close brackets, because those brackets may be paired with open brackets or non-consecutive close brackets. We only want to give reduced cost to those consecutive brackets that are matched with other consecutive brackets. We could, in advance of composition with the basic labeled bracketing

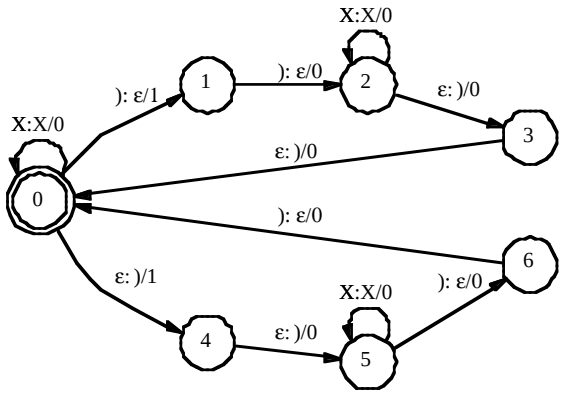


Figure 3: Example transducer giving cost 1 to multiple consecutive bracket insertion and deletion. 'X' is intended to range over all symbols.

edit transducer, pass the input through an arbitrary number of transducers that, for any given k, map the input to an output by deleting k consecutive close brackets and inserting k consecutive close brackets, at a cost of 1. Figure 3 shows one such transducer, for two consecutive brackets. Because an arbitrary number of these would need to apply in order to cover all possible movements, this is not a particularly useful approach.

Luckily, we can find the exact cost that corresponds to this approach, without having to actually perform these transductions, very quickly in the following manner:

1. Find the least cost path p through the composition with our basic labeled bracketing edit transducer
2. Calculate the total cost c of the edits in p
3. Find the length of the longest consecutive string of deleted close brackets m and the length of the longest consecutive string of inserted close brackets n in p
4. $r = \min(m,n)$
5. If $r \geq 2$
   a) remove r consecutive deleted and r consecutive inserted close brackets from p
   b) $c = c - (r - 1)$
   c) return to 3.
5. return c

This will yield the same edit cost that would have resulted from applying the arbitrary transductions described in the previous paragraph. To see this, one must recall that all consecutive close bracket symbols come immediately after terminal items. Because the terminal items cannot be inserted or deleted, this means that the only way to reposition close brackets into the correct location is to delete them and re-insert. The deletion or insertion of any other symbols will not change this. Hence the deletion and insertion of consecutive close brackets will be part of the least cost path through the basic transducer. If there exist consecutive deleted brackets and consecutive inserted brackets of the same length, then there would have been a better path between the input and gold standard parses that involved pairing these consecutive brackets, and hence ac-

Parser	F-measure	Label edit	Span edit	Rule edit
Charniak (2000)	89.74	87.95	86.5	85.6
Collins (2000)	89.71	87.94	86.4	85.5
Collins (1997)	88.24	86.45	84.5	84.0
Roark (2001)	86.71	85.04	82.9	83.3

Table 3: Comparison of four parsers with four measures of accuracy

cruing a single edit. The above algorithm will find all such instances, and adjust the cost accordingly.

We will call the score that results from this method of calculating the edit distance between two labeled bracketings, which counts movements of constituents as a single edit, "Label edit distance" to contrast it with the other edit distance scores that we have discussed, "Rule edit distance", which is based on the edit distance between ordered sequences of rules, and "Span edit distance", which is based on the edit distance between ordered constituents with span and label.

In the next section, we will present scores using different measures for four treebank-style parsers from the literature. This will be followed by a discussion of how these approaches generalize.

4. Evaluation

To evaluate these edit distance measures, we will measure performance of the output from four different parsers. Each of the four parsers was trained on sections 2-21 of the Penn Wall St. Journal Treebank, and tested on section 23. We can thus compare the parsers with each other using different measures, and compare the measures by looking at the distinctions that are made by the measures between the variously performing parsers.

The parsing outputs are those taken from the latest version of the parser presented in Charniak (2000), and the output from Collins (2000), Collins (1997), and Roark (2001a)[4]. For each of the edit distance measures – label, rule, and span – we can define the error rate, which is the number of edits per 100 events (i.e. rules or constituents) in the gold standard parse. The accuracy is then defined as 100 minus the error rate. The accuracy measures that we used were: (i) F-measure; (ii) Label accuracy (based on label edit distance); (iii) Span accuracy (based on span edit distance); and (iv) Rule accuracy (based on rule edit distance). To avoid penalizing the parsers for POS tagging errors, the POS tags in rule instances used for the rule edit distance were replaced with the terminals. Table 3 shows the four different scores of the four parsers.

All three of our edit distance metrics are harder on the parsers than the F-measure score, with rule edit distance giving the lowest scores overall. This is perhaps not surprising given the particular test domain, since the Penn Treebank annotation is known for having many flat constituents, most notably the base NP constituents. In this case, then, we can have structures with no rules correct, but some constituent span predictions correct.

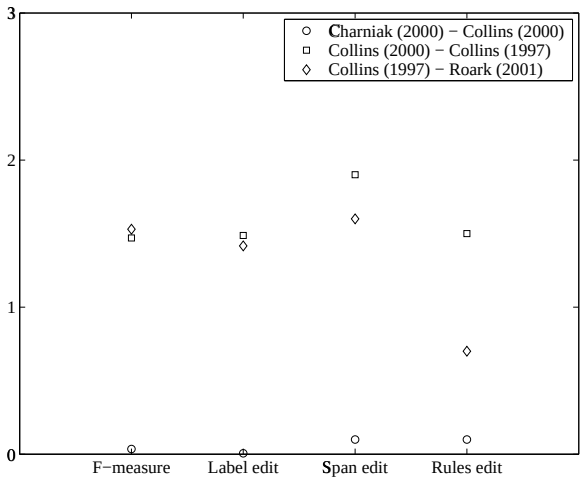


Figure 4: The difference between accuracy scores with F-measure and the three edit distances, for the four parsers: Charniak(2000) minus Collins (2000); Collins (2000) minus Collins (1997); and Collins (1997) minus Roark (2001)

It is interesting to note that, despite the differences in scores – with F-measure giving the highest scores, and rule accuracy generally the lowest scores – there is always the same ordering among parsers. To see how close the distinctions between the parsers are under the four different measures, in figure 4 we plotted the differences in score between parsers ranked n and (n+1), for each of the four measures. As one can see, the distinctions that are being made between parsers are remarkably consistent between the measures. The only exception to this is that the Roark (2001) parser does relatively better in rule accuracy – in fact half a point better than span accuracy, while all the other parsers had worse rule accuracy than span accuracy. This may be due to the fact that the Roark parser is a left-to-right incremental parser, making predictions about children of a left-hand side given the context, in contrast to the others, which also make predictions about heads of constituents given governing heads. Thus it may tend to do a better job choosing children of the left-hand side than it does choosing specific constituents. Alternately, it might be that this parser is making more grave attachment errors than the others, and hence benefitting from the rule-oriented evaluation, which has been shown to be more lenient on these errors. If this were true, however, we would expect the labeled bracketing accuracy to be similarly affected, which it is not.

This is all well-and-good for overall scores, but how do these measures differ on a sentence by sentence basis. Figure 5 plots each sentence from Charniak (2000), with its labeled bracketing error rate versus its E-measure, i.e. 100 minus F-measure. Along the line the scores are identical[5].

[4]The F-measure accuracy for the Charniak parser is 0.2 better than the published result. In addition to these four parsers, we also attempted to obtain the output from one other high accuracy parser, but were unable to do so.

[5]The labeled bracketing error rate goes above 100, because there can be more edits required than there are non-terminals in the parse, e.g. if there is nothing right.

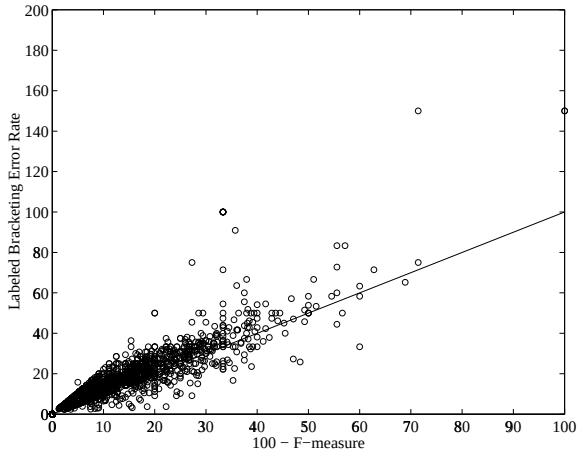


Figure 5: 100 - F-measure (E-measure) versus labeled bracketing edit rate for each sentence in section 23 from Charniak (2000)

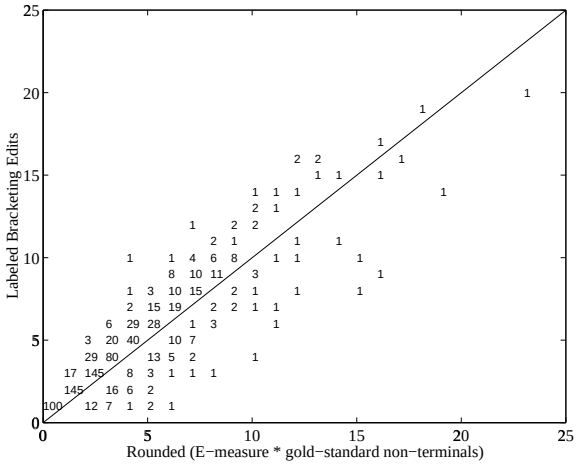


Figure 6: Number of sentences at (x,y), where x is E-measure * non-terminals rounded to the nearest integer, and y is labeled bracketing edits.

Most of the parses are close, but there are some outliers. A brief inspection of some of the outliers provides no surprises. Those out along the x-axis result from constituent attachment decisions that make many constituents incorrect with respect to their span. Those closer to the y-axis result from large differences in precision and recall – i.e. perfect precision and poor recall or vice versa – which tend to give higher F-measure scores than if we simply count the number of edits required to match the gold standard parse.

Most of the parses that have very divergent scores are quite small, so that a small number of required edits accounts for a large percentage of the non-terminals in the parse. These do not account for much when it comes to overall performance, because the raw numbers that they contribute are small. To try to get a better sense of where the bulk of the differences are, we plotted in figure 6 the raw numbers that fall into particular buckets. For the edit distance score, we have the number of edits. To approximate a number of errors in the precision and recall case, we took the E-measure times the number of non-terminals in the gold standard parse, and rounded it to the nearest integer. At each point (x,y), where x is the rounded approximation to errors from the E-measure, and y is the number of edits required for the labeled brackets, we plotted the number of sentences out of the test set of 2416 sentences that fall in that bucket. We omitted those sentences where x = y. As one can see, the bulk of the sentences that differ do so by one or two. Those that are greatly different are relatively rare.

To summarize this section, we have presented three evaluation measures based on edit distance. All three give lower scores than labeled precision and recall to the parsers evaluated, yet they maintain similar distinctions between the parsers. That is, these measures are just as useful as precision and recall in discriminating between parsers in the domain where precision and recall has been used most widely and most successfully. The labeled bracketing and rules edit rate scores give better scores than precision and recall for certain kinds of errors, although these parsers do not seem to be making these errors with much frequency.

In the next section, we will discuss how edit distance scoring allows for very natural generalizations for such things as heterogeneous labeling schemas.

5. Discussion

For the case of labeled bracketing, an edit distance approach provides a very natural way to extend evaluation to include such things as partial matches or equivalence classes. For example, suppose the gold standard parse includes things such as function tags, e.g. not simply NP, but NP-SBJ for subject NPs, or PP-TMP for temporal PPs. One may want to impose some cost for failure to label these tags onto the non-terminal node. This cost may be less than that imposed for a complete mismatch. In order to do this, all that one needs to do is include a transition in our transducer to map from one to the other with the desired cost. Similarly, one may want to include POS tags in the evaluation, but impose less of a cost for mis-tagging something that should be NN as, say, NNP, than one would for mistagging it as VBD. Errors in POS tagging as a whole could be assigned less cost than errors in non-POS labels.

Alternately, one might be more interested in whether or not the parser finds the categories of a shallow parse accurately. In this case, one might not impose any cost for deletion, just for insertion, i.e. more structure is okay. Some might consider this “dumbing down” the parser, but in fact getting very flat constituents high in the tree correct can be more difficult than making finer distinctions lower in the tree. For purposes such as information extraction, for example, a lot of hierarchical structure may be less important, which would lead one to an evaluation of this sort. Making deletion cost free is very straightforward.

Perhaps most importantly, edit distance generalizes to any sequence of parser decisions, including those of, say, a dependency parser. Given the left-to-right ordering of the words, we can stipulate an ordering among dependency relations to give an overall ordering of the dependencies. Edit distance can then straightforwardly apply in the same way

as it did to the sequence of rules. This approach was taken in evaluating a natural language generation system in Bangalore et al. (2000).

One benefit of using edit distance on a simple sequence of rules, constituents with span, or dependency relationships, is that existing error rate software, such as those used for speech recognition evaluation, can be used as is on the output. This is actually how the evaluation from the previous section was carried out for the rule edit distance and the constituent span edit distance[6]. This is beneficial, not only because one does not have to write new code to evaluate the output, but also because these evaluation routines often can output interesting diagnostic information, including frequent substitutions, merges, or splits.

One comment that is obvious, but deserves to be made nonetheless, is that the evaluation should fit the task. Why should the degree to which a parser can identify dependencies be a better evaluation than the degree to which it can identify constituents? Only insofar as some particular task requires lexical dependencies, not constituents. Parsing, unlike speech recognition, is not generally viewed as a task in and of itself. Rather, it is taken to be in service of something else, e.g. some kind of semantic processing or language modeling. This can also be true of the output of a speech recognizer – the recognized words might be used to attempt to classify the utterance, for example. In that case, it may be useful to measure WER, but ultimately the efficacy of the particular recognizer will be found in the classification accuracy. If an improvement can be had in recognizer accuracy that makes no difference in classification accuracy, then why bother. The same is true in parsing. The danger with embedding a parser in another system for evaluation is that it becomes more difficult to control for the other parts of the system. It thus is beneficial to be able to evaluate the output of the parser independently – but presumably with respect to its ability to recognize the hidden structures that are used by the system. In the absence of such a parser-independent criterion, objective evaluation is difficult. In its presence, error rate and accuracy based on edit distance is a good alternative for evaluation.

In this paper, we have presented three edit distance measures that can be used to evaluate parsers. The intent is not to exhaust the possibilities of the approach, but rather to show that, as a general framework, edit distance can provide the flexibility to meet the varied demands of parser evaluation. These measures can make similar distinctions between parsers as is made by precision and recall, yet repair some long-standing weaknesses of them. Edit distance can be applied to a variety of string representations of trees, including an ordered list of constituents with span, an ordered list of context-free rules, or even the labeled bracketing itself.

6. Acknowledgements

Thanks to Eugene Charniak and Michael Collins for sharing the outputs of their respective parsers. Thanks also to Mehryar Mohri and Srinivas Bangalore for useful discussion on this topic.

[6]Since the labeled bracketing edit distance required a special transducer, it was evaluated differently.

7. References

Srinivas Bangalore, Anoop Sarkar, Christine Doran, and Beth Ann Hockey. 1998. Grammar and parser evaluation in the xtag project. In *Workshop on the Evaluation of Parsing Systems*.

Srinivas Bangalore, Owen Rambow, and Steve Whittaker. 2000. Evaluation metrics for generation. In *International Conference on Natural Language Generation*.

Ezra Black, Steven Abney, Dan Flickenger, Claudia Gdaniec, Ralph Grishman, Philip Harrison, Donald Hindle, Robert Ingria, Frederick Jelinek, Judith Klavans, Mark Liberman, Mitchell P. Marcus, Salim Roukos, Beatrice Santorini, and Tomek Strzalkowski. 1991. A procedure for quantitatively comparing the syntactic coverage of english grammars. In *DARPA Speech and Natural Language Workshop*, pages 306–311.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Conference of the North American Chapter of the Association for Computational Linguistics*, pages 132–139.

Michael J. Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 16–23.

Michael J. Collins. 2000. Discriminative reranking for natural language parsing. In *The Proceedings of the 17th International Conference on Machine Learning*.

Daniel Jurafsky and James H. Martin. 2000. *Speech and Language Processing*. Prentice-Hall, New Jersey.

Mitchell P. Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Brian Roark. 2001a. *Robust Probabilistic Predictive Syntactic Processing*. Ph.D. thesis, Brown University. http://arXiv.org/abs/cs/0105019.

Brian Roark. 2001b. Storing automatically generated treebanks in lattices of derivations. In *Proceedings of the IRCS Workshop on Linguistic Databases*, pages 210–218.

Kuo-Chung Tai. 1979. The tree-to-tree correction problem. *Journal of the ACM*, 26(3):422–433.

Robert A. Wagner and Michael J. Fischer. 1974. The string-to-string correction problem. *Journal of the ACM*, 21(1):168–173.

Kaizhong Zhang and Dennis Shasha. 1989. Simple fast algorithms for the editing distance between trees and related problems. *SIAM J. Comput.*, 18(6):1245–1262.

Evaluating Syllabification: One Category Shared by Many Grammars

Karin Müller

Department of Computational Linguistics
University of Saarland
Saarbrücken, Germany
kmueller@coli.uni-sb.de

Abstract

We apply a series of context-free grammars to syllabification by using a supervised training method. In our experiments, we investigate various phonological grammars, which strongly differ in structure. A simple evaluation metric "word accuracy" supports grammar development by denoting an increasing performance for grammars enriched with linguistic structure. This evaluation, judging one single category shared by all grammars, is in strong contrast to PARSEVAL, which is designed for a single grammar evaluating (almost) all categories. Using a toy-treebank, we show that the PARSEVAL measures are hard to interpret, since the results are inconsistent with one another. It turns out that evaluating only a limited number of categories (here only one single category) is a harder evaluation measure than measuring the precision of all occurring substructures of a grammar.

1. Introduction

In computational linguistics, the PARSEVAL measures suggested by Black et al. (1991) are now the standard measure for evaluation of context-free grammars (CFGs). The measures quantify precision and recall of common parenthesis based on a treebank. The metrics focus on the precision of all substructures that are specified by a CFG. However, the PARSEVAL metrics are not suitable for all problems that can be described with probabilistic context-free grammars (PCFGs) especially in cases when partial structures are more interesting. In comparison to the field of parsing, there are no phonological treebanks of transcribed words. However, large pronunciation dictionaries are available which can be exploited for evaluation and training. We develop a series of grammars in our supervised experiments and evaluate them on partial structures. There are two main reasons why we chose an evaluation procedure different from PARSEVAL. Firstly, if we had chosen the PARSEVAL evaluation metrics, a separate evaluation suite would have to be constructed for each grammar type. This would be very time-consuming. Secondly, we chose alternative evaluation metrics because syllabification tasks are usually evaluated either by syllable accuracy or even word accuracy, i.e. partial structures of a phonological tree are evaluated. Syllable accuracy means that each syllable is compared with an annotated syllabified corpus. If the system predicts the syllable boundary correctly, syllable accuracy increases. A stricter variant of measuring the capability of a system is to determine word accuracy, which means that each syllable boundary has to be predicted correctly within a word. We try to solve the evaluation problem by annotating and evaluating those structures which are usually referred to in the literature linked to syllabification, and which are shared by all grammars.

The paper is organized as follows: in Section 2, we introduce the syllabification task, as well as a series of phonological grammars describing German syllable structure. Section 3 discusses our evaluation measure in comparison to PARSEVAL. In Section 4, we conclude.

2. Syllabification

In text-to-speech (TTS) systems, like those described in Sproat (1998), the correct pronunciation of unknown and novel words is a crucial problem. Thus, TTS systems usually use large pronunciation dictionaries, however there are in all languages productive word formation processes which generate words that are new to the system. The correct pronunciation of a new word is not only dependent on the correct identification of phonemes but also on the correct assignment of syllables. Van Santen et al. (1997) showed that location in the syllable influences the duration of a phone. Furthermore, identifying syllables boundaries is essential for the application of phonological rules (Kahn (1976), Blevins (1995)), which is certainly the case e.g. for German syllable-final devoicing. Thus, we are interested in developing models that predict syllable boundaries of unknown words as well as possible. This means for context-free grammars, that we need a single category e.g. "SYL" (i) which spans a whole syllable, (ii) occurs in all grammars and (iii) which can be evaluated easily.

In our approach, we developed several context-free grammars and trained them on large automatically transcribed corpora extracted from newspaper corpora by looking up the words and their transcriptions in the pronunciation dictionary CELEX (Baayen et al. (1993)). The different grammars describe the internal structure of words and can be used to predict syllable boundaries after a training procedure. In our experiments, we use a supervised training method which is a combination of treebank and bracketed corpora training (Müller (2001)) exploiting the syllabification information of a pronunciation dictionary and the frequency information of a training corpus consisting of 182 000 words.

We investigate six different grammars to predict syllable boundaries by introducing new categories for each new grammar.

Treebank grammar. The first grammar describes a word as a sequence of syllables consisting of one or n phonemes. The analysis at the top of Figure 1 dis-

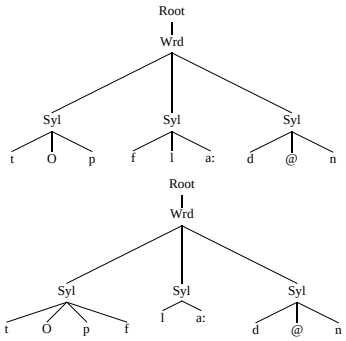


Figure 1: Treebank grammar

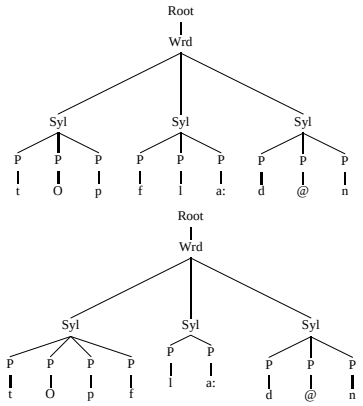


Figure 2: Phoneme grammar

plays a possible syllabification of the German word "Topfladen" ([tOpfla:d@n], which can be either translated by *top chapatti* (Top-fladen), or *pot shop* (Topfladen) (of course there are additional possible syllabifications)

Phoneme grammar. The second grammar introduces an abstract level between the phonemes and the syllables. Each phoneme is labeled by an abstract phoneme label. The grammar learns information about the complexity of a syllable. Figure 2 shows two possible analyses of the phoneme string [tOpfla:d@n] according to the phoneme grammar.

Consonant-vowel grammar. The third grammar distinguishes between consonants and vowels by labeling all phonemes either by a C or a V label. The grammar also demands a vowel inside of a syllable. The structure of the grammar is exemplified by Figure 3 displaying two possible syllabifications of the phoneme string [tOpfla:d@n].

Syllable structure grammar. The fourth grammar specifies syllable structure in more detail. Syllables are split into onset, nucleus and coda. The probability of a consonant depends on its occurrence in the onset or the coda. Two example trees of the phoneme string [tOpfla:d@n] are shown in Figure 4.

Positional syllable structure grammar. The fifth grammar further describes a phoneme depending on the position of the syllable within the word, and depending on the position of the phoneme within the syllable by enumerating the consonants. There are four possible positions of the syllable: word-initial, word-medial, word-final, and monosyllabic words. The consonants of the phonemes are enumerated according to their position inside of the syllable onset, or coda. The structure of the grammar is examplified in Figure 5

Advanced positional syllable structure grammar. An additional feature, cluster size is added to the last grammar. Thus, the consonants depend on their position within the cluster, and the size of the cluster. Figure 6 displays two examples.

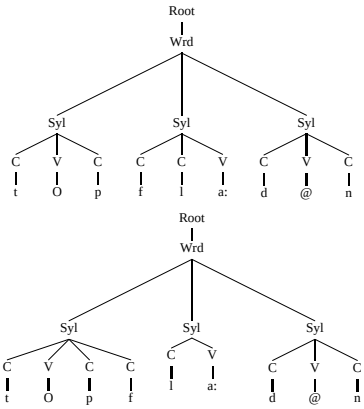


Figure 3: Consonant-vowel grammar

In a next step, the grammars are trained using a novel algorithm consisting of a combination of bracketed corpora and treebank training (see Müller (2001)). However, in contrast to older experiments (where we trained on a series of training corpora ranging from 4 500 to 2.1 million words), we use for our new experiments a fixed training corpus consisting of 182 000 words. Training on this corpus provides a probabilistic version for each of the six phonological grammars. Since all grammars have in common that they are written to predict syllable boundaries, they share the category "SYL" which spans a whole syllable. After training, we can use the most probable parse of a word, the so-called Viterbi parse, to read off the syllables of this word: all phonemes under a syllable node "SYL" belong to one syllable. In Section 3, we describe the performance for the trained grammars on a syllabification task using a huge evaluation corpus of about 240 000 words. Moreover, we try to relate these results to an evaluation using PARSEVAL measures for a toy-treebank.

3. Evaluation

As already presented in Section 2, our system is designed to predict syllable boundaries using probabilistic phonological grammars. For parsing, we used the implementation of Schmid (2000). Our evaluation corpus consists of about 242 000 correctly syllabified words. For syllabification of these words, we used the CELEX dictio-

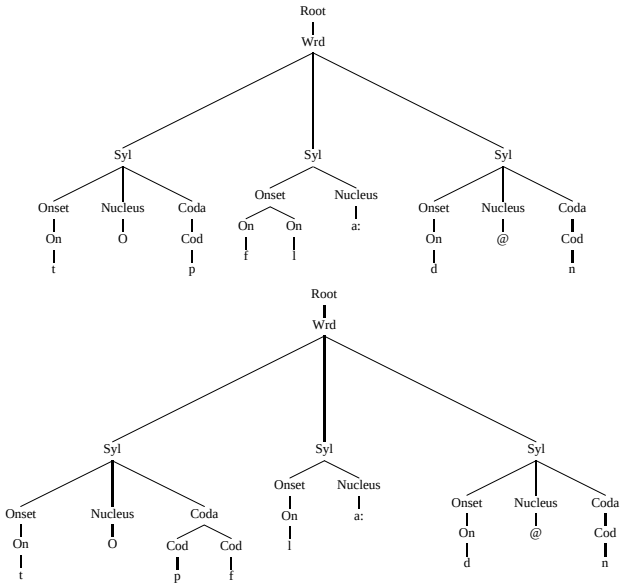


Figure 4: Syllable structure grammar

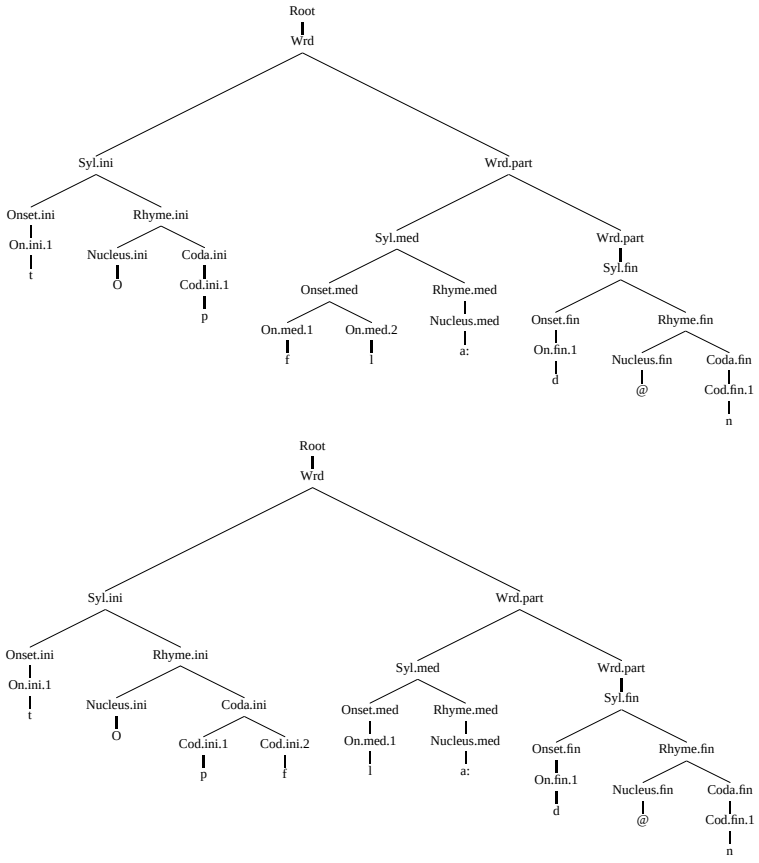


Figure 5: Positional syllable structure grammar

nary. For accuracy measurement, the raw phoneme strings of each word of the evaluation corpus are parsed with our various PCFGs, and the Viterbi parses are taken to extract the syllables of the phoneme strings. Then, the result is compared with the annotated variant in the evaluation corpus. Word accuracy is computed by counting the number of correctly syllabified words, and by dividing this number by the size of the evaluation corpus.

3.1. Evaluation Results

Figure 1 shows our evaluation results. Column 1 displays the series of grammars we investigated, and Column 2 displays the corresponding accuracy values.

The evaluation shows that the grammar with the richest

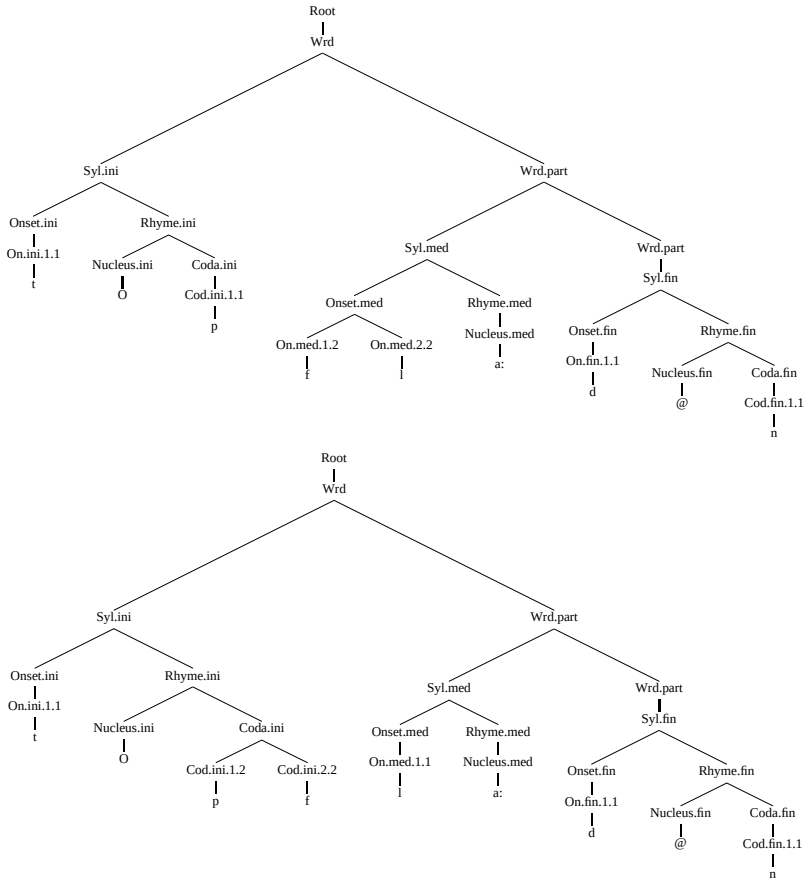


Figure 6: Advanced positional syllable structure grammar

grammar	word accuracy
phoneme grammar	62.37
treebank grammar	71.01
consonant-vowel grammar	93.31
syllable structure grammar	94.12
positional syllable structure grammar	96.42
advanced pos. syllable structure grammar	96.48

Table 1: Word accuracy of the probabilistic phonological grammars trained on a corpus of 182 000 words, and evaluated on a corpus of 242 000 words.

structure, the advanced positional syllable structure grammar, reaches the highest performance of 96.48% word accuracy for a training corpus size of 182 000 words. In general, the more linguistic knowledge is added to the grammar, the higher the accuracy of the grammar is. In contrast to the linguistic grammars, the results of the treebank grammar strongly depend on the size of the training corpus as reported in Müller (2001). They showed that even the simplest grammar, the phoneme grammar, was better than the treebank grammar until the treebank grammar was trained with a corpus size of 77 800. Of course, the low accuracy rates of the treebank grammar (trained on small corpora) were due to the high number of syllables that have not been seen in the training corpus.

3.2. Comparison to PARSEVAL

In this section, we want to exemplify that the problem of using PARSEVAL measures for this series of grammars is that an increase (or decrease) of the PARSEVAL measures can hardly be interpreted in terms of syllabification. In more detail, we show that it is simply unclear what it means for syllabification if two structurally varying grammars yield different values for "labeled precision".

The following example clarifies the problem why we choose an evaluation measure different from PARSEVAL. Let us suppose that

(i) the evaluation corpus consists of one single word, namely the above mentioned example word "Topfladen",

(ii) all six trained grammars predict the (wrong) syllable structure, Top-fladen ([tOp][fla:][d@n]) shown at the top of Figures 1-6,

(iii) the correct syllabification of Topfladen is annotated as Topf-laden ([tOpf][la:][d@n]) coded in six different treebanks shown at the bottom of Figures 1-6,

(iv) we evaluate our series of phonological grammars with the PARSEVAL measure "labeled precision".

Under these assumptions, all grammars fail in solving the syllabification task: all grammars yield a word accuracy of 0%, and a syllable accuracy of 33%. However, if the

grammars	analyses	PARSEVAL without preterminals	labeled precision
treebank grammar (Figure 1)	tree at the top tree at the bottom	**Wrd(0:9)** **Wrd(0:9)**	 1/1 = 100%
phoneme grammar (Figure 2)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl(0:3), Syl(3:6), **Syl(6:9)** **Wrd(0:9)**, Syl(0:4), Syl(4:6), **Syl(6:9)**	 2/4 = 50%
consonant-vowel grammar (Figure 3)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl(0:3), Syl(3:6), **Syl(6:9)** **Wrd(0:9)**, Syl(0:4), Syl(4:6), **Syl(6:9)**	 2/4 = 50%
syllable structure grammar (Figure 4)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl(0:3), Syl(3:6), **Syl(6:9)**, **Onset(0:1)**, Coda(2:3), Onset(3:5), **Onset(6:7)**, **Coda(8:9)** **Wrd(0:9)**, Syl(0:4), Syl(4:6), **Syl(6:9)**, **Onset(0:1)**, Coda(2:4), Onset(4:5), **Onset(6:7)**, **Coda(8:9)**,	 5/9 = 55.5%
positional syllable structure grammar (Figure 5)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl.ini(0:3), **Onset.ini(0:1)**, Rhyme.ini(1:3), Coda.ini(2:3), Wrd.part(3:9), Syl.med(3:6), Onset.med(3:5),**Rhyme.med(5:6)**, **Wrd.part(6:9)**, **Syl.fin(6:9)**, **Onset.fin(6:7)**, **Rhyme.fin(7:9)**, **Coda.fin(8:9)** **Wrd(0:9)**, Syl.ini(0:4), **Onset.ini(0:1)**, Rhyme.ini(1:4), Coda.ini(2:4), Wrd.part(4:9), Syl.med(4:6), Onset.med(4:5), **Rhyme.med(5:6)**, **Wrd.part(6:9)**, **Syl.fin(6:9)**, **Onset.fin(6:7)**, **Rhyme.fin(7:9)**, **Coda.fin(8:9)**	 8/14 = 57.1%
advanced positional syllable structure grammar (Figure 6)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl.ini(0:3), **Onset.ini(0:1)**, Rhyme.ini(1:3), Coda.ini(2:3), Wrd.part(3:9), Syl.med(3:6), Onset.med(3:5),**Rhyme.med(5:6)**, **Wrd.part(6:9)**, **Syl.fin(6:9)**, **Onset.fin(6:7)**, **Rhyme.fin(7:9)**, **Coda.fin(8:9)** **Wrd(0:9)**, Syl.ini(0:4), **Onset.ini(0:1)**, Rhyme.ini(1:4), Coda.ini(2:4), Wrd.part(4:9), Syl.med(4:6), Onset.med(4:5), **Rhyme.med(5:6)**, **Wrd.part(6:9)**, **Syl.fin(6:9)**, **Onset.fin(6:7)**, **Rhyme.fin(7:9)**, **Coda.fin(8:9)**	 8/14 = 57.1%

Table 2: PARSEVAL measure "labeled precision" (omitting preterminals) calculated on the basis of the examples shown in Figures 1-6

grammars	analyses	PARSEVAL with preterminals	labeled precision
treebank grammar (Figure 1)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl(0:3), Syl(3:6), **Syl(6:9)** **Wrd(0:9)**, Syl(0:4), Syl(4:6), **Syl(6:9)**	 2/4 = 50%
phoneme grammar (Figure 2)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl(0:3), Syl(3:6), **Syl(6:9)**,**P(0:1)**, **P(1:2)**, **P(2:3)**, **P(3:4)** **P(4:5)**, **P(5:6)**, **P(6:7)**, **P(7:8)**, **P(8:9)** **Wrd(0:9)**, Syl(0:4), Syl(4:6), **Syl(6:9)** **P(0:1)**, **P(1:2)**, **P(2:3)**, **P(3:4)** **P(4:5)**, **P(5:6)**, **P(6:7)**, **P(7:8)**, **P(8:9)**	 11/13 = 84.6%
consonant-vowel grammar (Figure 3)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl(0:3), Syl(3:6), **Syl(6:9)**,**C(0:1)**, **V(1:2)**, **C(2:3)**, **C(3:4)**, **C(4:5)**, **V(5:6)**, **C(6:7)**, **V(7:8)**, **C(8:9)** **Wrd(0:9)**, Syl(0:4), Syl(4:6), **Syl(6:9)**,**C(0:1)**, **V(1:2)**, **C(2:3)**, **C(3:4)** **C(4:5)**, **V(5:6)**, **C(6:7)**, **V(7:8)**, **C(8:9)**	 11/13 = 84.6%
syllable structure grammar (Figure 4)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl(0:3), Syl(3:6), **Syl(6:9)**, **Onset(0:1)**, **On(0:1)**, **Nucleus(1:2)** Coda(2:3), **Cod(2:3)**, Onset(3:5), On(3:4), **On(4:5)**, **Nucleus(5:6)**, **Onset(6:7)** **On(6:7)**, **Nucleus(7:8)**, **Coda(8:9)**, **Cod(8:9)** **Wrd(0:9)**, Syl(0:4), Syl(4:6), **Syl(6:9)**, **Onset(0:1)**, **On(0:1)**, **Nucleus(1:2)** Coda(2:4), **Cod(2:3)**, Cod(3:4), Onset(4:5), **On(4:5)**, **Nucleus(5:6)**, **Onset(6:7)** **On(6:7)**, **Nucleus(7:8)**, **Coda(8:9)**, **Cod(8:9)**	 13/18 = 72.2%
positional syllable structure grammar (Figure 5)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl.ini(0:3), **Onset.ini(0:1)**, **On.ini.1(0:1)**, Rhyme.ini(1:3), **Nucleus.ini(1:2)** Coda.ini(2:3), **Cod.ini.1(2:3)**, Wrd.part(3:9), Syl.med(3:6), Onset.med(3:5) On.med.1(3:4), On.med.2(4:5), **Rhyme.med(5:6)**, **Nucleus.med(5:6)**, **Wrd.part(6:9)** **Syl.fin(6:9)**, **Onset.fin(6:7)**, **On.fin.1(6:7)**, **Rhyme.fin(7:9)**, **Nucleus.fin(7:8)** **Coda.fin(8:9) Cod.fin.1(8:9)** **Wrd(0:9)**, Syl.ini(0:4), **Onset.ini(0:1)**, **On.ini.1(0:1)**, Rhyme.ini(1:4), **Nucleus.ini(1:2)** Coda.ini(2:4), **Cod.ini.1(2:3)**, Cod.ini.2(3:4), Wrd.part(4:9), Syl.med(4:6), Onset.med(4:5) On.med.1(4:5), **Rhyme.med(5:6)**, **Nucleus.med(5:6)**, **Wrd.part(6:9)**, **Syl.fin(6:9)** **Onset.fin(6:7)**, **On.fin.1(6:7)**, **Rhyme.fin(7:9)**, **Nucleus.fin(7:8)** **Coda.fin(8:9) Cod.fin.1(8:9)**	 15/23 = 65.2%
advanced positional syllable structure grammar (Figure 6)	tree at the top tree at the bottom	**Wrd(0:9)**, Syl.ini(0:3), **Onset.ini(0:1)**, **On.ini.1.1(0:1)**, Rhyme.ini(1:3) **Nucleus.ini(1:2)**, Coda.ini(2:3), Cod.ini.1.1(2:3), Wrd.part(3:9), Syl.med(3:6), Onset.med(3:5), On.med.1.2(3:4), On.med.2.2(4:5), **Rhyme.med(5:6)**, **Nucleus.med(5:6)** **Wrd.part(6:9)**, **Syl.fin(6:9)**, **Onset.fin(6:7)**, **On.fin.1.1(6:7)**, **Rhyme.fin(7:9)** **Nucleus.fin(7:8)**,**Coda.fin(8:9) Cod.fin.1.1(8:9)** **Wrd(0:9)**, Syl.ini(0:4), **Onset.ini(0:1)**, **On.ini.1.1(0:1)**, Rhyme.ini(1:4), **Nucleus.ini(1:2)**, Coda.ini(2:4), Cod.ini.1.2(2:3), Cod.ini.2.2(3:4), Wrd.part(4:9) Syl.med(4:6), Onset.med(4:5), On.med.1.1(4:5), **Rhyme.med(5:6)**, **Nucleus.med(5:6)**, **Wrd.part(6:9)**, **Syl.fin(6:9)**, **Onset.fin(6:7)**, **On.fin.1.1(6:7)**, **Rhyme.fin(7:9)**, **Nucleus.fin(7:8)**, **Coda.fin(8:9) Cod.fin.1.1(8:9)**	 14/23 = 60.9%

Table 3: PARSEVAL measure "labeled precision" (including preterminals) calculated on the basis of the examples shown in Figures 1-6

PARSEVAL measure "labeled precision" is expected to be useful for the syllabification task, then "labeled precision" should express that all grammars perform equally good (or bad) in our toy-setting.

Table 2 displays the results of "labeled precision". The matching brackets are shown in bold. Following the suggestion of Manning and Schütze (1999), the root node "Root" is not taken into account. Moreover, we omitted comparisons of pre-terminal nodes, since Manning and Schütze (1999) suggest to evaluate tagging and parsing separateley. In this evaluation, the simplest grammar, the treebank grammar, achieves the highest value for labeled precision (100%), since only the word-node is taken into account. The phoneme and the consonant-vowel grammar achieve the lowest values for labeled precision (50%). Table 3 also displays the results of "labeled precision", but here, we include the comparison of pre-terminals, due to the fact that we never applied our grammars using a

seperate tagger. Here, the treebank grammar achieves the lowest value for labeled precision (50%). The phoneme grammar, and the consonant-vowel grammar achieve the highest values for labeled precision (84.6%).

Thus, the results of both evaluations are hard to interpret, since they are inconsistent with one another. Furthermore, neither the first evaluation (omitting pre-terminals), nor the second evaluation (including pre-terminals) correspond to syllable accuracy, or word accuracy.

For these reasons, we doubt that the PARSEVAL measures are useful for evaluation of phonological grammars, at least for our grammars, which we developed for the syllabification task in mind. In contrast, we focus on evaluation of partial structures, namely on the category "SYL", and measure how good the grammars detect this single category on the word level. Interestingly, it seems that evaluating only a limited number of categories (here only a single category) is a harder evaluation measure than measuring the precision of all occurring substructures of a grammar.

3.3. Grammar Transformation: An Attempt to Map Word Accuracy to PARSEVAL

In this section, we discuss a grammar transformation enabling the measurement of word accuracy via PARSEVAL measures. In more detail, it could be suggested that the output of the phonological parser can be transformed to a tree, where all categories are removed except for the categories "Root", "SYL", and the terminals. However, if we follow this suggestion, there appear some problems. For the transformed grammar,

(i) the remaining category "SYL" is a pre-terminal node, which is usually NOT evaluated according to PARSEVAL; at least, if we follow Manning and Schütze (1999), who suggest to treat the tagging and parsing problem separateley.

(ii) all phonological information about syllable structure is lost, i.e., the syllabification problem is transformed to a tagging problem. However, we proved in recent work (Müller (2001), Müller (2002)) that it is advantageous to regard syllabification as a parsing problem.

(iii) although syllabification is a kind of segmentation, i.e., a one-dimensional process on a sequence of phonemes, we experienced, the more linguistic knowledge is added to the grammar, the higher the word accuracy of the grammar is. Thus, in our point of view, it is more adequate to model syllabification as a higher-dimensional process.

For these reasons, we prefer to use phonological enriched context-free grammars for stochastic inference and an evaluation focusing on partial structures most important for the particular task.

4. Conclusion

We presented an approach to supervised learning and automatic detection of syllable boundaries. In our experiments, we used a variety of grammars, which strongly differ in structure.

An evaluation using the standard measure for syllabification "word accuracy" shows that the grammar with the richest structures, the advanced positional syllable structure grammar, reaches the highest performance of 96.48% word accuracy for a training corpus size of 182 000 words. In general, the more linguistic knowledge is added to the grammar, the higher the word accuracy of the grammar is.

This evaluation, judging one single category of many grammars, is in strong contrast to PARSEVAL, which is designed for a single grammar evaluating (almost) all categories.

In a second evaluation using the original PARSEVAL measures on a toy-treebank, and a simple variant of the PARSEVAL measures, the results of both evaluations are hard to interpret, since they are inconsistent with one another. Furthermore, we found that neither the first evaluation (omitting pre-terminals), nor the second evaluation (including pre-terminals) correspond to syllable accuracy, or word accuracy.

Moreover, it turns out that evaluating only a limited number of categories (here only a single category) is a harder evaluation measure than measuring the precision of all occurring substructures of a grammar.

Lastly, we discussed a grammar transformation enabling the measurement of word accuracy via PARSEVAL measures. Here, it was necessary to reduce the syllabification problem to a tagging problem. However, we believe that it is advantageous to regard syllabification as a parsing problem.

For these reasons, future work will still use phonological enriched context-free grammars for stochastic inference and evaluations focusing on partial structures most important for the particular phonological task.

REFERENCES

Harald R. Baayen, Richard Piepenbrock, and H. van Rijn. 1993. The CELEX lexical database—Dutch, English, German. (Release 1)[CD-ROM]. Philadelphia, PA: Linguistic Data Consortium, Univ. Pennsylvania.

E. Black, S. Abney, D. Flickenger, C. Gdaniec, R. Grishman, P. Harrison, D. Hindle, R. Ingria, F. Jelinek, J. Klavans, M. Liberman, M. Marcus, S. Roukos, B. Santorini, and T. Strzalkowski. 1991. A Procedure for Qualitatively Comparing the Syntactic Coverage of English Grammars. In *Proceedings of Fourth DARPA Speech and Natural Language Workshop*.

Juliette Blevins. 1995. The Syllable in Phonological Theory. In John A. Goldsmith, editor, *Handbook of Phonological Theory*, pages 206–244, Blackwell, Cambridge MA.

Daniel Kahn. 1976. *Syllable-based Generalizations in English Phonology*. Ph.D. thesis, Massachusetts Institute of Technology, MIT.

Christopher Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA.

Karin Müller. 2001. Automatic Detection of Syllable Boundaries Combining the Advantages of Treebank and Bracketed Corpora Training. In *Proc. 39th Annual Meeting of the ACL*, Toulouse, France.

Karin Müller. 2002. Probabilistic Context-Free Grammars for Phonology. Submitted.

Helmut Schmid. 2000. LoPar. Design and Implementation. [http://www.ims.uni-stuttgart.de/projekte/gramotron/SOFTWARE/LoPar-en.html].

Richard Sproat, editor. 1998. *Multilingual Text-to-Speech Synthesis: The Bell Labs Approach*. Kluwer Academic, Dordrecht.

Jan P.H. Van Santen, Chilin Shih, Bernd Möbius, Evelyne Tzoukermann, and Michael Tanenblatt. 1997. Multilingual duration modeling. In *Proceedings of the European Conference on Speech Communication and Technology (Eurospeech)*, volume 5, pages 2651–2654, Rhodos, Greece.

Towards Comparing Parsers from Different Linguistic Frameworks An Information Theoretic Approach

Gabriele Musillo and Khalil Sima'an

Language and Inference Technology Group (LIT)
Institute for Logic, Language and Computation (ILLC)
University of Amsterdam
Nieuwe Achtergracht 166, 1018 WV Amsterdam, The Netherlands
musillo@science.uva.nl; khalil.simaan@hum.uva.nl

Abstract

Various efforts have been undertaken for developing methods for parser evaluation (Black et al., 1991; Lin, 1995; Carroll et al., 1996; Lin, 1998; Carroll et al., 1998; Carroll et al., 1999). These efforts concentrated on developing measures of parser performance, e.g. PARSEVAL (Black et al., 1991) labeled recall/precision for phrase-structure annotations. Different problems have been identified in the existing evaluation methods, but one of these problems strikes us as particularly challenging. The current benchmark for parser evaluation, Penn Wall Street Journal (WSJ) tree-bank (Marcus et al., 1993), cannot be used for the evaluation of parsers that are based on linguistic theories or annotation schemes that differ (*essentially*) from the annotation scheme found in this tree-bank. This problem can be restated as follows: how can parsers from different linguistic frameworks be *compared* in a quantitative and thorough manner? In this paper, we address this problem and suggest a new methodology for comparing parsers. The new methodology integrates Information Theoretic measures together with the PARSEVAL measures in a way that allows direct comparison of parsers that originate from different linguistic frameworks.

1. Introduction

In the last decade or so, a relatively large body of work in Computational Linguistics has been directed at the development and application of different parsing models for natural language processing e.g. (Brill, 1993; Magerman, 1993; Bod, 1995; Charniak, 1996; Eisner, 1996; Ratnaparkhi, 1997; Collins, 1997; Carroll et al., 1998; Lin, 1998; Chiang, 2000; Sima'an, 2000). Much work has been concentrating on how to extract stochastic models from existing tree-banks. One tree-bank in particular, the Penn Wall Street Journal tree-bank (Marcus et al., 1993) has received much attention in these efforts. Regardless of the reasons for this situation, this tree-bank has become a kind of bench mark for the evaluation and comparison of parsers. However, parsers that do not abide by the same linguistic framework as the WSJ tree-bank, or parsers for other languages than American English, are hard to compare to the mainstream which has been tested on the WSJ tree-bank. In this paper we address the issue of parser comparison across different linguistic frameworks. This problem is interesting as it touches on linguistic issues, but also on the question "how to view the role of language structure". In this preliminary report on our research into this question, we assume that the parsers that are being compared are built for the same language. We also assume the existence of a corpus of utterances, i.e. that parser comparison takes place on specific domains of language use. Furthermore, in our evaluation here we do not take into consideration the important aspect of parser efficiency.

The structure of this paper is as follows. Section 2 discusses the linguistic aspects of how to compare parsers that originate from different linguistic frameworks. Section 3 argues that parser comparison needs more thought than parser evaluation, exactly because the scores need to be compared somehow. Section 4 we extend PARSEVAL with an Information Theoretic methodology, which allows parser comparison across different linguistic frameworks. Finally, section 5 concludes this paper.

2. How to compare parsers?

Remarkably, current parser evaluation practice seems to have been limited to a single benchmark tree-bank (Penn WSJ tree-bank). The evaluation of newly developed parsers proceeds by testing on a held-out portion of this tree-bank. When the parsers are acquired from the tree-bank itself, evaluation (and comparison) of the parsers is based on the PARSEVAL measures (Black et al., 1991). However, when a new parser is devised and this parser employs a different linguistic framework than the annotation scheme of the Penn WSJ tree-bank (e.g. dependency grammar), a serious problem arises. The problem lies in how to relate the different syntactic schemes to one another. There seem to be two related ideas on how to address this problem: (1) devising a general, syntactic scheme, a kind of "common ground", which serves as an "interlingua", or (2) devising mappings between each pair of syntactic schemes. In this section we argue that both suggestions seem not be workable in practice. We review the more popular among the two, i.e. the methods of devising mappings from the WSJ annotation scheme to other schemes. Subsequently we suggest that it is more expedient to develop different tree-banks of the same corpus of utterances, each in a different syntactic annotation scheme.

2.1. Common syntactic scheme?

At first glance, the problem of specifying the syntactic "common ground" seems to rely on the choice of some general linguistic framework to which both parsers' outputs can be mapped in order to compare them. However, selecting a common linguistic framework seems a hopeless task:

the problem lies in anticipating the kinds of linguistic information that a new linguistic theory might be interested in. Take for example the *Minipar* parser (Lin, 1998) or the *Link parser* (Sleator D, 1991), the first outputs one type of dependencies, while the second outputs so called "links". Both parsers do not exactly coincide with the traditional grammatical relations used in other frameworks (or with the WSJ annotation). How this kind of syntactic information would be anticipated by a common framework is not completely clear. We believe that a common framework will always be contested as being more favorable to some parser than another. Moreover, it seems to us that the goal of devising such a framework coincides with the ultimate goal, which has evaded the syntactic linguistic work for so many years now. This might be inherent to deciding on the so called "borders" of syntax, which seems to overlap with morphology on the one side, and with semantics and pragmatics on the other. It is highly doubtful that such a framework can be developed.

2.2. Mappings between syntactic schemes

Various researchers (Lin, 1995; Carroll et al., 1998) have developed methods that attempt at transforming the the Penn WSJ format into the different output formats of their parsers. However, there is in these efforts a hidden assumption: a complete mapping can be constructed, which maps a WSJ parse-tree into a parse-tree in any of these schemes. Apart from the linguistic arguments that exist against the "relatively shallow" WSJ style of annotation, there is a serious problem in assuming the existence of a complete, possibly deterministic mapping.

The arguments against this practice emerge from linguistic frameworks (e.g. dependency grammar) that differ to a large extent from the framework employed in the WSJ tree-bank. The main arguments address the risks that accompany mapping parse-trees from one framework to another. When a parse-tree is mapped from one framework to another, one might

- risk losing linguistic information (e.g. some dependencies not found in the WSJ),
- face ambiguity, since the categories provided by the parser might map onto different WSJ categories (possibly dependent on context).

These problems suggest that the comparison of two parsers can not rely on mapping the output onto some pre-selected linguistic framework, since different linguistic frameworks address different syntactic aspects. Nevertheless, we find in the literature various empirical efforts aimed at devising such mappings, notably (Lin, 1995; Carroll et al., 1998). Next we first address what it takes to devise a mapping from Phrase-Structure to Dependency grammar and vice versa. Then we shortly discuss some of the problems that exist in the mappings devised by (Lin, 1995; Carroll et al., 1998).

2.3. What is necessary for a mapping?

We concentrate on two popular linguistic frameworks of syntax: dependency and phrase-structure. Although this could be an illusive task, we give here a simplified description of the two, each in a single line. Phrase-structure syntax allows describing the syntactic structure of utterances in terms of the phrases which constitute them, using a hierarchical and recursive set of concepts. In contrast, dependency syntax aims at making explicit the dependencies between the pairs of words in the utterance. Both approaches aim at facilitating the discovery of argument-structure, which is often assumed by subsequent semantic processing.

It has been suggested by (Hudson, 1984; Covington, 1992; Covington, 1994) that, according to a phrase-structure grammar, constituency is basic and dependency (or government) is derived, whereas according to dependency grammar, dependency is basic and constituency is derived. If this claim is correct, then a transformation procedure and its inverse that map phrase-structure parses to dependency parses could be defined; in that case, phrase-structure and dependency grammars are to a large extent isomorphic.

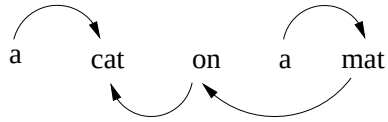


Figure 1: A dependency parse

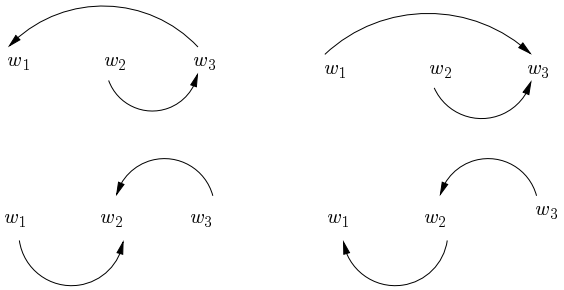


Figure 3: Different possible dependency trees

The problem lies, of course, in the kind of concepts that each framework presupposes: the types of dependencies and the constituent types must refer to the same abstract syntactic concepts, which is usually not the case. In order to shed some light on the problematic aspects of such a mapping, consider the expression *a cat on a mat* and the plausible dependency parse for it shown in figure 1; at least three distinct phrase-structure parses may be projected from it as shown in figure 2. Clearly, there is here a problem of how to decide on the single correct phrase-structure parse, given the dependency structure. The reverse mapping can also be problematic: consider the following unlabeled bracketing $(w1(w2w3))$ of an expression $w1w2w3$. At least four dependency parses can be generated from it as shown in figure 3. Again, a principled choice of the single correct dependency parse is not easy and demands a procedure for recognizing the head word of each phrasal category. The problem, however, in devising head word

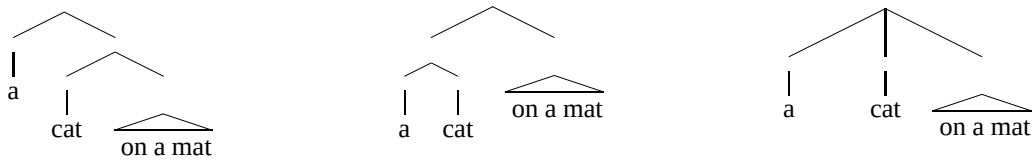


Figure 2: Three different phrase-structure trees for same dependency structure

recognition procedures for an existing tree-bank, has been exemplified by the various versions of the head recognition procedure developed by for the WSJ tree-bank (Magerman, 1993; Collins, 1997; Buchholz et al., 1999; Eisner, 2001). In any case, it seems that the problems that accompany these mappings can be summarized in two elements (1) a common set of concepts that underlie the types in each of the two frameworks, and (2) a clear and well founded definition of a head recognition procedure. Let us consider two attempts at developing such mappings.

2.4. Lin's proposal

In (Lin, 1995), a dependency parse of a sentence is defined as a set of tuples. Such dependency tuples consist of 5 components: a *dependent*, a *PoS*, a *position*, a *head* and a *type*. This last component is optional. Lin defines the values that can be assigned to these components as follows: a word in a sentence to be parsed is assigned to the *dependent* variable, *PoS* represents its lexical category, the head word on which the value of *dependent* depends is assigned to the *head* variable, the *position* takes a value in the set $\{<, >, <<, >>, <<<, ..., *, ?\}$. Remarkably, no well-defined set of values is defined for the optional component *type*. Furthermore, Lin gives no hints at how to label *head-dependent* relations.

Lin presents an algorithm to transform a constituency tree into a dependency tree. His transformation procedure exploits suggestions made in (Magerman, 1993) for determining *lexical representatives* of phrases. Whether the notion of *lexical representative* coincides with the notion of *head* as used in dependency grammar, is not clear. Suppose, for the sake of the argument, that *lexical representatives* are *heads* and consider the *wh*-interrogative, parsed according to the bracketing guidelines for the Penn Tree Bank as shown at the left side of figure 4. Let us apply Magerman's rules to it. According to these rules, the head-word of *SBARQ* is the head-word of *SQ*, that is *propose*. Therefore, the head-word of *which measures* is a dependent of *propose*. However, consider the *wh*-interrogative on the right hand side of figure 4. The head-word of *SBARQ* is *think*. Therefore, the *WHNP which measures* is not any more a dependent of *proposed*, it is a dependent of the head word *think*. This implies that transforming standard phrase structure analysis into some dependency representation in this way results in loss of information. Such information represented by the position of a trace is of course relevant to semantic interpretation. Our examples clearly show that such a transformation procedure fails to detect a dependency that relates a dependent to a "lower head" (one that does not percolate across the constituent boundaries).

2.5. Carroll et. al's proposal

Carroll et al. (Carroll et al., 1998) has proposed some relational evaluation measures that exhibit some resemblance to Lin's. They describe a corpus annotation scheme that encodes grammatical relations between heads and dependents. We believe that Carroll's proposal is somehow superior to Lin's in a few aspects. Firstly, the set of dependency types or grammatical relations is well-defined and constitutes a hierarchy. This allows robust and shallow evaluation. Secondly, grammatical relations are strictly speaking not dependency relations; the external argument of a "subject control verb" is grammatically related to the control verb and to the controlled verb (e.g. *I promise to come*, where *I* is related to both *promise* and *come*). Finally, grammatical relations are specified even for moved phrases that do not occur in a canonical position. This addresses a problem in Lin's proposal, mentioned above.

From the experience described in (Carroll et al., 1998), it might seem that dependency types (or grammatical relations) are easy to specify and extract from phrase-structure. Nevertheless, this is true only because (Carroll et al., 1998) assumes that the phrase-structure grammar is an explicit, determinate set of rules. As Carroll et al. recognize, extracting grammatical relations from an implicit grammar, induced automatically from a tree-bank, is much harder to do consistently. In addition, the grammatical relations in (Carroll et al., 1998) do not capture some relevant information. For instance, topicalized constituents of the Penn Tree Bank (bearing the TPC tag) are ignored, because they are allegedly difficult to specify under which conditions a constituent is topicalized.

Clearly, from these examples we observe that developing deterministic, complete mappings between phrase-structure and dependency grammars is a tedious and risky task. For a nice review of the problems that arise in relating Dependency to Phrase-Structure syntax see (Schneider, 1998). We believe that the development of different tree-banks, each in another linguistic annotation scheme, for the same corpus of utterances might provide a more fruitful path to proceed. When a pair of parallel tree-banks exits, it is possible to explore automatic means for learning complex, stochastic mappings between the two. More importantly, a pair of parallel tree-banks for the same corpus of utterances may serve as a suitable infrastructure for the comparison of parsers from different linguistic frameworks as we describe in the rest of this paper.

3. Comparison: more than evaluation

In line with current practice, we believe that empirical parser evaluation requires a manually constructed, gold-

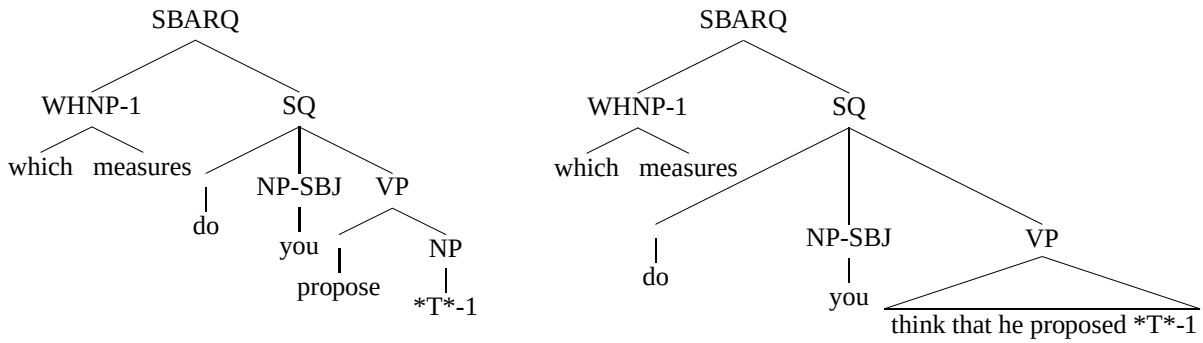


Figure 4: Two Penn WSJ style parse-trees

standard tree-bank and suitable measures of the "similarity" between the analyses output by the parser and the corresponding analyses that are found in the tree-bank. Usually, the measure of "analyses similarity" consists of different figures pertaining to coverage (or recall) and accuracy (or precision). In contrast, the task of *comparing* parsers can be more complex than the evaluation of a single parser (or the comparison of parsers that share the same output scheme). When two parsers are being compared, another major issue, beside evaluation, must be addressed: how to compare the similarity measures across different kinds of parser outputs (possibly originating from different linguistic frameworks)? We believe that the latter question is of theoretical importance and we address it in this section. First, however, we need to discuss the multiple possibilities for parser comparison and provide the argumentation that underlies the specific choices that we make.

3.1. Task-oriented comparisons

Initially, we distinguish between two goals of parser comparison: (1) the suitability of the parser to a given task, and (2) the suitability of the parser as a model of syntactic language processing. Although (ideally) the two goals are strongly related, in practice they might imply different comparison methodologies. The comparison of parser's suitability for a specific task is usually guided by some detailed specification of the requirements which the parser must meet. For example, the simplified Question-Answering task requires the parser to output (as fast as possible) the main predicate-argument structure of the input. In contrast, more complex tasks, such as the task of Machine Translation, will possibly require a much more detailed syntactic analysis of the input. Task-oriented comparison is interesting and useful, but is strongly specific to the task at hand, which means that it does not constitute a general comparison methodology.

3.2. Qualitative comparisons

When the parser comparison is not tied to a specific task[1], parser comparison is aimed at investigating the utility of the different models underlying the parsers. Clearly, a *qualitative* comparison, based on theoretical considerations of coverage of language phenomena (e.g. (Carroll and Weir, 1997)), could be illuminating. Issues such as "what phenomena the parser can be expected to cover" and "what quality of the output is provided by the parser" are important for advancing the state of the art. However, qualitative comparisons become more powerful when they are supplemented with quantitative comparisons that are based on actual empirical evidence (weighted according to expected frequency of occurrence). This is because the theoretical investigations might pay too much attention to relatively infrequent phenomena and less attention to frequent (yet seemingly irrelevant) phenomena. Empirical, quantitative parser comparison aims at providing an answer to the question: what quality of output is provided by the parser and how does it compare to other parsers? Before we address this question, however, we address a related idea which is currently being floated as an alternative for (full) parser comparison: partial evaluation.

[1]This is currently the case in parser evaluations on the Wall Street Journal corpus, for example.

3.3. Partial evaluation based comparisons

Because of the problematic mappings between the different linguistic frameworks, it seems suitable to consider only some of the issues upon which these frameworks agree. For example, one could conduct comparisons on the main predicate-argument structure of the input, or on recall/precision for the set of predicate-argument structures for the verbs in the input utterance. Similar suggestions have been made in order to contrast the so-called "shallow" parsers to existing "full" parsers, by e.g. listing the recall/precision on each phrasal category (or kind of dependency) separately (Tjong Kim Sang and Déjean, 2001). These suggestions for *partial evaluation* usually provide a detailed and informative listing of various aspects of the parser's behavior. We think that these should be taken more seriously in *practical parser evaluation*. It is important to have detailed lists of scores of a parser on different tasks. However, partial evaluation has its limitations, even for practical comparisons. It is very hard to predict what elements in the output of a given parser could be important. For example, predicate-argument structures, which take only verbal predicates into consideration, are useless for some applications where prepositional phrases are important (e.g. domains of travel information, or money transactions etc). Another weakness of partial evaluation is that, by definition, it does not answer the need for a "bottom line figure" which summarizes the behavior of the parser, and allows direct comparison to other parsers.

There is, moreover, a more urgent matter, which par-

tial evaluation does not address, and which is of theoretical importance. This concerns the question: how much information[2] does the parser return, and what is it's quality? Answering this theoretical question is important for advancing the state of the art in natural language processing. In the light of the current divergence in parser output formats (e.g. shallow vs. deep parsers) and given the differences between the linguistic frameworks, it becomes important to be able to measure differences in the informativeness of parsers, even when their outputs are not directly comparable.

3.4. Comparison on parallel tree-banks

As argued in Section 2., for comparing parsers that come from different linguistic frameworks (or different depths of analysis) it is necessary to maintain some kind of mapping between the outputs of the parsers. The mapping might be realized in one of two manners:

Explicit: a tree-bank exists in one annotation scheme (according to some linguistic framework) together with a sound, complete and correct mapping which translates every analysis in the tree-bank into the corresponding analysis in the other linguistic framework,

Implicit: two parallel tree-banks[3] of the same corpus of utterances, each annotated according to one of the linguistic frameworks.

We already showed in Section 2., that developing an explicit deterministic mapping seems a hard task. Therefore, we advocate the use of implicit mappings that are embodied in parallel tree-banks of the same corpus of utterances. It is evident that given such pairs of tree-banks, different automatic methods can be explored for learning complex, stochastic mappings between the two tree-banks. How to acquire these mappings is an interesting subject of research but is beyond the scope of this paper.

4. An Information Theoretic proposal for parser comparison

Suppose we are given two parsers P_1 and P_2 which have different output schemes, respectively, L_1 and L_2. Suppose also we are given a corpus of utterances C, and tree-banks TB_1 and TB_2 that are annotated versions of C according to schemes L_1 and L_2, respectively. In order to ground the discussion, the reader might want to imagine that L_1 is dependency grammar (Mel'čuk, 1988) and L_2 is phrase structure grammar (Manning and Schutze, 1999); or alternatively, L_1 could be the scheme output by a shallow parser and L_2 a "deeper" linguistic scheme. The question is, how do we compare P_1 and P_2 in a way that takes into consideration not only the coverage/accuracy but also the informativeness of their output? Below we discuss the two issues of coverage/accuracy and informativeness separately. Subsequently we propose combined measures which allow comparison.

[2]This is opposed to the practical question: how much information can the parser provide for this or that task?

[3]We will keep the term tree-bank when we refer to a bag of utterance-analysis pairs, where the analyses are syntactic structures according to some linguistic framework, e.g. dependency grammar or phrase-structure grammar.

4.1. Coverage/accuracy: generalizing PARSEVAL

The PARSEVAL measures of labeled constituent recall and precision (Black et al., 1991) have been central in current efforts at parser evaluation on the current American English language benchmark (Penn WSJ tree-bank). It has often been claimed that these measures are not suitable for evaluating e.g. dependency syntax. Indeed, when taken literally, constituency can be meaningless when evaluating dependency syntax. However, the PARSEVAL measures can easily be generalized to deal with whatever kind of parses as long as they can be represented as sets of relations. For example, a labeled constituent $\langle i, j, XP\rangle$ is a relation (where i and j are the positions of the left-most and right-most words respectively and XP is the label of the constituent); a labeled dependency $\langle h, d, L\rangle$ is a relation (where h and d are the positions of the head-word and the dependent, and L is the label of the dependency). A parse-tree, whether in dependency syntax or in phrase-structure, can be represented as a set of such relations (cf. (Goodman, 1998)). Recall and precision, as a direct generalization of the notation used in PARSEVAL, are defined as measures over sets of such relations. If a given parser outputs parse T for sentence U for which the gold standard parse is G, Goodman defines[4]:

$Recall(T,G) = \frac{|G \cap T|}{|G|}$ $Precision(T,G) = \frac{|G \cap T|}{|T|}$

We believe that the PARSEVAL measures can be generalized further to stricter recall/precision measures, where a parse-tree is viewed as a set of relations that range over different aspects of syntax, e.g. relations in which the labeled constituent is head-lexicalized, and possibly supplemented with the set of subcategorization frames of its head word. The notions of recall/precision over sets of relations are general enough to accommodate a wide range of aspects of parse-trees, including suggestions for partial evaluation, e.g. on the basis of predicate-argument structures of verbs.

Hence, the parsers P_1 and P_2, assumed earlier, can be evaluated on their own tree-banks TB_1 and TB_2 using the PARSEVAL recall/precision measures. However, the PARSEVEAL measures of recall/precision do not address the problem of comparison across different output/annotation schemes. To arrive at a suitable comparison methodology we first need to define measures of the informativeness of the output of a given parser.

4.2. Informativeness of a parser

What makes a parser informative? To answer this we turn to the Information Theoretic concept of compression. Suppose we are given two parsers. The one parser outputs only unlabeled bracketed parse-trees, while the other labels the same parse-trees with different syntactic categories. The output of the second parser can be described as more informative. As it turns out, the kind of concepts, e.g. phrasal categories or dependency types, which the parser includes in its output determine its informativeness. For example, a parser that marks the difference between singular/plural noun and verb phrases could be more informative than another that does not do so (all else being equal, of

[4]If $(|T| == 0)$ then $Precision(T,G) = 0$ by definition.

course).

In general, a *linguistic concept* is viewed as a set of word sequences, e.g. the noun-phrase concept consists of a sequence of all noun constituents. Here, we take a slightly different perspective on this notion: a concept is a probability distribution over word sequences (Manning and Schutze, 1999). In a generative grammar, a finite set of concepts is specified hierarchically (and recursively). If the concepts are "stricter" or sharper they will tend to be more informative, provided that the strictness captures regularities in the language. This sense of an "informative annotation scheme" is strongly related to the notion of a "compression code" in the communication over a noisy channel view in Information Theory. The more the annotation scheme allows to compress a large corpus of utterances from the language, the more informative this annotation scheme is.

4.3. Cross entropy of an annotation scheme

Technically speaking, in language modeling techniques that originate from the speech community, the "goodness" of a model is captured through the notion of Perplexity of the model on a corpus of utterances. The strongly related notion of *Cross Entropy* is also known from the statistical parsing literature, e.g. (Manning and Schutze, 1999); it captures the average amount of surprise that the model encounters when parsing the utterances in the corpus. A model that captures the regularities in the corpus in a better way, through more adequate syntactic constructs and concepts, will encounter less surprises. How do we apply this idea to parser comparison where we want to measure the informativeness of an annotation scheme (the output of the parser)?

We stress that we would like to compare the outputs of the parsers, rather than their ambiguity resolution capacity[5]. To do so, we suggest a method for measuring a kind of "cross entropy" between each of the tree-banks TB_i and the corpus of utterances C. If the tree-bank parse-trees capture the regularities in the corpus utterances in a better way, the cross entropy will be smaller. But, how do we define this "cross entropy between a tree-bank (rather than a model) and a corpus"?

Although it is not a trivial task, we believe that every tree-bank annotation scheme, whether phrase-structure, head-lexicalized phrase-structure or dependency structure, allows the extraction of a probabilistic model which we will call the "basic generative model". The "basic generative model" must fulfill the following requirements:

1. the rewrite rules of this model must coincide with the atomic units assumed by the linguistic framework, and

2. only the information that exactly coincides with *the logical constraints on the composition] operators*[6] *that originate from the linguistic framework should be available as conditioning context for the model parameters*.

For example, for (context-free) phrase-structure grammars, the logical constraint on the substitution operator is *category substitutability*, which implies that the conditioning context in context-free rule probabilities consists of the label of the left-hand side of the rule, as in standard Probabilistic Context-Free Grammars. Accordingly, it is not suitable to condition the probabilities of the extracted basic model on e.g. the label of the parent node[7]. In effect, beyond the necessary conditions, the basic generative model assumes independence between the different basic rewrite units $r_1 \cdots, r_n$ that generate a parse-tree T, i.e. $P(T) = \prod_{i=1}^{n} P(r_i|\phi(r_i))$, where $P(r_i|\phi(r_i))$ is the relative frequency of r_i in the tree-bank, conditioned on the necessary information only.

Let us consider a few example frameworks from the literature. For phrase-structure annotations, as we just said, the basic model is a Probabilistic Context Free Grammar (PCFG) (Jelinek et al., 1990) (the so-called Tree-bank Grammar (Charniak, 1996)); for a dependency syntax tree-bank, this is a probabilistic generative model in which the dependency probabilities are conditioned on the head-words (see e.g. model 3 of (Eisner, 1996) without conditioning on the preceding child, i.e. 0^{th}-order Markov model for generation of dependents); for a head-lexicalized tree-bank, where head words augment the phrasal nonterminals, the basic model is similar to model 1 of (Collins, 1996) (simplified to exclude "distance measures").

Having extracted a basic generative model from the tree-bank, it becomes easy to specify how the measure of Cross Entropy between the model and the corpus can be estimated. Let model μ be a probabilistic generative model and let C be a corpus of utterances. The cross-entropy is defined by

$$H_\mu(C) = \lim_{|C|\to\infty} \frac{1}{|C|} \sum_{U \in C} \log P(U|\mu)$$

where $|C|$ is the number of utterances U in C. When C is a large corpus, it is possible to estimate this by dropping the limit:

$$\hat{H}_\mu(C) = \frac{1}{|C|} \sum_{U \in C} \log P(U|\mu)$$

Note that for our goal of comparison, it is enough to estimate only roughly the cross entropy on a reasonably large corpus (clearly, the larger the better).

4.4. Per bit recall/precision

For integrating the measures of disambiguation and informativeness of a parser, we will argue for the notion of "per bit disambiguation capacity". This new notion addresses the question: how good is the quality of the output of the parser given its informativeness? This notion is obtained by integrating the PARSEVAL Precision/Recall measures of disambiguation with the Cross Entropy of the annotation scheme (which is the scheme used in the output

[5]The latter issue has been addressed in the recall/precision aspect of the evaluation methodology suggested here.

[6]The composition operators that are used for the construction of parse-trees from the basic rewrite rules.

[7]Note that we say this is not suitable only for the goal of measuring the informativeness of the output of the parser, not for the ability of the parser to disambiguate.

of the parser):

$$Per\ bit\ Recall = \frac{Recall}{CrossEntropy}$$

$$Per\ bit\ Precision = \frac{Precision}{CrossEntropy}$$

The intuition underlying these notions is that frameworks that leave the parses more ambiguous, will demand less effort during parsing and so recall/precision must be discounted accordingly. We claim that the new measures of per bit recall and precision for different parsers can be compared directly, even when the parsers originate from different linguistic frameworks.

4.5. Discussion

We note that there exist various notions that are strongly related to Cross Entropy as a measure of model goodness. One of these notions is the *description length* (or the theoretical Kolmogorov complexity), another is the *message length* (Rissanen, 1983). However, the Cross Entropy measure is the most directly applicable among these closely related notions because of its direct interpretation in terms of smoothed relative frequency.

We expect two issues to constitute the critical points in the application of the methodology proposed in this paper: (1) the development of parallel tree-banks for the same corpus of utterances, (2) the benchmarking of methods for the extraction of basic probabilistic models from these tree-banks. The first issue is critical because it is labor intensive; the second because it demands further specification of what constitutes a basic model that can be extracted from a newly developed tree-bank, especially when this concerns simplified frameworks such as those used by shallow parsers. Despite of these possible difficulties, we believe that our proposal could provide a theoretical departure point towards workable approximations.

5. Conclusions

We presented a preliminary, informal discussion of what it takes to develop a methodology for parser comparison across different linguistic frameworks. We have presented a simple Information Theoretic approach to avoid the problems that arise in mapping between different linguistic frameworks. This approach takes into account the specific concepts of the framework, circumventing the problem of information loss.

As a positive side to this proposal, we envision that for a given domain of language use there will be different tree-banks, each according to a different linguistic framework. This will allow the development of automatic approximations for domain-specific mappings between the different frameworks (e.g. using Machine Learning techniques). Furthermore, this also enables the exploration of complementary aspects of the different existing linguistic frameworks, possibly leading to better stochastic parsers.

Future work in this direction might concentrate on evaluating existing parsers from dependency and phrase-structure grammar, and comparing them using the present approach. Another line of work might concentrate on the application of Machine Learning or stochastic methods to the induction of approximate mappings between these different frameworks. Finally, we are intrigued by the possibility of empirical studies that combine aspects from different frameworks based on parallel tree-banks.

Acknowledgments: We thank Maarten de Rijke and Gabriel Infante-Lopez for discussion and comments on an earlier version. This research was supported by a grant from the Netherlands Organization for Scientific Research (NWO), under project number 220-80-001.

6. References

E. Black et al. 1991. A procedure for Quantitatively Comparing the Syntactic Coverage of English Grammars. In *Proceedings of the February 1991 DARPA Speech and Natural Language Workshop*.

R. Bod. 1995. *Enriching Linguistics with Statistics: Performance models of Natural Language*. PhD thesis, ILLC-dissertation series 1995-14, University of Amsterdam.

E. Brill. 1993. *Transformation-Based Learning*. Phd Thesis , University of Pennsylvania.

S. Buchholz, J. Veenstra, and W. Daelemans. 1999. Cascaded grammatical relation assignment. In *In Proceedings of EMNLP/VLC-99*, pages 239–246.

J. Carroll and D. Weir. 1997. Encoding frequency information in lexicalized grammars. In *In ACL/SIGPARSE workshop on Parsing Technologies (IWPT), MIT, Cambridge. Also to appear in Data Oriented Parsing, R. Bod, R. Scha and K. Sima'an (editors), CSLI publications, 2002.*

J. Carroll, T. Briscoe, N. Calzolari, S. Federici, S. Montemagni, V. Pirrelli, G. Grefenstette, A. Sanfilippo, G. Carroll, and M. Rooth. 1996. Sparkle poject. http://www.ilc.pi.cnr.it/sparkle/wp1-prefinal/node10.html.

J. Carroll, T. Briscoe, and A. Sanfilippo. 1998. Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, pages 447–454, Granada, Spain.

J. Carroll, G. Minnen, and T. Briscoe. 1999. Corpus annotation for parser evaluation. In *Poroceedings of the EACL-99 Post-conference Workshop on Linguistically Interpreted Corpora*, pages 35–41, Bergen, Norway.

E. Charniak. 1996. Tree-bank Grammars. In *Proceedings AAAI'96*, Portland, Oregon.

D. Chiang. 2000. Statistical parsing with an automatically-extracted tree adjoining grammar. In *Proceedings of the 38^{th} Annual Meeting of the Association for Computational Linguistics (ACL'00)*, pages 456–463, Hong Kong, China.

M. Collins. 1996. A new statistical parser based on bigram lexical dependencies. In *Proceedings of the 34th Annual Meeting of the ACL*, pages 184–191.

M. Collins. 1997. Three generative, lexicalized models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the ACL and the 8th Conference of the EACL*, pages 16–23, Madrid, Spain.

Michael A. Covington. 1992. GB Theory as Dependency Grammar. Technical Report AI-1992-03, Athens, GA.

Michael A. Covington. 1994. An empirically motivated reinterpretation of dependency grammar. Technical Report AI-1994-01, Athens, GA.

J. Eisner. 1996. Three new probabilistic models for dependency parsing: An exploration. In *Proceedings of COLING-96*, pages 340–245, Copenhagen, Denmark.

J. Eisner. 2001. Smoothing a probabilistic lexicon via syntactic transformations. PhD thesis, Department of Computer Science, UPenn.

J.T. Goodman. 1998. *Parsing Inside-Out*. PhD thesis, Departement of Computer Science, Harvard University, Cambridge, Massachusetts, May.

R. Hudson. 1984. *Word Grammar*. Basil Blackwell.

F. Jelinek, J.D. Lafferty, and R.L. Mercer. 1990. *Basic Methods of Probabilistic Context Free Grammars, Technical Report IBM RC 16374 (#72684)*. Yorktown Heights.

Dekang Lin. 1995. A dependency-based method for evaluating broad-coverage parsers. In *Proceedings of IJCAI-95*.

Dekang Lin. 1998. Dependency-based evaluation of minipar. In *Workshop on the Evaluation of Parsing Systems*, Granada, Spain, May.

D. Magerman. 1993. Parsing as statistical pattern recognition. PhD Thesis, Stanford University.

C. Manning and H. Schutze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA.

M. Marcus, B. Santorini, and M. Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn treebank. *Computational Linguistics*, 19:313–330.

I.A. Mel'čuk. 1988. *Dependency Syntax: Theory and Practice*. State University of New York Press.

A. Ratnaparkhi. 1997. A linear observed time statistical parser based on maximum entropy models. In *Proceedings of Empirical Methods in NLP, EMNLP-2*, pages 1–10.

J. Rissanen. 1983. A universal prior for integers and estimation by minimum description length. *The Annuals of Statistics*, 11 (2):416–431.

G. Schneider. 1998. A linguistic comparison of constitency, dependency and link grammar. Master thesis, Institut fur Informatik, Universitat Zurich.

K. Sima'an. 2000. Tree-gram Parsing: Lexical Dependencies and Structual Relations. In *Proceedings of the* 38^{th} *Annual Meeting of the Association for Computational Linguistics (ACL'00)*, pages 53–60, Hong Kong, China.

Temperley D Sleator D. 1991. Parsing english with a link grammar. Technical report, Carnegie Mellon University Computer Science.

Erik F. Tjong Kim Sang and Hervé Déjean. 2001. Introduction to the conll-2001 shared task: Clause identification. In *Proceedings of CoNLL-2001*, pages 53–57. Toulouse, France.

Evaluation of the Gramotron Parser for German

Franz Beil[1], Detlef Prescher[2], Helmut Schmid[3], Sabine Schulte im Walde[3]

[1]TEMIS, Rue de Ponthieu 59, 75008 Paris, France
franz.beil@temis-group.com

[2]DFKI GmbH, Language Technology Lab, Stuhlsatzenhausweg 3, 66123 Saarbrücken, Germany
prescher@dfki.de

[3]IMS, Universität Stuttgart, Azenbergstr. 12, 70174 Stuttgart, Germany.
{Helmut.Schmid,Sabine.Schulte.im.Walde}@IMS.Uni-Stuttgart.DE

Abstract

The paper describes an experiment in inside-outside estimation of a lexicalized probabilistic context free grammar for German. Grammar and formalism features which make the experiment feasible are described. Successive models are evaluated on precision and recall of phrase markup consisting of labels for noun chunks and subcategorization frames. Our approach to parsing is a blend of symbolic and stochastic methods where we use evaluation results in both incremental grammar development and validation of selected output to be used in lexical semantic clustering. Our results are that (i) scrambling-style free phrase order, case morphology, subcategorization, and NP-internal gender, number and case agreement can be dealt within a lexicalized probabilistic context-free grammar formalism, and (ii) inside-outside estimation appears to be beneficial, however relies on a carefully built grammar and an evaluation based on carefully selected linguistic criteria. Additionally, we report experiments on overtraining with inside-outside estimation, especially focusing on comparison of the results of mathematical and linguistic evaluations.

1. Introduction

From 1997 to 2000, the Gramotron group of the Institute for Natural Language Processing at Stuttgart University developed a stochastic parser for German (Beil et al. (1999), Schulte im Walde et al. (2001)). The symbolic component of the final parsing system is a manually written context-free grammar consisting of several thousand head-marked rules. Its stochastic component consists of probability weights assigned to the lexicalised grammar rules and to the lexical choice events by the so-called inside-outside algorithm (Lari and Young, 1990), the standard procedure for unsupervised training of a stochastic context-free grammar parsing free text. For training and parsing, the implementations of Carroll (1997b) and Schmid (1999a) were used.

The Gramotron parsing system was designed to be used for the induction of a semantically annotated lexicon of German nouns and verbs (Rooth et al., 1999). Accordingly, the grammar development focus was on the recognition of the grammatical relations between nouns and verbs.

Furthermore, since the parsing results were an intermediate step in an experiment to learn a semantic lexicon, reliable parsing results had to be acquired rapidly. We decided for an incremental grammar development, thus minimizing grammar development efforts in the early project phase.

The context-free grammar for German was developed in three stages: for (i) verb-final clauses, (ii) relative clauses, and (iii) verb-first and verb-second clauses. In this paper, we describe a concluded experiment and evaluation of the parsing system covering constructions (i) and (ii).

Grammar development and stochastic training was controlled by two types of evaluation: (i) an information-theoretic evaluation based on perplexity values measured on training and test corpora of free text, and (ii) a linguistic evaluation of noun chunks with case features and verb frame recognition on a manually annotated test corpus.

2. Data

The data for our experiments are two sub-corpora extracted from a 200 million token newspaper corpus, (a) a sub-corpus containing 450,000 verb-final clauses with a total of 4 million words, and (b) a sub-corpus containing 1,1 million relative clauses with a total of 10 million words. Apart from non-finite clauses as verbal arguments, there are no further clausal embeddings, and the clauses do not contain any punctuation except for a terminal period. The average clause length is 9.16 and 9.12 words per clause, respectively.

We used a finite-state morphological analyser (Schiller and Stöckert, 1995) to assign multiple morphological features such as part-of-speech tag, case, gender and number to the corpus words, partly collapsed to reduce the number of analyses. For example, the word *Bleibe* (either the case ambiguous feminine singular noun 'residence' or a person and mode ambiguous finite singular present tense verb form of 'stay') is analysed as follows:

```
analyse> Bleibe
1. Bleibe+NN.Fem.Akk.Sg
2. Bleibe+NN.Fem.Dat.Sg
3. Bleibe+NN.Fem.Gen.Sg
4. Bleibe+NN.Fem.Nom.Sg
5. *bleiben+V.1.Sg.Pres.Ind
6. *bleiben+V.1.Sg.Pres.Konj
7. *bleiben+V.3.Sg.Pres.Konj
```

Reducing the ambiguous categories leaves the two morphological analyses

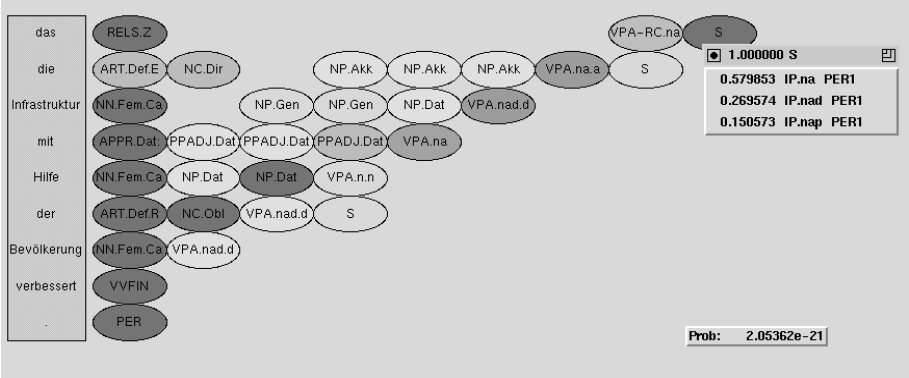


Figure 1: Chart Browser for Grammar Development

```
Bleibe { NN.Fem.Cas.Sg, VVFIN }
```

Apart from assigning morphological analyses the tool in addition serves as lemmatiser (cf. (Schulze, 1996)).

3. The German Context-Free Grammar

The context-free grammar consists of 5,033 rules with lexical head markings. With very few exceptions (rules for coordination, S-rule), the rules do not have more than two daughters. The 220 terminal categories in the grammar correspond to the collapsed corpus tags assigned by the morphology.

Grammar development is facilitated by (a) grammar development environment of the feature-based grammar formalism YAP (Schmid, 1999b), and (b) a chart browser that permits a quick and efficient discovery of grammar bugs (Carroll, 1997a). Figure 1 shows that the ambiguity in the chart is quite considerable even though grammar and corpus are restricted.

The grammar covers 92.43% of the verb-final and 91.70% of the relative clauses, i.e. the respective part of the corpora are assigned parses.

In the following, we describe two essential parts of the grammar, the noun chunks and the definition of subcategorisation frames. For details concerning prepositional phrases, adjectival chunks, adverbial chunks, complex determiners, and the treatment of coordination see (Schulte im Walde, 2000).

3.1. Noun Chunks (NCs)

On nominal categories, in addition to the four cases `Nom`, `Gen`, `Dat`, and `Akk`, case features with a disjunctive interpretation (such as `Dir` for `Nom` or `Akk`) are used. The grammar is written in such a way that non-disjunctive features are introduced high up in the tree. Figure 2 illustrates the use of disjunctive features in noun projections: the terminal `NN` contains the four-way ambiguous `Cas` case feature; the N-bar (`NN1`) and noun chunk `NC` projections disambiguate to two-way ambiguous case features `Dir` and `Obl`; the weak/strong (`Sw/St`) feature of `NN1` allows or prevents combination with a determiner, respectively; only at the noun phrase `NP` projection level, the case feature appears in disambiguated form. The use of disjunctive case features results in some reduction in the size of the parse forest. Essentially the full range of agreement inside the noun phrase is enforced. Agreement between the subject `NP` and the tensed verb is not enforced by the grammar, in order to control the number of parameters and rules.

The noun chunk definition refers to Abney's chunk grammar organisation (Abney, 1996): the noun chunk (`NC`) is a projection that excludes post-head complements and (adverbial) adjuncts introduced higher than pre-head modifiers and determiners, but includes participial pre-modifiers with their complements.

3.2. Subcategorisation Frames

The grammar distinguishes four subcategorisation frame classes: active (`VPA`), passive (`VPP`), non-finite (`VPI`) frames, and copula constructions (`VPK`). A frame may have maximally three arguments. Possible arguments in the frames are nominative (n), dative (d) and accusative (a) NPs, reflexive pronouns (r), PPs (p), and non-finite VPs (i). The grammar does not distinguish plain non-finite VPs from *zu*-non-finite VPs. The grammar is designed to distinguish between PPs representing a verbal complement or adjunct: only complements are referred to by the frame type. The number and the types of frames in the different frame classes are given in Table 1.

Frame Class	#	Frame Types

`VPA`	16	`n, na, nd, np, nad, nap, ndp`
		`ni, di, nai, ndi`
		`nr, nar, ndr, npr, nir`

`VPP`	18	`n, np-s, d, dp-s, p, pp-s`
		`nd, ndp-s, np, npp-s, dp, dpp-s`
		`i, ip-s, ni, nip-s, di, dip-s`

`VPI`	8	`-, a, d, p, r, ad, ap, dp, pr`
`VPK`	2	`n, i`

Table 1: Subcategorisation Frame Types

German, being a language with comparatively free phrase order, allows for scrambling of arguments. Scrambling is reflected in the particular sequence in which the arguments of the verb frame are saturated. Compare Figure 3 as example of a canonical subject-object order within an active transitive frame *der sie liebt* 'who loves her' and its scrambled object-subject order *den sie liebt* 'whom she loves'.

Abstracting from the active and passive realisation of

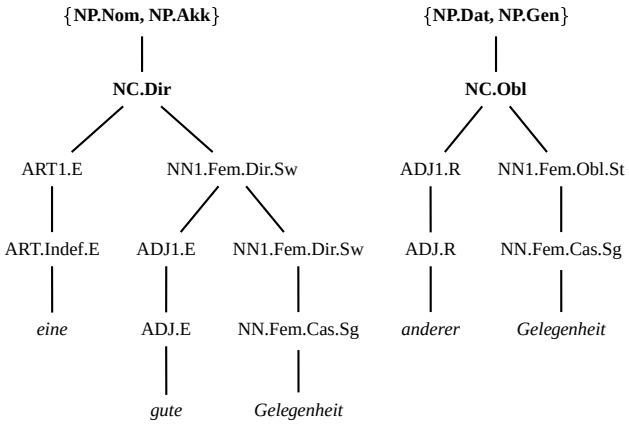


Figure 2: Noun Projections

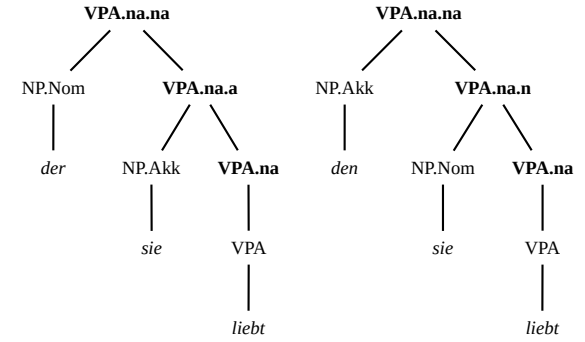


Figure 3: Realising Scrambling Effect in the Grammar Rules

an identical underlying deep-level syntax we generalise over the alternation by defining a top-level subcategorisation frame type, e.g. `IP.nad` for `VPA.nad`, `VPP.nd` and `VPP.ndp-s` (with `p-s` a prepositional phrase within passive frame types representing the deep-structure subject, realisable only by PPs headed by *von* or *durch* 'by'); see Figure 4 for an example, presenting the relative clauses *der die Frau verfolgt* 'who follows the woman', *die verfolgt wird* 'who is followed' and *die von dem Mann verfolgt wird* 'who is followed by the man'.

4. Probability Model

The probabilistic grammars are parsed with `LoPar`[1] (Schmid, 1999a), a head-lexicalised probabilistic context-free parser. The parser is an implementation of the Left-Corner algorithm for parsing and of the Inside-Outside algorithm for parameter estimation. Probabilistic context-free parsing (Lari and Young, 1990) maps a CFG to a probability model by assigning a probability to each grammar rule.

Innovative features of LoPar are head lexicalisation, lemmatisation, parameter pooling, and a sophisticated smoothing technique.

[1]LoPar is basically a re-implementation of the Galacsy tools which were developed by Glenn Carroll in the SFB, but LoPar provides additional functionality.

Syntactically, a head-lexicalised probabilistic context-free grammar (HPCFG) (Carroll, 1995; Carroll and Rooth, 1998) is a PCFG in which one of the right hand side categories of each grammar rule is marked as the head of the projection. The lexical head of a terminal category is the respective word form. Thus, lexical head properties, i.e. words, are propagated through head chains.

HPCFGs assign the following probability[2] to a parse tree `T`:

$$
\begin{aligned}
P(T) \;=\;& P_{start}(\text{cat}(\text{root}(T))) \cdot \\
& P_{start}(\text{head}(\text{root}(T))|\text{cat}(\text{root}(T))) \cdot \\
& \prod_{n \in T} P_{rule}(\text{rule}(n)|\text{cat}(n), \text{head}(n)) \cdot \\
& n \text{ : non-terminal} \\
& \prod_{\substack{n \in T \\ n \neq root(T)}} P_{choice}(\text{head}(n)|\text{cat}(n), \text{cat}(\text{p}(n)), \text{head}(\text{p}(n))) \cdot \\
& \prod_{n \in T} P_{rule}(\texttt{<terminal>}|\text{cat}(n), \text{head}(n)) \cdot \\
& n \text{ : terminal} \\
& \prod_{n \in T} P_{lex}(\text{word}(n)|\text{cat}(n), \text{head}(n)) \\
& n \text{ : terminal}
\end{aligned}
$$

Five families of probability distributions are relevant here. $P_{start}(C)$ is the probability that C is the category of the root node of a parse tree. $P_{start}(h|C)$ is the probability that a root node of category C bears the lexical head h. $P_{rule}(r|C,h)$ is the probability that a node of category C with lexical head h is expanded by rule r. $P_{choice}(h|C,C_p,h_p)$ is the probability that a (non-head) node of category C has the lexical head h given that the parent category is C_p and the parent head is h_p. $P_{rule}(\texttt{<terminal>}|C,h)$ is the probability that a node of category C with lexical head h is a terminal node. $P_{lex}(w|C,h)$, finally, is the probability that a terminal node with category C and lexical head h expands to the word form w.

In order to reduce the prohibitively large number of lexical parameters that have to be estimated, we employed linguistic generalisations for parameter reduction: lemmatisation and parameter pooling. Using uninflected lemma rather than inflected word form for lexicalisation eliminates splitting of estimated frequencies among inflectional forms. Parameter pooling is based on the assumption that lexical choice probabilities are unlikely to depend on inflectional features like gender, case, number etc. of categories or argument order in verb frames. For instance, there are (at least) nine different inflectional patterns for projecting the adjective *alt* (old) and *Buch* (book) to an `NN1` category. Instead of assigning a lexical choice probability

$$P_{choice}(alt|\texttt{ADJ.w}_\texttt{i}, \texttt{NN1.x}_\texttt{i}\texttt{.y}_\texttt{i}\texttt{.z}_\texttt{i}, Buch)$$

[2]The auxiliary functions `cat`, `head`, `p(arent)`, `word` and `rule` return the syntactic category, the lexical head, the parent node, the dominated word or the expanding grammar rule of a node. `root` returns the root node of a parse tree and `<terminal>` is a constant.

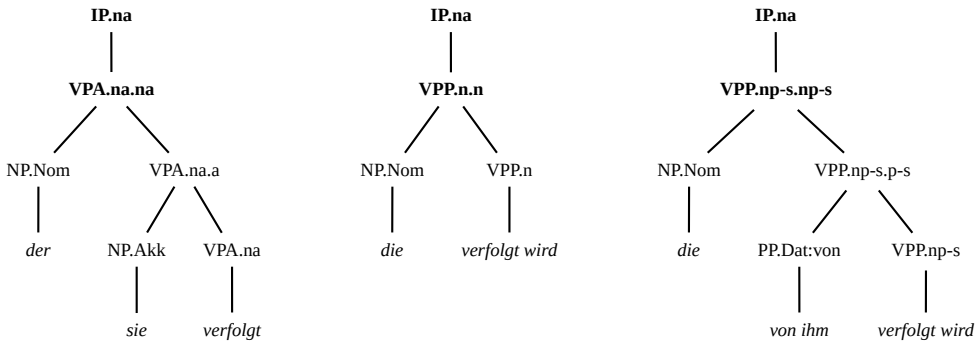


Figure 4: Generalising over the Active-Passive Alternation of Subcategorisation Frames

for each possible combination of w, x, y, z, the combinations are pooled to a single distribution

$$P_{choice}(alt|\texttt{ADJ}, \texttt{NN1}, \text{Buch})$$

for all inflectional variations of `NN1 -> ADJ NN1`. We obtain a single probability distribution for adjectival modifiers. In result, frequent observation of *altes Buch* in the trainng data also increases the probability of *alter Bücher*. For argument filling into verb frames, categories of the form `VP.x.y` are pooled to `VP.x` and active, passive and non-finite verb frames are pooled according to shared arguments, disregarding the saturation state of the frame. For instance, P_{choice} of a particular noun is the same as accusative NP head in the transitive active frame or nominative NP head in the passive frame of a particular verb (*[dass] sie den Hund füttert 'she feeds the dog'*, *der Hund gefüttert wird 'the dog is fed'*).

5. Grammar Training

5.1. Training Strategy

The training in our main experiment was performed in the following steps:

1. Initialisation of all CFG rules with identical frequencies. (Comparative initialisations with random frequencies had no effect on the model development.)

2. Unlexicalised training: The training corpus was parsed once, re-estimating the frequencies twice.

3. Lexicalisation: The unlexicalised model was turned into a lexicalised model by (i) setting the probabilities of the lexicalised rule probabilities to the values of the respective unlexicalised probabilities and (ii) initialising the lexical choice and lexicalised start probabilities uniformly.

4. Lexicalised training:
   Three training iterations were performed on the training corpus, re-estimating the frequencies after each iteration.

For training the model parameters we used 90% of the corpora, a total of 1.4 million clauses. The remaining 10% of serve as heldout data to measure overtraining.

Our experiments have shown that training an unlexicalised model first improves overall results. The optimal training strategy proceeds with few parameter re-estimations of an unlexicalised model. Without re-estimations or with a large number of re-estimations the model was effected to its disadvantage. With less unlexicalised training more changes during lexicalised training take place later on.

Comparative numbers of iterations (up to 40 iterations) in lexicalised training showed that more iterations did not have any further effect on the model.

6. Evaluation

Our evaluation methods were chosen to monitor the development of the grammar, to control the grammar training, and compare different training regimes. As part of our larger project of lexical semantic clustering, the parsing system had the specific task to collect corpus frequencies for pairs of a verbal head and its subcategorisation frame and frequencies for the nominal fillers of slots in a subcategorisation frame. The linguistic evaluation focuses on the reliability of these parsing results.

6.1. Mathematical evaluation

A		B		C	
1:	52.0199	1:	53.7654	1:	49.8165
2:	25.3652	2:	26.3184	2:	23.1008
3:	24.5905	3:	25.5035	3:	22.4479
⋮	⋮	⋮	⋮	⋮	⋮
15:	24.2861	57:	25.0549	90:	22.1443
16:	24.2861	58:	25.0549	95:	22.1443
17:	24.2867	59:	25.055	96:	22.1444

Table 2: Overtraining (iteration: cross-entropy on heldout data)

In order to control the amount of unlexicalised training, we measured overtraining by comparing the perplexity of the model on training and heldout data (or, respectively, cross-entropy[3] on heldout data in the experiments

[3]For a corpus consisting of sentences of a certain average length (`avg`), one can easily transform these cross-entropy values (`cross`) to the better known values of word perplexity (perp)

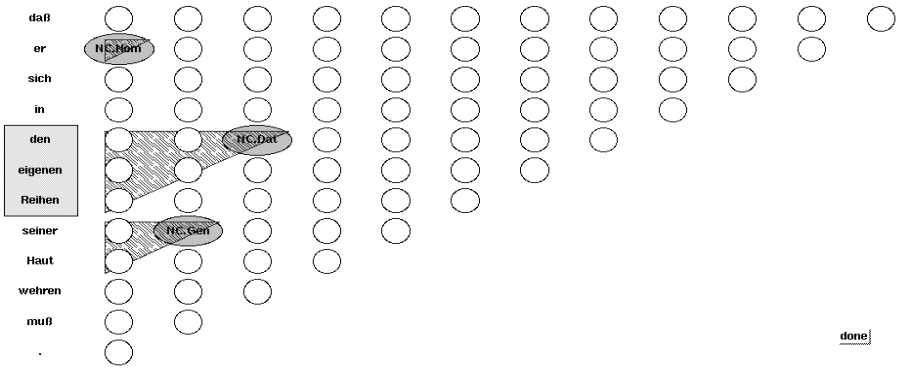


Figure 5: Chart Browser for manual constituent markup

in (Beil et al., 1999)). While perplexity on training data is theoretically guaranteed to converge through subsequent iterations, increasing perplexity on heldout data indicates overtraining. Table 2 shows comparisons of different sizes of training and heldout data (training/heldout) for unlexicalised training in an older experiment (Beil et al., 1999): (A) 50k/50k, (B) 500k/500k, (C) 4.1M/500k. The overtraining effect is indicated by the increase in cross-entropy from the penultimate to the ultimate iteration in the tables.

In previous experiments (Beil et al., 1999), we compared in more detail the mathematical evaluation with the linguistic evaluation of precision/recall measures on categories of different complexity through iterative unlexicalised training. The comparison shows that the mathematical criterion of overtraining may lead to bad results from a linguistic point of view. While precision/recall measures for low-level structures such as NCs converge, iterative unlexicalised training up to the overtraining threshold is disadvantageous for the evaluation of complex categories like subcategorisation frames. We observed precision/recall values for verb frames settling even below the results with a randomly initialised grammar. So the mathematical evaluation can only serve as a rough indicator whether the model reaches towards an optimum, but linguistic evaluation determines the optimum.

6.2. Linguistic evaluation

Although an appropriate treebank is available for German (the NEGRA treebank, cf. Skut et al. (1997) for an overview), we did not use it for our evaluation. One reason for this is the restriction of our initial grammar development to verb final and relative clauses while the treebank, of course, annotates full clauses. It turned out to be difficult to extract respective sub-treebanks. On the other hand, we did not intend to carry out the standard parser evaluation method of measuring precision/recall on phrase boundaries and crossing brackets (the PARSEVAL scheme) for which treebanks are widely used. Bracketing information is rather uninteresting for our objectives and we reckoned that rich structures as generated by our grammar would likely punished by the crossing bracket measure. (For a more general overview of problems using the crossing brackets measure for parser evaluation see (Carroll et al., 1998).)

Moreover, in transforming our bracketing to treebank annotation standards, we feared to loose too much information deemed important for our evaluation. In our efforts to find a transformation that maps treebank structures to a selection of ours (noun and verb chunks), we found two mapping problems: (i) mapping treebank phrase spans to our chunk spans and (ii) finding an information-preserving mapping from our labels to treebank labels. Concerning the first, it turned out to be difficult to define noun chunk ends within treebank NPs. An even harder problem is finding the rich information in our verbal category labels (i.e. type and frame annotation) in treebank VPs.

So we decided to build our own test data: Rather than pursuing the efforts of finding an appropriate treebank-to-gramotron transformation, we performed detailed evaluations of individual frames and of a set of selected verbs.

Test data The linguistic parameters of the models were evaluated concerning the identification of NCs and subcategorisation frames. We randomly extracted 200 relative clauses and 200 verb-final clauses from the test data and hand-annotated the relative clauses with noun chunk labels, and all of the clauses with frame labels. In addition, we extracted 100 randomly chosen relative clauses for each of the six verbs *beteiligen* ‘participate’, *erhalten* ‘receive’, *folgen* ‘follow’, *verbieten* ‘forbid’, *versprechen* ‘promise’, *versuchen* ‘try’, and hand-annotated them with their subcategorisation frames. The particular selection of verbs aims to be representative for the variety of verb frames defined in our grammar.

The manual annotation was facilitated by use of a chart browser. The labellers filled the appropriate chart cells with category names by selecting category labels from a given list that is displayed on clicking a cell. Figure 5 gives an example of NC-labelling which visualises the determination of NC-ranges via cell selection. Frames are annotated as IP

using the formula

$$\texttt{perp} = 10^{\texttt{avg}^{-1} \cdot \texttt{cross}}$$

(assuming that the cross-entropy is computed by a logarithm based on 10). For example, an average lenghth of avg=9.2 and a cross-entropy of cross=24.2 yields a word perplexity perp=427.0, which is a value comparable to the values presented in Schulte im Walde et al. (2001).

labels, i.e. they are always in the same chart cell and frame ranges are trivial.

Best-first consistency Our linguistic evaluation of the probability models is a version of measuring best-first consistency (Briscoe and Carroll, 1993). We made the models determine the Viterbi parses (i.e. maximum probability parses) of the test data and extracted the categories of interest (i.e. noun chunks and subcategorisation frame types). Only the relevant categories but not the entire Viterbi parses were compared with the annotated data. NCs were evaluated according to (i) range and (ii) range and label, i.e. category name. The subcategorisation frames were evaluated according to the frame label only. Precision and recall measures are defined as follows:

$$precision = \frac{correct}{guesses} \qquad recall = \frac{correct}{baseline}$$

with *baseline* referring to the set of annotated categories in the test corpus, *guesses* referring to the set of range/label annotated categories identified in Viterbi parses, and *correct* counting the cases where the chunk/label identified by the parser is a match to the annotator's choice ($correct = guesses \cap baseline$).

Overall results The precision values of the "best" model according to the training strategy were as in Table 3.

Noun Chunks		Subcategorisation Frames on Sub-Corpora	
range	range+label	relative clauses	verb final clauses
98%	92%	63%	73%

Subcategorisation Frames on Specific Verbs					
beteiligen 'participate'	*erhalten* 'receive'	*folgen* 'follow'	*verbieten* 'forbid'	*versprechen* 'promise'	*versuchen* 'try'
48%	61%	88%	59%	80%	49%

Table 3: Precision Values on Noun Chunks and Subcategorisation Frames

For comparison reasons, we evaluated the subcategorisation frames of 200 relative clauses extracted from the training data. Interestingly, there were no striking differences concerning the precision values.

Evaluation of training regimes Figure 6 present the strongly different development of noun chunk and subcategorisation frame representations within the models, ranging from the untrained model until the fifth iteration of lexicalised training. NCs were modelled sufficiently by an unlexicalised trained grammar. Unexpectedly, lexicalisation impaired the modelling slighlty. This observation is supported by related experiments of German noun chunking on an unrestricted text corpus (Schmid and Schulte im Walde, 2000). It remains to be explored whether the number of low-frequent nominal heads is—despite the use of lemmatisation for parameter reduction—still prohibitively large because of the pervasive morpho-syntactic process of noun compounding in German.

Verb phrases in general needed a combination of unlexicalised and lexicalised training, but the representation strongly depended on the specific item. Unlexicalised training advanced frequent phenomena (compare, for example, the representation of the transitive frame with direct object for *erfahren* and with indirect object for *folgen*), lexicalisation and lexicalised training improved the lexicalised properties of the verbs, as expected.

Parameter pooling Regarding the frame evaluation, we also did a test on the effects of parameter pooling in lexicalised traininng. Without pooling of frame categories the precision values for low-frequent phenomena such as non-finite frame recognition was significantly lower, e.g. the precision for the verb *versuchen* was 9% less than with pooling. This result suggests investigations into the importance of training data size and research into other pooling possibilities.

6.3. Error Analysis

A detailed investigation of frame recognition showed the following interesting feature developments:

- Highly common subcategorisation types such as the transitive frame are learned in unlexicalised training and then slightly unlearned in lexicalised training. Less common subcategorisation types such as the demand for an indirect object are unlearned in unlexicalised training, but improved during lexicalised training.
- It is difficult and was not effectively learned to distinguish between prepositional phrases as verbal complements and adjuncts.
- The active present perfect verb complexes and passive of condition were confused, because both are composed by a past participle and a form of *to be*, e.g. *geschwommen ist* 'has swum' vs. *gebunden ist* 'is bound'.
- Copula constructions and passive of condition were confused, again because both may be composed by a past participle and a form of *to be*, e.g. *verboten ist* 'is forbidden' vs. *erfahren ist* 'is experienced'.
- Noun chunks belonging to a subcategorised non-finite clause were partly analysed main verb arguments. For instance, *der ihn zu überreden versucht* 'who him$_{acc}$ tried to persuade' was parsed as demanding an accusative plus a non-finite clause instead of recognising that the accusative object is subcategorised by the embedded infinitival verb.
- Reflexive pronouns may trigger either a reflexive or, by virtue of projecting to an accusative or dative noun chunk, a transitive frame. The correct or wrong choice of frame type containing the reflexive pronoun was learned consequently right or wrong for different verbs. For instance, the verb *sich befinden* 'to be situated' was generally parsed as a transitive, not as inherently reflexive.

6.4. Shortcomings and evaluation alternatives

We are aware that there are some desirable aspects missing from our evaluation.

Firstly, we did not evaluate the relations between lexical heads directly, the main task our parsing system was designed for. Subcategorisation frame and noun chunk label

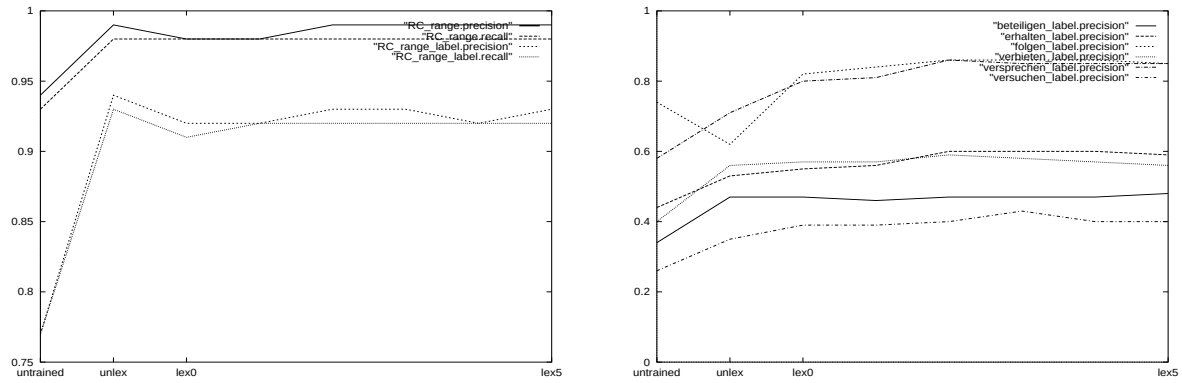


Figure 6: Development of Precision and Recall Values on Noun Chunk Range and Label (left-hand side), and Precision Values on Subcategorisation Frames for Specific Verbs (right-hand side)

recognition serve only as indirect evidence of how well our model does on recognising scrambling of verbal arguments. Because noun chunk annotation is not confined to verb argument slots—PP embedded noun chunks were annotated as well—and a detailed error analysis on noun chunk labels is missing, it remains unclear whether scrambled nominal arguments are subject to more errors than the remarkable 92% precision on NC labels suggests. Similarly, correctly recognised verb frames with a prepositional argument have not been evaluated as to whether the assigned PP argument is actually the correct one.

Secondly, we did not evaluate the correctness of lexical heads of phrases.

Relevant evaluation schemes that capture our shortcomings are the evaluation of dependency structure as described in (Lin, 1995) or the proposal of evaluating of grammatical relations of Carroll et al. (1998). Both evaluation proposals address the importance of selectively evaluating parsing systems with respect to specific types of syntactic phenomena rather than measuring overall performance as in "traditional" evaluation schemes. Selective evaluation is a definite desideratum for our own evaluation task. The proposals also point to a way to automatically extract evaluation relevant relations from an annotated corpus. Inquiring about the feasibility of mapping Negra, the treebank for German, to a respective test corpus will hopefully provide a more comprehensive basis for our future evaluations of head–head relations.

7. Conclusion

Our approach to parsing is a combination of symbolic and stochastic methods. The symbolic component usually involves a very high degree of overgeneration leaving disambiguation to the stochastic component. To facilitate disambiguation by statistical means, the symbolic component relies on certain categorial generalizations and uses non-standard categories to reduce the parameter space or allow for parameter pooling. We used evaluation results in both incremental grammar development and validation of selected output to be used in lexical semantic clustering.

Our principal result is that scrambling-style free-er phrase order, case morphology and subcategorization, and NP-internal gender, number and case agreement can be dealt with in a head-lexicalized PFCG formalism. A second result is that inside-outside estimation appears to be beneficial, however relies on a carefully built grammar where parses can be evaluated by carefully selected linguistic criteria.

Furthermore, we reported experiments on overtraining with inside-outside estimation. These experiments are made possible by the carefully built grammar and our evaluation tools, especially allowing to compare and to relate the results of our mathematical and linguistic evaluation. In combination, these provide a general framework for investigating training regimes for lexicalized PCFGs.

However, there are two relevant aspects missing from our evaluation. First, we did not evaluate grammatical relations directly. Frame and NC case recognition give only a crude idea of how well our model does on recognizing e.g. scrambled subject and direct object. Because NC evaluation is not confined to verb argument slots, the picture is distorted. Second, we did not evaluate the correctness of lexical heads of phrases. Clearly, if we can overcome our difficulties to map Negra, the treebank for German, to a respective test corpus, a more valuable basis for future evaluations of head–head relations supplied by the gramotron parsing system is provided.

Finally, although there is no guarantee that the maximization of the likelihood of the training data (which the inside-outside algorithm performs) also improves the linguistic correctness of the resulting syntactic analyses, our experiments show that in practice this is the case. Gaining more insight into the relationship between linguistic plausibility and likelihood of linguistic analyses will be an interesting future research topic.

8. References

Steven Abney. 1996. Chunk stylebook. Technical report, SfS, Universität Tübingen.

Franz Beil, Glenn Carroll, Detlef Prescher, Stefan Riezler, and Mats Rooth. 1999. Inside-outside estimation of a lexicalized PCFG for German. In *Proceeding of the 37th Annual Meeting of the Association for Computational Linguistics (ACL-99)*, College Park, Maryland.

Ted Briscoe and John Carroll. 1993. Generalised prob-

abilistic LR parsing for unification-based grammars. *Computational Linguistics*, 19(1):25–60.

Glenn Carroll and Mats Rooth. 1998. Valence induction with a head-lexicalized PCFG. In *Proceedings of EMNLP-3*, Granada.

John Carroll, Ted Briscoe, and Antonio Sanfilippo. 1998. Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st International Conference on Language Resources and Evaluation*, Granada, Spain.

Glenn Carroll. 1995. *Learning Probabilistic Grammars for Language Modeling*. Ph.D. thesis, Department of Computer Science, Brown University.

Glenn Carroll, 1997a. *Manual pages for* `charge`, `hyparCharge`. IMS, Universität Stuttgart.

Glenn Carroll, 1997b. *Manual pages for* `supar`, `ultra`, `hypar`. IMS, Universität Stuttgart.

K. Lari and S. J. Young. 1990. The estimation of stochastic context-free grammars using the inside-outside algorithm. *Computer Speech and Language*, 4:35–56.

Dekang Lin. 1995. A dependency-based method for evaluating broad-coverage parsers. In *IJCAI-95*.

M. Rooth, S. Riezler, D. Prescher, G. Carroll, and F. Beil. 1999. Inducing a semantically annotated lexicon via EM-based clustering. In *Proc. of ACL'99*.

Anne Schiller and Chris Stöckert, 1995. *DMOR*. IMS, Universität Stuttgart.

Helmut Schmid and Sabine Schulte im Walde. 2000. Robust German Noun Chunking with a Probabilistic Context-Free Grammar. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING-00)*, pages 726–732, Saarbrücken, Germany, August.

Helmut Schmid, 1999a. *LoPar. Design and Implementation*. Insitut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

Helmut Schmid. 1999b. *YAP: Parsing and Disambiguation with Feature-Based Grammars*. Ph.D. thesis, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart.

Sabine Schulte im Walde, Helmut Schmid, Mats Rooth, Stefan Riezler, and Detlef Prescher. 2001. Statistical grammar models and lexicon acquisition. In *Linguistic Form and its Computation*. CSLI, Stanford, CA.

Sabine Schulte im Walde. 2000. The German statistical grammar model: Development, training and linguistic exploitation. Arbeitspapiere des Sonderforschungsbereichs 340 *Linguistic Theory and the Foundations of Computational Linguistics* 162, Institut für Maschinelle Sprachverarbeitung, Universität Stuttgart, December.

Bruno Maximilian Schulze, 1996. *GermLem – ein Lemmatisierer für deutsche Textcorpora*. IMS, Universität Stuttgart.

Wojciech Skut, Brigitte Krenn, Thorsten Brants, and Hans Uszkoreit. 1997. An annotation scheme for free word order languages. In *Proceedings of the Fifth Conference on Applied Natural Language Processing ANLP-97*, Washington, DC.

Evaluating a Wide-Coverage CCG Parser

Stephen Clark and Julia Hockenmaier

Division of Informatics, University of Edinburgh
2 Buccleuch Place, Edinburgh. EH8 9LW
Scotland, UK
{stephenc,julia}@cogsci.ed.ac.uk

Abstract

This paper compares three evaluation metrics for a CCG parser trained and tested on a CCG version of the Penn Treebank. The standard Parseval metrics can be applied to the output of this parser; however, these metrics are problematic for CCG, and a comparison with scores given for standard Penn Treebank parsers is uninformative. As an alternative, we consider two evaluations based on head-dependencies; one considers local dependencies defined in terms of the derivation tree, and one considers dependencies defined in terms of the CCG categories. The latter set of dependencies includes long-range dependencies such as those inherent in coordination and extraction phenomena.

1. Introduction

In this paper, we compare the advantages and shortcomings of three evaluation metrics for a statistical parser based on Combinatory Categorial Grammar (CCG, Steedman (2000)). The parser (described in Hockenmaier and Steedman (2002b)) is trained and tested on a treebank of CCG normal-form derivations which has been derived (semi)-automatically from the Penn Treebank (Marcus et al., 1993).

We apply the standard Parseval metrics to compare the derivation trees produced by the parser with those in the gold standard. However, CCG derivation trees are binary-branching, and the set of CCG categories is much larger than the set of nonterminal labels in the Penn Treebank. Therefore, a comparison with Parseval figures given for standard Penn Treebank parsers is uninformative. Furthermore, in the presence of left and right modifiers to the same constituent, there are equivalent normal-form derivations, which Parseval does not take into account.

We also consider two dependency evaluations. Like the standard Penn Treebank parsers of Collins (1999) and Charniak (2000), the CCG parser models word-word dependencies defined in terms of local rule applications. Collins (1999) proposes an evaluation based on these dependencies, which we apply to our parser. This allows a direct comparison with Collins' parser and overcomes the problem of equivalent normal-form derivations.

Unlike the phrase-structure trees returned by standard Penn Treebank parsers, CCG derivations encode the long range dependencies involved in constructions such as raising, control, extraction and coordination. In order to evaluate another CCG parser, Clark et al. (2002) introduce an evaluation which incorporates the long range, as well as local, dependencies. This evaluation is applied to the output of the normal-form parser, using the Clark et al. parser to extract the relevant dependencies from the derivation trees. This evaluation is much closer to the dependency-based evaluations of Lin (1995) and Carroll et al. (1998).

2. Combinatory Categorial Grammar

A CCG grammar consists of a lexicon, which pairs words with lexical categories, and a set of combinatory rules, which specify how categories combine. Categories are either atomic or complex. Examples of atomic categories include $S[dcl]$ (declarative sentence), NP (noun phrase), N (noun) and PP (prepositional phrase).

Complex categories are functors which express the type and directionality of their arguments, and the type of the result. For example, the category for the transitive verb *bought* specifies that one NP is required to the right of the verb, and one NP to the left, resulting in a sentence:

(1) bought := $(S[dcl]\backslash NP)/NP$

Other examples of complex categories expressing subcategorisation are as follows ($[pt]$ denotes a past participle and $[pss]$ denotes a passive):

(2) has := $(S[dcl]\backslash NP)/(S[pt]\backslash NP)$
been := $(S[pt]\backslash NP)/(S[pss]\backslash NP)$
bought := $(S[pt]\backslash NP)/NP$
bought := $S[pss]\backslash NP$

Complex categories of the form X/X or $X\backslash X$ can express modification:

(3) big := N/N
quickly := $(S\backslash NP)\backslash(S\backslash NP)$

Constituents combine according to a set of combinatory rules, including function application, function composition and type-raising (see Steedman (2000) for the details). For example, the following derivation uses forward ($>$) and backward ($<$) application:

(4)
IBM	*quickly*	*bought*	*Lotus*
NP	$(S\backslash NP)/(S\backslash NP)$	$(S[dcl]\backslash NP)/NP$	NP
		$S[dcl]\backslash NP$ (>)	
	$S[dcl]\backslash NP$ (>)		
$S[dcl]$ (<)			

Composition and type-raising are necessary for certain types of extraction and coordination phenomena. In the following object-extraction example, type-raising ($>T$) first turns the NP for *IBM* into a functor looking for a verb-phrase, which then combines with the category for *bought* using forward composition ($>B$):

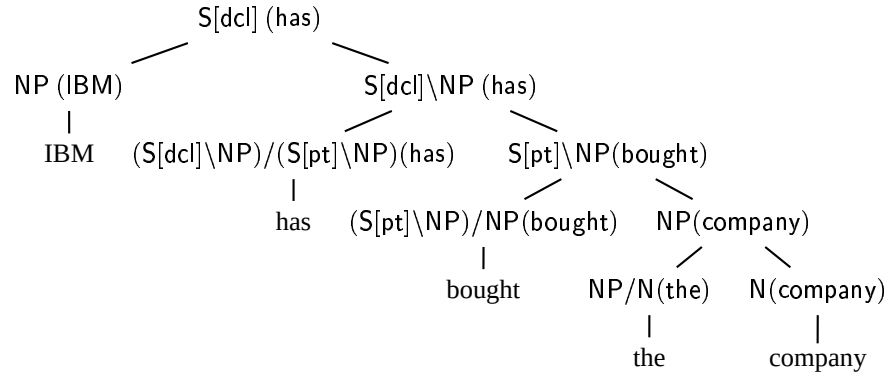


Figure 1: A derivation tree marked with heads

(5) *the company* *that* *IBM* *bought*

NP (NP\NP)/(S/NP) NP (S\NP)/NP

S/(S\NP) (>T)

S/NP (>B)

NP\NP (>)

NP (<)

Note that the use of composition introduces so-called "spurious ambiguity", in which distinct derivations for a sentence lead to the same semantic interpretation. Even a simple sentence such as *IBM bought Lotus* has several derivations, one using only function application, and the others using type-raising and composition. However, all derivations lead to the same interpretation: that *IBM* is the buyer and *Lotus* is the buyee.

One solution to the problem of spurious ambiguity is to only apply function composition when syntactically necessary; such a derivation is called *normal-form*. The corpus that we use to train and test the parser described here contains only normal-form derivations.

3. The parser

The parser that we evaluate is described in Hockenmaier and Steedman (2002b), and is based on a generative model of CCG derivation trees. Like most recent work in statistical parsing – including the generative models of Collins (1997) and Charniak (2000) – the parser models the word-word dependencies defined by local subtrees. Each constituent is assumed to have one lexical head (a word and its lexical category). The example derivation in Figure 1 shows how heads are percolated through the derivation tree.

The statistical model assumes a top-down tree-generating process in which heads are generated at the maximal projection of a constituent. Unless this maximal projection is the root of the entire tree, the constituent is a complement or adjunct of another constituent, and there is a dependency between the the heads of both constituents. This dependency is expressed in the statistical model by conditioning the head of complements or adjuncts on the head of the parent node and the local tree which defines the dependency relation. For example, in Figure 1, *bought* is not only conditioned on its lexical category $(S[pt]\backslash NP)/NP$, but also on the fact that it appears within a local tree with head word *has*, parent $S[dcl]\backslash NP$, left (head) daughter $(S[dcl]\backslash NP)/(S[pt]\backslash NP)$ and right (non-head) daughter $S[pt]\backslash NP$.

The parser is trained and tested on a treebank of CCG normal-form derivations. This corpus, which we call CCGbank, has been derived (semi)-automatically from the Penn Treebank (Marcus et al., 1993), using sections 02-21 for training and section 23 for testing. For further details of CCGbank we refer readers to Hockenmaier and Steedman (2002a).

4. Evaluation metrics

This section describes the different evaluation metrics, which we illustrate by evaluating the (fictitious) output tree in the bottom of figure 2 against the correct derivation given in the top of figure 2.

4.1. Parseval

The first measures are the standard Parseval metrics bracketed precision/recall and labelled precision/recall used to compare the normal-form derivation trees produced by the parser with those in the gold standard (section 23 of the CCGbank).

Following common practice, we disregard punctuation marks. Since CCG derivation trees are at most binary branching, punctuation marks introduce a separate level into the tree, which we also disregard in the evaluation.

Consider the trees given in figure 2. Discarding the lexical categories (but not their unary projections), the gold standard has six nodes, three of which are correctly identified in the output tree. The output tree has seven nonterminal nodes. Hence, labelled and bracketed precision are both 3/7; labelled and bracketed recall are both 3/6. Note that Parseval does not take the correctness of lexical categories into account, which is important for CCG since categories encode subcategorisation information. Therefore, we also give the accuracy of lexical categories (again disregarding punctuation marks), which in this case is 4/6.

4.2. Dependency evaluation 1

Collins (1999) gives an alternative evaluation to Parseval, measuring the recovery of word-word dependencies. According to his definition, there is a dependency between two words w and w' if the parse contains a local tree such that w' is the head of this tree and w is the head of a non-

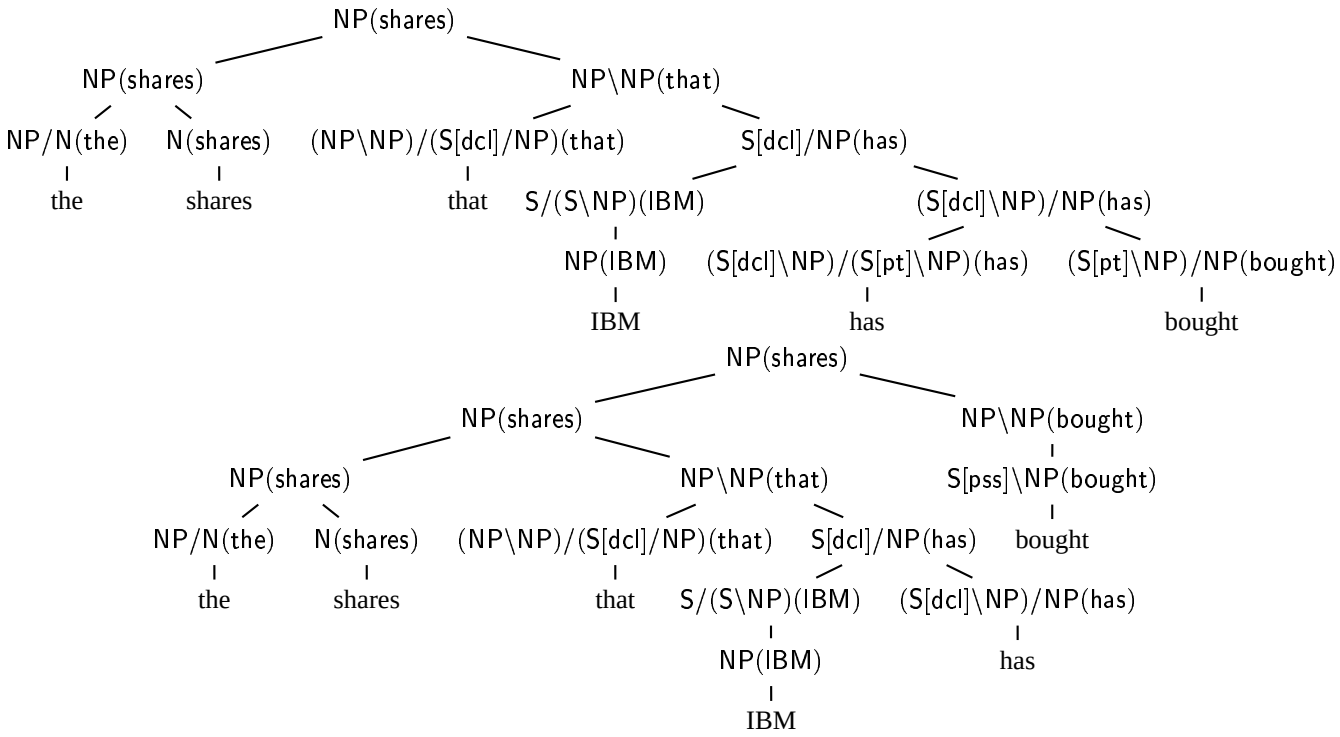
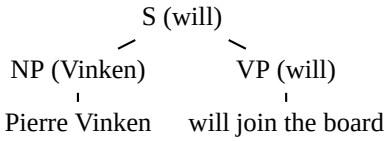


Figure 2: Example trees for evaluation: the top tree is the gold standard.

head daughter. The following tree defines a dependency between *Vinken* and *will*:



The dependency relation is determined by the label of the parent node (S), the label of the head daughter (VP), the label of the non-head daughter (NP), and the direction of the non-head daughter (left): $\langle \mathrm{S}, \mathrm{VP}, \mathrm{NP}, \mathrm{left} \rangle$. Furthermore, if the non-head daughter is a complement, its category carries a complement feature $-\mathrm{C}$. In Collins' original evaluation, coordinate constructions are distinguished by a further element CC. We adapt this evaluation to CCG; however, since the directionality of the head is directly encoded in the categories, there is no need for this feature. A similar comment applies to the complement feature. Also, in CCGbank, binary nodes within a coordinate construction carry a special coordination feature, and so the CC-feature is redundant as well.

The way in which these dependencies are defined means there is exactly one relation to be determined for each word. There is a special relation for the head of the sentence (which is not dependent on any other word). Collins gives scores for labelled and unlabelled dependencies. Unlabelled dependency scores only take into account whether there is a relation between w and w' such that w' is the head and w its modifier or complement, but not whether the local tree which defines this dependency is correctly labelled.

Returning to our example, the gold standard in figure 2 defines the following dependencies:

Relation $\langle Parent, Head, Sister \rangle$	**Head**	**Dep**
$\langle \mathrm{NP}, \mathrm{N}, \mathrm{NP/N} \rangle$	*shares*	*the*
$\langle$Head$\rangle$		*shares*
$\langle \mathrm{NP}, \mathrm{NP}, \mathrm{NP\backslash NP} \rangle$	*shares*	*that*
$\langle \mathrm{NP\backslash NP}, (\mathrm{NP\backslash NP})/(\mathrm{S[dcl]/NP}), \mathrm{S[dcl]/NP} \rangle$	*that*	*has*
$\langle \mathrm{S[dcl]/NP}, (\mathrm{S[dcl]\backslash NP})/\mathrm{NP}, \mathrm{S}/(\mathrm{S\backslash NP}) \rangle$	*has*	*IBM*
$\langle ((\mathrm{S[dcl]\backslash NP})/\mathrm{NP}, (\mathrm{S[dcl]\backslash NP})/(\mathrm{S[pt]\backslash NP}),$ $(\mathrm{S[pt]\backslash NP})/\mathrm{NP} \rangle$	*has*	*bought*

These are the dependencies in the incorrect analysis:

Relation $\langle Parent, Head, Sister \rangle$	**Head**	**Dep**
$\langle \mathrm{NP}, \mathrm{N}, \mathrm{NP/N} \rangle$	*shares*	*the*
$\langle$Head$\rangle$		*shares*
$\langle \mathrm{NP}, \mathrm{NP}, \mathrm{NP\backslash NP} \rangle$	*shares*	*that*
$\langle \mathrm{NP\backslash NP}, (\mathrm{NP\backslash NP})/(\mathrm{S[dcl]/NP}), \mathrm{S[dcl]/NP} \rangle$	*that*	*has*
$\langle \mathrm{S[dcl]/NP}, (\mathrm{S[dcl]\backslash NP})/\mathrm{NP}, \mathrm{S}/(\mathrm{S\backslash NP}) \rangle$	*has*	*IBM*
$\langle \mathrm{NP}, \mathrm{NP}, \mathrm{NP\backslash NP} \rangle$	*shares*	*bought*

Thus, according to this measure, five out of six dependencies are correct. Note that this measure is not always affected by errors in the lexical categories. For example, the dependency between *has* and *that* is considered correct, even though the gold standard analyses *has* as an auxiliary and the incorrect derivation analyses *has* as a transitive verb.

4.3. Dependency evaluation 2

The parser in Clark et al. (2002) can be used to yield a third measure. This parser uses CCG categories extended with head and dependency information and captures the "deep" dependencies inherent in cases such as raising, control, and extraction and coordination phenomena, as well

as the standard local dependencies. Figure 3 is an example from Clark et al. (2002), with the links expressing dependencies. (The labels are omitted for clarity.) Note that *investors* and *managers* are both subjects of *want*, and subjects of *lock*.

An example of an extended category for the transitive verb *bought* is as follows:

(6) bought := $(S_{bought}\backslash NP_1)/NP_2$

There are two dependency relations encoded: the subject of the transitive verb – here marked 1 – and the direct object – here marked 2. The subscript on the S category indicates that the head of the resulting sentence is *bought*. Since the argument slots in CCG categories correspond closely to the grammatical relations used by Carroll et al. (1998), this dependency evaluation is very much in the spirit of the Carroll et al. evaluation (and that of Lin (1995)).

A dependency is formally defined as a 4-tuple: $\langle h_f, f, s, h_a \rangle$, where h_f is the head word of the functor, f is the functor category (extended with dependency information), s is the argument slot, and h_a is the head word of the argument. For example, in the sentence *IBM bought Lotus*, the subject-verb dependency is as follows:

(7) $\langle bought, (S\backslash NP_1)/NP_2, 1, IBM \rangle$

The category set used by the parser consists of 398 category types (chosen according to frequency), derived automatically from the CCGbank. Each category has been manually marked-up with head and dependency information, and at this stage we encode every argument slot as a dependency relation. In future work we may use only a subset of the argument slots.

In order to recover such dependencies from the trees produced by the normal-form parser, the Clark et al. (2002) parser is run over the trees output by the normal-form parser, tracing out the derivations and outputting the dependencies. This method can also be applied to the trees in the test set, in order to provide a set of gold standard dependency structures. Note that the marked-up categories used by the Clark et al. parser are necessary to obtain these dependencies; without this information, they cannot be derived from the local dependencies used in the first dependency evaluation.

The evaluation metrics we use are precision and recall over the dependencies (labelled and unlabelled). To obtain a point for a labelled dependency, the head, dependent, functor category, and slot must all be correct. To obtain a point for an unlabelled dependency, the head and dependent must have appeared together in some relation (in any order) in the gold standard. The dependencies obtained from the trees in Figure 2 are given in table 1. The scores for the incorrect tree are $3/6$ labelled precision, $3/7$ labelled recall, $5/6$ unlabelled precision, and $5/7$ unlabelled recall.

5. Results and discussion

The results for the three evaluation metrics on Section 23 of CCGbank are given in Table 2. BP is bracketed precision; LP is labelled precision; UP is unlabelled precision. BR, LR, UR are defined similarly for recall. The scores for each evaluation are accumulated over all sentences in the test set, rather than averaged per sentence. We also give the score for accuracy of the lexical categories.

Gold standard
$\langle the, NP/N_1, 1, shares \rangle$
$\langle that, (NP\backslash NP_1)/(S[dcl]_2\backslash NP), 1, shares \rangle$
$\langle that, (NP\backslash NP_1)/(S[dcl]_2\backslash NP), 2, has \rangle$
$\langle has, (S[dcl]\backslash NP_1)/(S[pt]_2\backslash NP), 1, IBM \rangle$
$\langle has, (S[dcl]\backslash NP_1)/(S[pt]_2\backslash NP), 2, bought \rangle$
$\langle bought, (S[pt]\backslash NP_1)/NP_2, 1, IBM \rangle$
$\langle bought, (S[pt]\backslash NP_1)/NP_2, 2, shares \rangle$
Example tree
$\langle the, NP/N_1, 1, shares \rangle$
$\langle that, (NP\backslash NP_1)/(S[dcl]_2\backslash NP), 1, shares \rangle$
$\langle that, (NP\backslash NP_1)/(S[dcl]_2\backslash NP), 2, has \rangle$
$\langle has, (S[dcl]\backslash NP_1)/NP_2, 1, IBM \rangle$
$\langle has, (S[dcl]\backslash NP_1)/NP_2, 2, shares \rangle$
$\langle bought, S[pss]\backslash NP_1, 1, shares \rangle$

Table 1: Dependencies for the trees in Figure 2

Accuracy of lexical categories			
92.0%			
Parseval			
LP	LR	BP	BR
81.6%	81.9%	85.5%	85.9%
Tree dependencies			
Labelled recall		Unlabelled recall	
84.0%		90.1%	
"Deep" dependencies			
LP	LR	UP	UR
83.7%	84.2%	90.5%	91.1%

Table 2: Results for the three evaluation metrics

5.1. The Parseval scores

It is hard to draw conclusions from the Parseval scores because of the difficulty in comparing results across different tree representations. Our figures are below the 88.3%/88.0% labelled precision/recall of Collins (1999). However, a direct comparison of the Parseval result with Penn Treebank parsers is not informative, even for the same set of sentences. Because Penn Treebank trees are very flat, they contain far fewer brackets than CCG derivation trees; hence the rate of crossing brackets (and bracketed precision and recall) will automatically be much lower than for a grammar which produces at most binary-branching trees. The flat trees also mean that Parseval is too lenient towards mis-attachments produced by Penn Treebank parsers (Manning and Schütze, 1999). Furthermore, the set of node labels for Penn Treebank trees and the set of CCG categories are not comparable.

Hockenmaier (2001) notes a further problem with applying Parseval metrics to CCG derivation trees. Consider verb phrases, $(S\backslash NP)$, which can have left and right modifiers ($(S\backslash NP)/(S\backslash NP)$ and $(S\backslash NP)\backslash(S\backslash NP)$) with the following two rule instantiations:

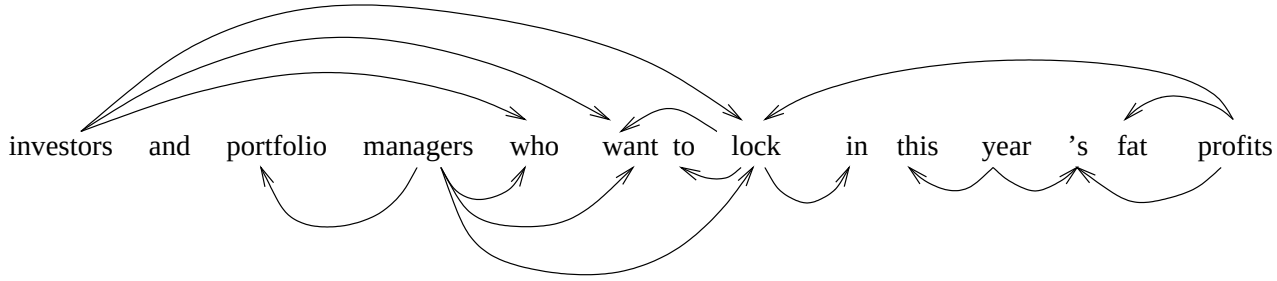
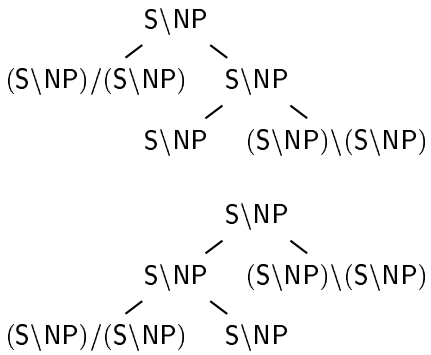


Figure 3: Example dependency structure

(8) $S\backslash NP \rightarrow (S\backslash NP)/(S\backslash NP)\ \ S\backslash NP$
$S\backslash NP \rightarrow S\backslash NP\ \ (S\backslash NP)\backslash(S\backslash NP)$

For any parsing model which is defined in terms of (possibly headed or lexicalized) local trees, the following two trees are equivalent:



This is also the case for the normal-form parser described above. A similar problem arises with coordinations/lists involving more than two conjuncts. The dependency evaluations described below do not suffer from this problem because the same dependencies are produced for each derivation.

5.2. Dependency evaluation 1

As expected, the results for the tree dependencies are higher than the Parseval scores. Unlike Parseval, the dependency measure is neutral with respect to the branching factor of the trees produced by the grammar. In particular, for a given sentence, the number of dependencies is identical to the number of words in the sentence. Since this is the same for any parser, unlabelled recovery of dependencies can be used to indicate how parsers based on different grammars compare. Note that our unlabelled figures (90.1% recall) are similar to those of Collins (90.9%).

However, a direct comparison with the labelled figures given by Collins (88.3% recall) is again problematic. First, the sets of labels are very different. In order for labelled dependencies as defined by a CCG derivation tree to be correct, complement-adjunct distinctions as well as extraction cases have to be correctly recovered. Extraction is not indicated in the trees returned by Collins' parser, and can therefore not be evaluated. Mistaking a complement daughter for an adjunct or vice versa has a much greater effect on the labelled scores for CCG than for Penn Treebank parse trees. In Collins' parser, the complement-adjunct distinction is only expressed in the label of the particular node in question. However, in CCG this can affect the entire tree below the parent – both the subtree underneath the head daughter and the subtree underneath the non-head daughter.

In addition, Collins performs the following preprocessing steps on the output of his parser and the Gold standard: all POS tags are replaced by a single token "TAG". All complement markings on the parent and head node are removed, so that one attachment decision made higher up in the tree does not affect the evaluation of its daughter. We cannot readily perform the same preprocessing steps: the choice of lexical categories can affect the tree at several levels, not just at the leaf nodes; furthermore, complement-adjunct distinctions are also encoded in all intermediate categories, not just a constituent's maximal projection.

5.3. Dependency evaluation 2

One of the advantages of a dependency-style evaluation is that the scores can be broken down by relation, as shown in Table 4, which gives scores for some of the most frequent types.[1] The table also gives some indication of the kinds of relations used in the evaluation.

The relations are defined in terms of CCG categories, which raises the question of how these compare with a more generic set such as that proposed by Carroll et al. (1998). First note that there are many more relations in our scheme: around 700 in total compared with 20 for Carroll et al. We have so many relations because each argument slot in each category (of which there are 398) encodes a separate relation.

Clearly there is room for generalisation in our scheme. For example, Carroll et al. have one relation for subjects, whereas we have a different relation for each category type encoding a subject. Examples of two categories encoding subject relations are $(S[dcl]\backslash NP_1)/NP_2$ and $(S[b]\backslash NP_1)/NP_2$. In future work we will investigate mapping our relations onto Carroll et al.'s.

One potential weakness of our evaluation (which follows from encoding all argument slots as relations) is that some relations are effectively counted more than once. For

[1] #ref is the number of dependencies with the given relation type in the gold standard; #test is the number of dependencies with the given relation type produced by the parser; LP/LR are labelled precision/recall; and the F-score is calculated as (2*LP*LR)/(LP+LR).

$\langle P, H, S \rangle$	#ref	#test	LP%	LR%
$\langle NP, NP, NP\backslash NP \rangle$	3,765	3,626	75.2	72.4
$\langle HEAD \rangle$	2371	2367	94.5	94.4
$\langle NP, NP, NP[conj] \rangle$	935	1,075	61.3	70.5
$\langle S[dcl]\backslash NP, S[dcl]\backslash NP, (S\backslash NP)\backslash(S\backslash NP) \rangle$	914	905	60.9	60.3
$\langle S[dcl]\backslash NP, (S[dcl]\backslash NP)/NP, NP \rangle$	880	858	86.7	84.6
$\langle S[pss]\backslash NP, S[pss]\backslash NP, (S\backslash NP)\backslash(S\backslash NP) \rangle$	442	470	70.6	75.1

Table 3: Some dependency relations in evaluation 1

Functor	Slot	Category description	LP %	# test	LR %	# ref	F-score
$N_X/N_{X,1}$	1	*nominal modifier*	94.4	7,856	93.2	7,955	93.8
$NP_X/N_{X,1}$	1	*determiner*	96.7	4,548	96.4	4,566	96.5
$(NP_X\backslash NP_{X,1})/NP_2$	2	*np modifying preposition*	82.1	2,659	81.2	2,690	81.6
$(NP_X\backslash NP_{X,1})/NP_2$	1	*np modifying preposition*	76.0	2,449	76.2	2,443	76.1
$(S_X\backslash NP_Y)\backslash(S_{X,1}\backslash NP_Y)/NP_2$	2	*vp modifying preposition*	68.7	1,327	66.1	1,379	67.4
$(S_X\backslash NP_Y)\backslash(S_{X,1}\backslash NP_Y)/NP_2$	1	*vp modifying preposition*	66.2	1,247	65.0	1,271	65.6
$(S[dcl]\backslash NP_1)/NP_2)$	1	*transitive verb*	83.2	885	82.0	898	82.6
$(S[dcl]\backslash NP_1)/NP_2)$	2	*transitive verb*	80.3	885	78.4	907	79.3
$(S_X\backslash NP_Y)\backslash(S_{X,1}\backslash NP_Y)$	1	*adverbial modifier*	81.5	961	82.2	953	81.8
(PP/NP_1)	1	*preposition complement*	61.5	993	75.7	807	67.9
$(S[b]\backslash NP_1)/NP_2)$	2	*infinitival transitive verb*	86.6	719	85.2	731	85.9
$(S[dcl]\backslash NP_{X,1})/(S[b]_2\backslash NP_X)$	2	*auxiliary*	97.6	631	98.6	625	98.1
$(S[dcl]\backslash NP_{X,1})/(S[b]_2\backslash NP_X)$	1	*auxiliary*	92.2	638	95.0	619	93.6
$(S[b]\backslash NP_1)/NP_2$	1	*infinitival transitive verb*	80.6	566	83.1	549	81.8
$(NP_X/N_{X,1})\backslash NP_2$	1	*s genitive*	96.6	472	95.2	479	95.9
$(NP_X/N_{X,1})\backslash NP_2$	2	*s genitive*	92.5	482	95.3	468	93.9
$(S[dcl]\backslash NP_1)/S[dcl]_2$	1	*sentential complement verb*	93.0	431	95.5	420	94.2
$(NP_X\backslash NP_{X,1})/(S[dcl]_2\backslash NP_X)$	1	*subject relative pronoun*	71.9	295	72.6	292	72.2
$(NP_X\backslash NP_{X,1})/(S[dcl]_2\backslash NP_X)$	2	*subject relative pronoun*	94.5	289	95.5	286	95.0

Table 4: Results for dependency evaluation 2 by relation; only a subset of the relations are shown

example, in the sentence *John has been eating beans*, *John* is evaluated as a subject three times: as the subject of *has*, *been* and *eating*. But if the subject of *eating* is correct in this example, then the subjects of the auxiliary verbs will be correct as well.

We would also like to make a distinction between arguments that have been extracted from a predicate, and those that are "in situ". Currently the direct object of a verb, for example, is the same relation whether it has been extracted or not. It would be useful to at least have the option to make this distinction.

5.4. Comparing the dependency evaluations

The dependencies expressed in dependency evaluation 1 are not simply a subset of the relations used in the second dependency evaluation. When the relations are broken down individually, this leads to an interesting comparison.

In dependency evaluation 2, it is possible to determine how well nominal prepositions have been recovered, whereas in dependency evaluation 1, we can only evaluate how well NP postmodifiers have been recovered.

In contrast to dependency evaluation 2, dependency evaluation 1 includes a separate relation for the head of a sentence $\langle HEAD \rangle$ (assuming a single head for each sentence, including coordinate structures).

In dependency evaluation 1, it can be seen for each type of constituent whether coordination is recovered properly, e.g. $\langle NP, NP, NP[conj] \rangle$. In dependency evaluation 2, coordination relations are not represented explicitly.

Some relations in dependency evaluation 1 (like the direct object of transitive declaratives, $\langle S[dcl]\backslash NP, (S[dcl]\backslash NP)/NP, NP \rangle$) seem to be the same as in evaluation 2. However, in dependency evaluation 1 only non-extracted cases are taken into account.

6. Conclusion

We have presented three evaluations for a wide-coverage CCG parser. Of these, Parseval seems the least appropriate, especially if a comparison is to be made with existing Penn Treebank parsers. In an attempt to compare with the Collins parser, we adopted a dependency evaluation in which dependencies are defined in terms of local trees; however, the different labelling used in the CCG derivation tree compared to the Penn Treebank made the comparison of labelled dependencies problematic. The comparison of unlabelled dependencies was more appropriate, however.

One of the features of CCG is its analysis of long-range dependencies. In an attempt to incorporate such dependencies into the evaluation, we proposed a second dependency evaluation, in which the dependency relations are defined in terms of the CCG categories. This is closer to evaluations based on grammatical relations, although if a comparison is to be made with parsers using such an evaluation, a map-

ping is required between the CCG dependencies and the set of grammatical relations.

7. Acknowledgements

This research was funded by EPSRC grant GR/M96889/01 and an EPSRC studentship to the second author. We would like to thank Mark Steedman for his guidance and expert help with this work.

8. References

John Carroll, Ted Briscoe, and Antonio Sanfilippo. 1998. Parser evaluation: a survey and a new proposal. In *Proceedings of the 1st LREC Conference*, pages 447–454, Granada, Spain.

Eugene Charniak. 2000. A maximum-entropy-inspired parser. In *Proceedings of the 1st Meeting of the NAACL*, pages 132–139, Seattle, WA.

Stephen Clark, Julia Hockenmaier, and Mark Steedman. 2002. Building deep dependency structures using a wide-coverage CCG parser. In *Proceedings of the 40th Meeting of the ACL (to appear)*, Philadelphia, PA.

Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Meeting of the ACL*, pages 16–23, Madrid, Spain.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Julia Hockenmaier and Mark Steedman. 2002a. Acquiring compact lexicalized grammars from a cleaner treebank. In *Proceedings of the Third International Conference on Language Resources and Evaluation (to appear)*, Las Palmas, Spain.

Julia Hockenmaier and Mark Steedman. 2002b. Generative models for statistical parsing with Combinatory Categorial Grammar. In *Proceedings of the 40th Meeting of the ACL (to appear)*, Philadelphia, PA.

Julia Hockenmaier. 2001. Statistical parsing for CCG with simple generative models. In *Proceedings of Student Research Workshop, 39th Meeting of the ACL*, Toulose, France.

Dekang Lin. 1995. A dependency-based method for evaluating broad-coverage parsers. In *Proceedings of IJCAI-95*, pages 1420–1425, Montreal, Canada.

Christopher Manning and Hinrich Schütze. 1999. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, MA.

Mitchell Marcus, Beatrice Santorini, and Mary Marcinkiewicz. 1993. Building a large annotated corpus of English: The Penn Treebank. *Computational Linguistics*, 19(2):313–330.

Mark Steedman. 2000. *The Syntactic Process*. The MIT Press, Cambridge, MA.

A Comparison of Evaluation Metrics for a Broad-Coverage Stochastic Parser

Richard Crouch, Ronald M. Kaplan, Tracy H. King, Stefan Riezler

Palo Alto Research Center
3333 Coyote Hill Road
Palo Alto, CA, 94025
{crouch|kaplan|thking|riezler}@parc.com

Abstract

This paper reports on the use of two distinct evaluation metrics for assessing a stochastic parsing model consisting of a broad-coverage Lexical-Functional Grammar (LFG), an efficient constraint-based parser and a stochastic disambiguation model. The first evaluation metric measures matches of predicate-argument relations in LFG f-structures (henceforth the LFG annotation scheme) to a gold standard of manually annotated f-structures for a subset of the UPenn Wall Street Journal treebank. The other metric maps predicate-argument relations in LFG f-structures to dependency relations (henceforth DR annotations) as proposed by Carroll et al. (Carroll et al., 1999). For evaluation, these relations are matched against Carroll et al.'s gold standard which was manually annnotated on a subset of the Brown corpus. The parser plus stochastic disambiguator gives an F-measure of 79% (LFG) or 73% (DR) on the WSJ test set. This shows that the two evaluation schemes are similar in spirit, although accuracy is impaired systematically by mapping one annotation scheme to the other. A systematic loss of accuracy is incurred also by corpus variation: Training the stochastic disambiguation model on WSJ data and testing on Carroll et al.'s Brown corpus data yields an F-score of 74% (DR) for dependency-relation match. A variant of this measure comparable to the measure reported by Carroll et al. yields an F-measure of 76%. We examine divergences between annotation schemes aiming at a future improvement of methods for assessing parser quality.

1. Introduction

Recent years have seen increased interest in parsing systems that capture predicate-argument relations instead of mere phrase-structure representations. In aiming for this goal, considerable progress has been made by combining systems of hand-coded, linguistically fine-grained grammars with robustness techniques and stochastic disambiguation models. However, it can reasonably be argued that the standard evaluation procedure for stochastic parsing—precision and recall of matching labeled bracketing to section 23 of the UPenn Wall Street Journal (WSJ) treebank (Marcus et al., 1994)—is not appropriate for assessing the quality of parsers on matching predicate-argument relations. A new standard for evaluation on predicate-argument relations and for annotating a gold standard is needed.

In this paper we present a stochastic parsing model consisting of a broad-coverage Lexical-Functional Grammar (LFG), a constraint-based parser and a stochastic disambiguation model, and discuss the evaluation of this system on two distinct evaluation metrics for assessing the quality of the stochastic parsing model on matching predicate-argument relations. The first evaluation metric measures matches of predicate-argument relations in LFG f-structures (henceforth the LFG annotation scheme) to a gold standard of manually annotated f-structures for a representative subset of the WSJ treebank. The evaluation measure counts the number of predicate-argument relations in the f-structure of the parse selected by the stochastic model that match those in the gold standard annotation.

The other metric we employed maps predicate-argument relations in LFG f-structures to the dependency relations (henceforth the DR annotation scheme) proposed by Carroll et al. (Carroll et al., 1999). Evaluation with this metric measures the matches of these relations to Carroll et al.'s gold standard corpus.

Our parser plus stochastic disambiguator gives an F-measure of 79% (LFG) or 73% (DR) on the WSJ test set, showing that the two evaluation schemes are similar in spirit. However, accuracy is systematically impaired by mapping one annotation scheme to the other. A systematic loss of accuracy is incurred also by corpus variation: Training the stochastic disambiguation model on WSJ data and testing on Carroll et al.'s Brown corpus data gives a DR F-measure of 74% for matching dependency relations. For a direct comparison of our results with Carroll et al.'s system, we also computed an F-measure that does not distinguish different types of dependency relations. Under this measure we obtain 76% F-measure.

One goal of this paper is to highlight possible pitfalls and error sources in translating between different annotation schemes and gold standards. We believe that a thorough investigation of divergences in annotation schemes will facilitate a future standard for predicate-argument evaluation and annotation.

This paper is organized as follows. After introducing the grammar and parser used in this experiment, we describe in section 2. the robustness techniques employed to reach 100% grammar coverage on unseen WSJ text (in the sense of the proportion of sentences for which at least one analysis is found). Furthermore, we give in section 3. a short account of the stochastic model used for disambiguating LFG parses. Experiments on evaluating the combined system of parser and stochastic disambiguator on the two distinct evaluation measures and corpora are described in section 4.

2. Robust Parsing using LFG

2.1. A Broad-Coverage Lexical-Functional Grammar

The grammar used for this project has been developed in the ParGram project (Butt et al., 1999). It uses LFG as a formalism, producing c(onstituent)-structures (trees) and

f(unctional)-structures (attribute value matrices) as output. The c-structures encode constituency. Each c-structure has at least one corresponding f-structure. F-structures encode predicate-argument relations and other grammatical information, e.g., number, tense. The XLE parser (Maxwell and Kaplan, 1993) was used to produce packed representations, specifying all possible grammar analyses of the input.

The grammar has 314 rules with regular expression right-hand sides which compile into a collection of finite-state machines with a total of 8,759 states and 19,695 arcs. The grammar uses several lexicons and two guessers: one guesser for words recognized by the morphological analyzer but not in the other lexicons and one for those not recognized. As such, most common and proper nouns, adjectives, and adverbs have no explicit lexical entry. The main verb lexicon contains 9,652 verb stems and 23,525 subcategorization frame-verb stem entries; there are also lexicons for adjectives and nouns with subcategorization frames and for closed class items such as prepositions.

For estimation and testing purposes using the WSJ treebank, the grammar was modified to parse part of speech tags and labeled bracketing. A stripped down version of the WSJ treebank was created that used only those POS tags and labeled brackets relevant and reliable for determining grammatical relations. The WSJ labels are given entries in a special LFG lexicon, and these entries constrain both the c-structure and the f-structure of the parse. For example, the WSJ's ADJP-PRD label must correspond to an AP in the c-structure and an XCOMP in the f-structure. In this version of the corpus, all WSJ labels with -SBJ are retained and are restricted to phrases corresponding to SUBJ in the LFG grammar; in addition, it contains NP under VP (OBJ and OBJth in the LFG grammar), all -LGS tags (OBL-AG), all -PRD tags (XCOMP), VP under VP (XCOMP), SBAR- (COMP), and verb POS tags under VP (V in the c-structure). For example, our labeled bracketing version of wsj_1305.mrg is *[NP-SBJ His credibility] is/VBZ_ also [PP-PRD on the line] in the investment community.*

Some mismatches between the WSJ labeled bracketing and the LFG grammar remain. These often arise when a given constituent fills a grammatical role in more than one clause, usually when it is a SUBJ or OBJ in one clause and also the SUBJ of an XCOMP complement. For example, in wsj_1303.mrg *Japan's Daiwa Securities Co. named Masahiro Dozen president.*, the noun phrase *Masahiro Dozen* is labeled as an NP-SBJ, presumably because it is the subject of a small clause complement. However, the LFG grammar treats it also as the OBJ of the matrix clause. As a result, the labeled bracketed version of this sentence does not receive a full parse, even though the LFG output from parsing its unlabeled, string-only counterpart is well-formed. Some other bracketing mismatches remain between this stripped down WSJ corpus and the LFG grammar; these are usually the result of adjunct attachment. Such mismatches occur in part because, besides minor modifications to match the bracketing for special constructions, e.g., negated infinitives, the grammar was not altered to mirror the WSJ bracketing.

2.2. Robustness Techniques

To increase robustness, the standard grammar has been augmented with a FRAGMENT grammar. This grammar parses the sentence as well-formed chunks specified by the grammar, in particular as Ss, NPs, PPs, and VPs. These chunks have both c-structures and f-structures corresponding to them, just as in the standard grammar. Any substring that cannot be parsed as one of these chunks is parsed as a TOKEN chunk. The TOKENs are also recorded in the c- and f-structures. The grammar has a fewest-chunk method for determining the correct parse. For example, if a string can be parsed as two NPs and a VP or as one NP and an S, the NP-S option is chosen.

A final capability of XLE that increases coverage of the standard plus fragment grammar on the WSJ corpus is a SKIMMING technique. Skimming is used to avoid timeouts and memory problems when parsing unusually difficult sentences in the corpus. When the amount of time or memory spent on a sentence exceeds a threshhold, XLE goes into skimming mode for the constituents whose processing has not been completed. When XLE skims these remaining constituents, it does a bounded amount of work per subtree. This guarantees that XLE finishes processing a sentence in a polynomial amount of time, although it does not necessarily return the complete set of analyses. In parsing section 23, 7.2% of the sentences were skimmed; 26.1% of the skimmed sentences resulted in full parses, while 73.9% were fragment parses.

The final grammar coverage achieved 100% of section 23 as unseen unlabeled data: 74.7% of those were full parses, 25.3% FRAGMENT and/or SKIMMED parses.

3. Discriminative Statistical Estimation from Partially Labeled Data

3.1. Exponential Probability Models on LFG Parses

The probability model we employed for stochastic disambiguation is the well-known family of exponential models. These models have already been applied successfully for disambiguation of various constraint-based grammars (LFG (Johnson et al., 1999), HPSG (Bouma et al., 2000), DCG (Osborne, 2000)).

In this paper we are concerned with conditional exponential models of the form:

$$p_{\boldsymbol{\lambda}}(x|y) = Z_{\boldsymbol{\lambda}}(y)^{-1} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)}$$

where $X(y)$ is the set of parses for sentence y, $Z_{\boldsymbol{\lambda}}(y) = \sum_{x \in X(y)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)}$ is a normalizing constant, $\boldsymbol{\lambda} = (\lambda_1, \ldots, \lambda_n) \in \mathbb{R}^n$ is a vector of log-parameters, $\boldsymbol{f} = (f_1, \ldots, f_n)$ is a vector of property-functions $f_i : \mathcal{X} \to \mathbb{R}$ for $i = 1, \ldots, n$ on the set of parses $\mathcal{X}$, and $\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)$ is the vector dot product $\sum_{i=1}^{n} \lambda_i f_i(x)$.

In our experiments, we employed around 1000 complex property-functions comprising information about c-structure, f-structure, and lexical elements in parses, similar to the properties used in Johnson et al. (1999). For example, there are property functions for c-structure nodes and c-structure subtrees, indicating attachment preferences. High versus low attachment is indicated by property functions counting the number of recursively embedded phrases.

Other property functions are designed to refer to f-structure attributes, corresponding to grammatical functions in LFG, or to atomic attribute-value pairs in f-structures. More complex property functions are designed to indicate, for example, the branching behaviour of c-structures and the (non)-parallelism of coordinations on both c-structure and f-structure levels. Furthermore, properties refering to lexical elements based on an auxiliary distribution approach as presented in Riezler et al. (2000) are included in the model. Here tuples of head words, argument words, and grammatical relations are extracted from the training sections of the WSJ, and fed into a finite mixture model for clustering grammatical relations. The clustering model itself is then used to yield smoothed probabilities as values for property functions on head-argument-relation tuples of LFG parses.

3.2. Discriminative Estimation

Discriminative estimation techniques have recently received great attention in the statistical machine learning community and have already been applied to statistical parsing (Johnson et al., 1999; Collins, 2000; Collins and Duffy, 2001). In discriminative estimation, only the conditional relation of an analysis given an example is considered relevant, whereas in maximum likelihood estimation the joint probability of the training data to best describe observations is maximized. Since the discriminative task is directly kept in mind during estimation, discriminative methods can yield improved performance. In our case, discriminative criteria cannot be defined directly with respect to "correct labels" or "gold standard" parses since the WSJ annotations are not sufficient to disambiguate the more complex LFG parses. However, instead of retreating to unsupervised estimation techniques or creating small LFG treebanks by hand, we use the labeled bracketing of the WSJ training sections to guide discriminative estimation. That is, discriminative criteria are defined with respect to the *set of parses consistent with the WSJ annotations*[1].

The objective function in our approach, denoted by $P(\boldsymbol{\lambda})$, is the joint of the negative log-likelihood $-L(\boldsymbol{\lambda})$ and a Gaussian regularization term $-G(\boldsymbol{\lambda})$ on the parameters $\boldsymbol{\lambda}$. Let $\{(y_j, z_j)\}_{j=1}^m$ be a set of training data, consisting of pairs of sentences y and partial annotations z, let $X(y, z)$ be the set of parses for sentence y consistent with annotation z, and $X(y)$ be the set of all parses produced by the grammar for sentence y. Furthermore, let $p[f]$ denote the expectation of function f under distribution p. Then $P(\boldsymbol{\lambda})$ can be defined for a conditional exponential model $p_{\boldsymbol{\lambda}}(z|y)$ as:

$$
\begin{aligned}
P(\boldsymbol{\lambda}) &= -L(\boldsymbol{\lambda}) - G(\boldsymbol{\lambda}) \\
&= -\log \prod_{j=1}^{m} p_{\boldsymbol{\lambda}}(z_j|y_j) + \sum_{i=1}^{n} \frac{\lambda_i^2}{2\sigma_i^2} \\
&= -\sum_{j=1}^{m} \log \frac{\sum_{X(y_j,z_j)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)}}{\sum_{X(y_j)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)}} + \sum_{i=1}^{n} \frac{\lambda_i^2}{2\sigma_i^2} \\
&= -\sum_{j=1}^{m} \log \sum_{X(y_j,z_j)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)} \\
&\quad + \sum_{j=1}^{m} \log \sum_{X(y_j)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)} + \sum_{i=1}^{n} \frac{\lambda_i^2}{2\sigma_i^2}.
\end{aligned}
$$

Intuitively, the goal of estimation is to find model parameters which make the two expectations in the last equation equal, i.e. which adjust the model parameters to put all the weight on the parses consistent with the partial annotation, modulo a penalty term from the Gaussian prior for too large or too small weights.

Since a closed form solution for such parameters is not available, numerical optimization methods have to be used. In our experiments, we adapted a conjugate gradient routine to our task (see Press (1992)), yielding a fast converging optimization algorithm where at each iteration the negative log-likelihood $P(\boldsymbol{\lambda})$ and the gradient vector have to be evaluated.[2]. For our task the gradient takes the form:

$$
\nabla P(\boldsymbol{\lambda}) = \left\langle \frac{\partial P(\boldsymbol{\lambda})}{\partial \lambda_1}, \frac{\partial P(\boldsymbol{\lambda})}{\partial \lambda_2}, \ldots, \frac{\partial P(\boldsymbol{\lambda})}{\partial \lambda_n} \right\rangle, \text{ and}
$$

$$
\begin{aligned}
\frac{\partial P(\boldsymbol{\lambda})}{\partial \lambda_i} &= -\sum_{j=1}^{m} \Big(\sum_{x \in X(y_j,z_j)} \frac{e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)} f_i(x)}{\sum_{x \in X(y_j,z_j)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)}} \\
&\quad - \sum_{x \in X(y_j)} \frac{e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)} f_i(x)}{\sum_{x \in X(y_j)} e^{\boldsymbol{\lambda}\cdot\boldsymbol{f}(x)}} \Big) + \frac{\lambda_i}{\sigma_i^2}.
\end{aligned}
$$

The derivatives in the gradient vector intuitively are again just a difference of two expectations

$$
-\sum_{j=1}^{m} p_{\boldsymbol{\lambda}}[f_i|y_j, z_j] + \sum_{j=1}^{m} p_{\boldsymbol{\lambda}}[f_i|y_j] + \frac{\lambda_i}{\sigma_i^2}.
$$

Note also that this expression shares many common terms with the likelihood function, suggesting an efficient implementation of the optimization routine.

4. Experimental Evaluation

Training: The basic training data for our experiments are sections 02-21 of the WSJ treebank. As a first step, all sections were parsed, and the packed parse forests unpacked and stored. For discriminative estimation, this data set was restricted to sentences which receive a full parse (in contrast to a FRAGMENT or SKIMMED parse) for both its partially labeled and its unlabeled variant. Furthermore, only sentences which received at most 1,000 parses were

[1] An earlier approach using partially labeled data for estimating stochastics parsers is Pereira and Schabes (1992) work on training PCFG from partially bracketed data. Their approach differs from the one we use here in that Pereira and Schabes take an EM-based approach maximizing the joint likelihood of the parses and strings of their training data, while we maximize the conditional likelihood of the sets of parses given the corresponding strings in a discriminative estimation setting.

[2] An alternative numerical method would be a combination of iterative scaling techniques with a conditional EM algorithm (Jebara and Pentland, 1998) However, it has been shown experimentally that conjugate gradient techniques can outperform iterative scaling techniques by far in running time (Minka, 2001).

taken under consideration. From this set, sentences from which a discriminative learner cannot possibly take advantage, i.e. sentences where the set of parses assigned to the partially labeled string was not a proper subset of the parses assigned the unlabeled string, were removed. These successive selection steps resulted in a final training set consisting of 10,000 sentence each with parses for partially labeled and unlabeled versions. Altogether there were 150,000 parses for partially labeled input and 500,000 for unlabeled input.

For estimation, a simple property selection procedure was applied to the full set of around 1000 properties. This procedure is based on a frequency cutoff on instantiations of properties for the parses in the labeled training set. The result of this procedure is a reduction of the property vector to about half of its size. Furthermore, a held-out data set was created from section 24 of the WSJ treebank for experimental selection of the variance parameter of the prior distribution. This set consists of 150 sentences which received only full parses, out of which the most plausible one was selected by manual inspection.

Testing: Two different sets of test data were used: (i) 700 sentences randomly extracted from section 23 of the WSJ treebank and given gold-standard f-structure annotations according to our LFG scheme, and (ii) 500 sentences from the Brown corpus given gold standard annotations by Carroll et al. (1999) according to their dependency relations (DR) scheme[3]. Both the LFG and DR annotation schemes are discussed in more detail below, as is a mapping from LFG f-structures to DR annotations.

Gold standard annotation of the WSJ test set was bootstrapped by parsing the test sentences using the LFG grammar and also checking for consistency with the Penn Treebank annotation. Starting from the (sometimes fragmentary) parser analyses and the Treebank annotations, gold standard parses were created by manual corrections and extensions of the LFG parses. Manual corrections were necessary in about half of the cases.

Performance on the LFG-annotated WSJ test set was measured using both the LFG and DR metrics, thanks to the LFG-to-DR annotation mapping. Performance on the DR-annotated Brown test set was only measured using the DR metric, owing to the absence of an inverse map from DR to LFG annotations.

Results: In our evaluation we report F-measures for the respective types of annotation, LFG or DR, and for three types of parse selection, (i) *lower bound*: random choice of a parse from the set of analyses, (ii) *upper bound*: selection of the parse with the best F-measure according to the annotation scheme used, and (iii) *stochastic*: the parse selected by the stochastic disambiguator. The *error reduction* row lists the reduction in error rate relative to the upper and lower bounds obtained by the stochastic disambiguation model. F-measures is defined as $2 \times precision \times recall/(precision + recall)$.

[3]Both corpora are available online. The WSJ f-structure bank at `www.parc.com/istl/groups/nltt/fsbank/`, and Carroll et al.'s corpus at `www.cogs.susx.ac.uk/lab/nlp/carroll/greval.html`.

Table 1 gives results for 700 examples randomly selected from section 23 of the WSJ treebank, using both LFG and DR measures. The effect of the quality of the parses on disambiguation performance can be illustrated by breaking down the F-measures according to whether the parser yields full parses or FRAGMENT or SKIMMED parses or both for the test sentences. The percentages of test examples which belong to the respective classes of quality are listed in the first row of Table 2. F-measures broken down according to classes of parse quality are recorded in the following rows. The first column shows F-measures for all parses in the test set, as in Table 1, the second column shows best F-measures when restricting attention to examples which receive only full parses. The third column reports F-measurs for examples which receive only non-full parses, i.e., FRAGMENT or SKIMMED parses or SKIMMED FRAGMENT parses. Columns 4–6 break down non-full parses according to examples which receive only FRAGMENT, only SKIMMED, or only SKIMMED FRAGMENT parses. Since most results on predicate-argument matching have been reported for length-restricted test sets (20–30 words), we also provide for comparison results for a subset of 500 sentences in our sample which had less than 25 words. These results are reported in Table 3.

Table 1: Disambiguation results for 700 examples randomly selected from section 23 of the WSJ treebank using LFG and DR measures.

	LFG	DR
upper bound	84.7	80.7
stochastic	78.7	72.9
lower bound	75.0	68.8
error reduction	38	35

Table 3: Disambiguation results on 500 examples restricted to < 25 words randomly selected from section 23 of the WSJ treebank using LFG and DR measures.

	LFG	DR
upper bound	88.0	85.4
stochastic	82.8	77.5
lower bound	78.0	72.6
error reduction	42	38

Results of the evaluation on Carroll et al.'s Brown test set are given in Tables 4 and 5. Table 4 presents an analysis of evaluation results according to parse-quality for the DR measure applied to the Brown corpus test set. In Table 5 we show the DR measure along with an evaluation measure which facilitates a direct comparison of our results to those of Carroll et al. (1999). Following Carroll et al. (1999) we count a depedency relation as correct if the gold standard has a relation with the same governor and dependent but perhaps with a different relation-type. This dependency-only (DO) measure thus does not reflect mismatches be-

Table 2: LFG F-measures broken down according to parse quality for the 700 WSJ test examples.

	all	full	non-full	fragments	skimmed	skimmed fragments
% of test set	100	74.7	25.3	20.4	1.4	3.4
upper bound	84.7	91.3	69.8	72.0	73.1	60.5
stochastic	78.8	84.6	65.2	67.4	67.8	55.9
lower bound	75.0	80.1	63.9	65.9	66.2	55.3

Table 4: DR F-measures broken down according to parse quality for the 500 Brown test examples.

	all	full	non-full	fragments	skimmed	skimmed fragments
% of test set	100	79.6	20.4	20.0	2.0	1.6
upper bound	79.6	84.0	65.2	65.2	55.5	52.9
stochastic	73.7	77.6	61.1	61.0	52.3	49.4
lower bound	70.8	74.4	58.8	58.7	50.8	48.3

tween arguments and modifiers in a small number of cases.

Table 5: Disambiguation results on 500 Brown corpus examples using DO measure and DR measures.

	DO	DR
upper bound	81.6	79.6
stochastic	75.8	73.7
lower bound	72.9	70.8
error reduction	33	34

5. Comparison of Evaluation Metrics

Tables 1 and 3 point to systematically lower F-scores under the DR measure than under the LFG measure, though both indicate similar reductions in error rate due to stochastic disambiguation.

5.1. LFG Evaluation Metric

The LFG evaluation metric is based on the comparison of 'preds-only' f-structures. A preds-only f-structure is a subset of a full f-structure that strips out grammatical attributes (e.g. tense, case, number) that are not directly relevant to predicate-argument structure. More precisely, a preds-only f-structure removes all paths through the f-structure that do not end in a PRED attribute. Figures 1 and 2 illustrate the difference between the full and preds-only f-structures for one parse of the sentence *Meridian will pay a premium of $30.5 million to assume a deposit of $2 billion.* As this example shows, the preds-only f-structure lacks some semantically important information present in the full f-structure, e.g. the marking of future tense, the marking of a purpose clause, and the attribute showing that *a deposit* is an indefinite.

Figure 2 also shows the set of individual feature specifications that define the preds-only f-structure. The first property indicates that the f-structure denoted by `n0` has the semantic form `sf(pay,i15,[n5,n3],[])` as the value of its PRED attribute. `pay` is the predicate, `i15` is a lexical id, `[n5,n3]` a list of f-structure nodes serving as thematic arguments, and `[]` an (empty) list of non-thematic arguments. The grammatical roles associated with thematic and non-thematic arguments are identified by the corresponding `subj`, `obj`, etc., predicates. In this experiment, we measured precision and recall by matching at the granularity of these individual features.

The matching algorithm attempts to find the maximum number of features that can be matched between two structures. It proceeds in a stratified manner, first maximizing the matches between attributes like `pred`, `adjunct` and `in_set`, and then maximizing the matches of any remaining attributes.

5.2. Comparison with DR Metric

As a brief review (see Carroll et al. (1999) for more detail), the DR annotation for our example sentence (obtained via the mapping described below) is

(aux _ pay will)	(subj pay Meridian _)
(detmod _ premium a)	(mod _ million 30.5)
(mod _ $ million)	(mod of premium $)
(dobj pay premium _)	(mod _ billion 2)
(mod _ $ billion)	(mod in $ deposit)
(dobj assume $ _)	(mod to pay assume)

Some obvious points of comparison with the f-structure features are: (i) The DR annotation encodes some information, e.g. the 'detmod' relation, that is not encoded in preds-only f-structures (though it is encoded in full f-structures). (ii) Different occurrences of the same word (e.g. "$") are distinguished via different lexical ids in the LFG representation but not in the DR annotations so that correctly matching DR relations can be problematic. (iii) The DR annotation has 12 relations instead of the 34 feature-specifications. This is because a given predicate-argument relation in the f-structure is broken down into several different feature-specifications. For example, the DR 'mod' relation involves an f-structure path through an ADJUNCT, IN_SET and two PRED attributes; the DR 'subj' relation is a combination of an f-structure PRED and SUBJ attribute. Thus the LFG metric is more sensitive to fine-grained aspects of predicate-

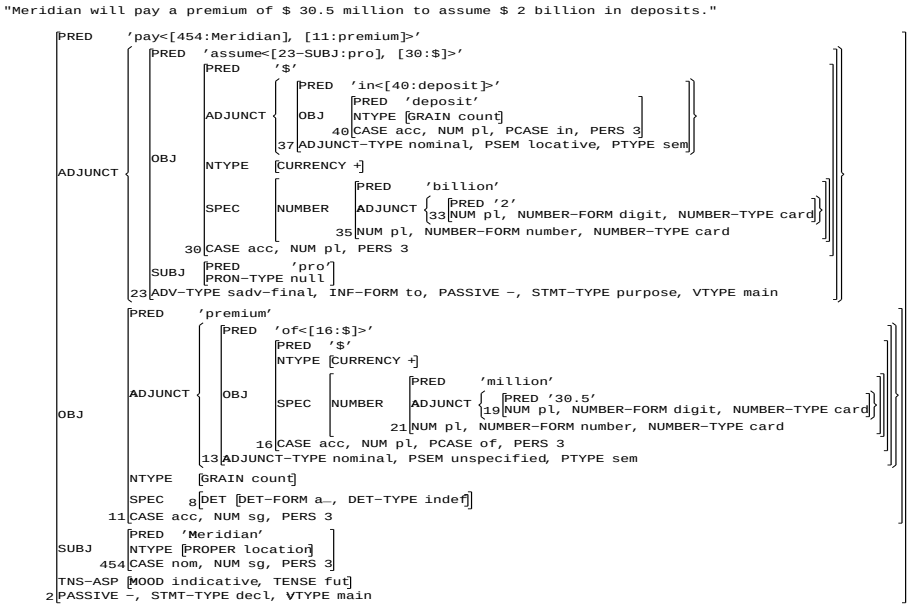
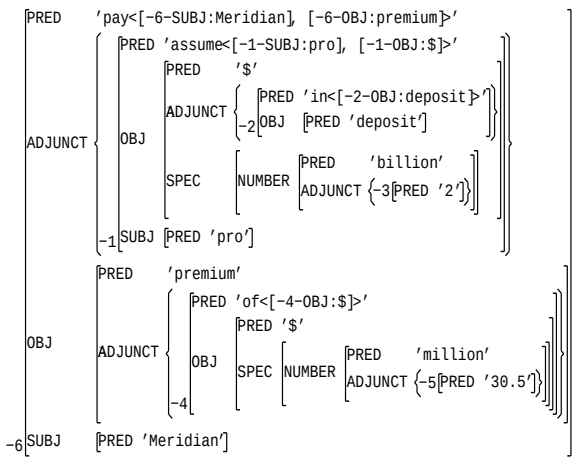


Figure 1: Full f-structure



pred(n0,sf(pay,i15,[n5,n3],[])) pred(n5,sf(Meridian,il4,[],[]))
pred(n3,sf(premium,i18,[],[])) pred(n19,sf('2',i70,[],[]))
pred(n28,sf('30.5',i26,[],[])) pred(n7,sf(assume,i64,[n8,n9],[]))
pred(n8,sf(pro,i107,[],[])) pred(n9,sf($,i67,[],[]))
pred(n17,sf(billion,i71,[],[])) pred(n11,sf(in,i84,[n12],[]))
pred(n12,sf(deposit,i86,[],[])) pred(n4,sf(million,i27,[],[]))
pred(n23,sf(of,i21,[n24],[])) pred(n24,sf($,i23,[],[]))
adjunct(n0,n2) in_set(n7,n2)
adjunct(n9,n14) in_set(n11,n14)
adjunct(n17,n18) in_set(n19,n18)
adjunct(n3,n20) in_set(n23,n20)
adjunct(n4,n31) in_set(n28,n31)
subj(n0,n5) subj(n7,n8)
obj(n0,n3) obj(n7,n9) obj(n11,n12) obj(n23,n24)
number(n16,n17) number(n26,n4) spec(n9,n16) spec(n24,n26)

Figure 2: Preds-only f-structure: graphical & clausal representation as produced by XLE

argument relations. However, it imposes a greater penalty than DR on a modifier that is misattached to something that does not have any other modifiers. The LFG measure counts both an extra ADJUNCT feature and an extra IN_SET feature as mismatches, whereas DR only counts a single mismatched MOD. Conversely, LFG gives more credit for getting the singleton attachments correct. Similarly for argument structure. The LFG metric penalizes getting arguments wrong, counting both a PRED and a grammatical relation mismatch, but conversely gives more credit if the argument structure is exactly right.

5.3. Mapping F-structures to DR Annotations

The DR evaluation metric matches the dependency relations provided by the Carroll et al. gold standard with relations determined from information contained in the LFG representations. This enables us to measure the accuracy of our system with a separately defined predicate-argument-oriented standard and to compare our results to other systems that may use the same metric (at this point, perhaps only the Carroll et al. grammar/parser). The DR metric also enables a cross-validation assessment of the LFG-derived predicate-argument measure.

Carroll and Briscoe provide conveniently downloadable files containing the raw input sentences and the corresponding sets of gold standard dependency relations. We assumed it would be relatively straightforward to run the sentences through our system and extract dependency relations that could be compared to the gold standard. But for reasons that ranged from the ridiculous to the sublime, this turned out to be a surprisingly difficult task. One of the lessons learned from this experiment is that even at the level of abstract dependencies it is still very hard to create a standard that does not incorporate unintended framework-specific idiosyncracies.

One set of problems arose from the way the sentences are recorded in the input file. The 'raw' sentences are not formed as they would appear in natural text. They are pro-

vided instead as pre-tokenized strings, with punctuation split off by spaces from surrounding words. Thus commas and periods stand as separate tokens and *I'm* and *clients' guilt* show up as *I 'm* and *clients ' guilt*. This preprocessed format may be helpful for parsing systems that embody this particular set of tokenizing conventions or that learn (a la tree bank grammars) from the data at hand. But our system includes a hand-written finite-state tokenizer that is tightly integrated with our grammar and lexicon, and it is designed to operate on text that conforms to normal typographical conventions. It provides less accurate guesses when text is ill-formed in this way, for example, introducing an ambiguity as to whether the quote in *clients ' guilt* is attached as a genitive marker to the left or as an open quote to the right. Another peculiar and troublesome feature of the raw text is that some non-linguistic elements such as chemical formulas are replaced by the meta-symbol <*formul*>; our tokenizer splits this up at the angle brackets and tries to guess a meaning for the word *formul* surrounded by brackets. Faced with these low-level peculiarities, our first step in the evaluation was to edit the raw text as best we could back into normal English.

The gold standard file presented another set of relatively low-level incompatibilities that resulted in spurious mismatches that were somewhat harder to deal with. First, the input sentences conform to American spelling conventions but the head-words in the gold standard relations use British spelling (*neighbor* is coded as *neighbour*). Second, in the gold standard the head-words are converted to their citation forms (e.g. "*walking*" in the text appears as *walk* in the relations). Generally these match the head-words that are easily read from the LFG f-structures, but there are many discrepancies that had to be tracked down. For example, our f-structures do not convert *should* to *shall*, as the gold standard does, whereas we do convert *himself* to *he* (with a reflexive feature) while the gold standard leaves it as *himself*. We ended up creating by trial-and-error a coercion table for this test set so that we could properly match different manifestations of the same head.

The experiment revealed some higher-level conceptual issues. In LFG it is the f-structure rather than the c-structure that most closely encodes the properties on which a non-tree, dependency-oriented evaluation should be based. So we defined our task to be the construction of a routine for reading dependencies from the f-structure alone. It turns out, however, that the Carroll et al. dependencies encode a mixture of superficial phrase-structure properties in addition to underlying dependencies, and it proved a challenge to recreate all the information relevant to a match from the f-structure alone. For example, our f-structures do not represent the categories (NP, S) of the phrases that correspond to the functions, but the gold standard dependencies make tree-based distinctions between non-clausal (e.g. NP) subjects, clausal (e.g. sentential) subjects, and open-complement (VP) subjects. We avoided this kind of discrepancy by neutralizing these distinctions in the gold standard prior to making any comparisons. As another example, our English grammar decodes English auxiliary sequences into features such as PERFECT, PROGRESSIVE, and PASSIVE while the gold standard provides a set of AUX relations that represent the left-to-right order in which *have* and *be* appeared in the original sentence. To obtain the intuitively correct matches, our mapping routine in effect had to simulate a small part of an English generator that decodes our features into their typical left-to-right ordering. In at least one case we simply gave up—it was too hard to figure out under which conditions there might have been do-support in the original string; instead, we removed the few aux-do relations from the gold standard before comparing.

There were a number of situations where it was difficult to determine exactly the gold standard coding conventions either from the documentation or from the examples in the gold standard file. Some of the confusions were resolved by personal communication with Carroll and Briscoe, leading in some cases to the correction of errors in the standard or to the clarification of principles. We discovered for some phenomena that there were simple differences of opinion of how a relation should be annotated. The corpus contains many parentheticals, for example, whose proper attachment is generally determined by extrasyntactic, discourse-level considerations. The default in the LFG grammar is to associate parentheticals at the clause-level whereas the Carroll-Briscoe gold standard tends to associate them with the constituent immediately to the left—a constituent that we cannot identify from the f-structure alone. As other examples, there are still some mysteries about whether and how unexpressed subjects of open-complements are to be encoded and whether and how the head of a relative clause appears in a within-clause dependency.

With considerable effort we solved most but not all of these cross-representation mapping problems, as attested by the relatively high F-scores we have reported. Our current results probably understate to a certain extent our true degree of matching, but the relative differences between sentences using the DR measure are quite informative. A low F-score is an accurate indication that we did not obtain the correct parse. For F-scores above 90 but below 100 it is often the case that we found exactly the right parse but our mapping routine could not produce all the proper relations.

6. Discussion

The general conclusion to draw from our results is that the two metrics, LFG and DR, show broadly similar behavior, for the upper bounds, for the lower bounds, and for the reduction in error relative to the upper bound brought about by the stochastic model. The correlation between the upper bound F-scores for the LFG and DR measures on the WSJ test set is .89. The lower reduction in error rate relative to the upper bound for DR evaluation on the Brown corpus can be attributed to a corpus effect that has also been observed by Gildea (2001) for training and testing PCFGs on the WSJ and Brown corpora.[4] Breaking down evaluation results according to parse quality shows that irrespective of evaluation measure and corpus around 5% overall per-

[4]Gildea reports a decrease from 86.1%/86.6% recall/precision on labeled bracketing to 80.3%/81% when going from training and testing on the WSJ to training on the WSJ and testing on the Brown corpus.

formance is lost due to non-full parses, i.e. FRAGMENT or SKIMMED parses or both.

While disambiguation performance of around 79% F-score on WSJ data seems promising, from one perspective it only offers a 4% absolute improvement over a lower bound random baseline. We think that the high lower bound measure highlights an important aspect of symbolic constraint-based grammars (in contrast to treebank grammars): the symbolic grammar already significantly restricts/disambiguates the range of possible analyses, giving the disambiguator a much narrower window in which to operate. As such, it is more appropriate to assess the disambiguator in terms of reduction in error rate (38% relative to the upper bound) than in terms of absolute F-score. Both the DR and LFG annotations broadly agree in their measure of error reduction.

Due to the lack of standard evaluation measures and gold standards for predicate-argument matching, a comparison of our results to other stochastic parsing systems is difficult at the moment. To our knowledge so far the only direct point of comparison is the parser of Carroll et al. (1999) which is also evaluated on Carroll et al.'s test corpus. They report an F-measure of 75.1% for a DO evaluation that ignores predicate labels but counts dependencies only. Under this measure, our system of parser and stochastic disambiguator achieves 75.8% F-measure. A further point of comparison is the parsing system presented by Bouma et al. (2000). They report comparable relations on lower bounds and upper bounds for their constraint-based parsing systems. On test corpora of a few hundred sentences of up to 20 words an upper bound of 83.7% F-score and a lower bound of 59% is reported; the best disambiguation models achieves 75% F-score.

7. References

Gosse Bouma, Gertjan von Noord, and Robert Malouf. 2000. Alpino: Wide-coverage computational analysis of Dutch. In *Proceedings of Computational Linguistics in the Netherlands*, Amsterdam, Netherlands.

Miriam Butt, Tracy King, Maria-Eugenia Niño, and Frédérique Segond. 1999. *A Grammar Writer's Cookbook*. Number 95 in CSLI Lecture Notes. CSLI Publications, Stanford, CA.

John Carroll, Guido Minnen, and Ted Briscoe. 1999. Corpus annotation for parser evaluation. In *Proceedings of the EACL workshop on Linguistically Interpreted Corpora (LINC)*, Bergen, Norway.

Michael Collins and Nigel Duffy. 2001. Convolution kernels for natural language. In *Advances in Neural Information Processing Systems 14(NIPS'01)*, Vancouver.

Michael Collins. 2000. Discriminative reranking for natural language processing. In *Proceedings of the Seventeenth International Conference on Machine Learning (ICML'00)*, Stanford, CA.

Dan Gildea. 2001. Corpus variation and parser performance. In *Proceedings of 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Pittsburgh, PA.

Tony Jebara and Alex Pentland. 1998. Maximum conditional likelihood via bound maximization and the CEM algorithm. In *Advances in Neural Information Processing Systems 11 (NIPS'98)*.

Mark Johnson, Stuart Geman, Stephen Canon, Zhiyi Chi, and Stefan Riezler. 1999. Estimators for stochastic "unification-based" grammars. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics (ACL'99)*, College Park, MD.

Mitchell Marcus, Grace Kim, Mary Ann Marcinkiewicz, Robert MacIntyre, Ann Bies, Mark Ferguson, Karen Katz, and Britta Schasberger. 1994. The Penn treebank: Annotating predicate argument structure. In *ARPA Human Language Technology Workshop*.

John Maxwell and Ron Kaplan. 1993. The interface between phrasal and functional constraints. *Computational Linguistics*, 19(4):571–589.

Thomas Minka. 2001. Algorithms for maximum-likelihood logistic regression. Department of Statistics, Carnegie Mellon University.

Miles Osborne. 2000. Estimation of stochastic attribute-value grammars using an informative sample. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, Saarbrücken.

Fernando Pereira and Yves Schabes. 1992. Inside-outside reestimation from partially bracketed corpora. In *Proceedings of the 30th Annual Meeting of the Association for Computational Linguistics (ACL'92)*, Newark, Delaware.

William H. Press, Saul A. Teukolsky, Willam T. Vetterling, and Brian P. Flannery. 1992. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, New York.

Stefan Riezler, Detlef Prescher, Jonas Kuhn, and Mark Johnson. 2000. Lexicalized Stochastic Modeling of Constraint-Based Grammars using Log-Linear Measures and EM Training. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics (ACL'00)*, Hong Kong.