ULPGC Universidad de Las Palmas de Gran Canaria | ESCUELA DE INGENIERÍA INFORMÁTICA

Trabajo de Fin de Grado

Comunicación política en redes sociales: análisis de popularidad y sentimientos en X/Twitter mediante ciencia de datos

TITULACIÓN: Grado en Ciencia e Ingeniería de Datos

AUTOR: Jaime Ballesteros Domínguez

TUTORIZADO POR:
José Miguel Santos Espino

Fecha 07/2025

ULPGC Universidad de Las Palmas de Gran Canaria | EII ESCUELA DE INGENIERÍA INFORMÁTICA

TFT04

SOLICITUD DE DEFENSA DE TRABAJO DE FIN DE TÍTULO

D./Dª Jaime Ballesteros Domínguez, autor del trabajo Comunicación política en redes sociales: análisis de popularidad y sentimientos en X/Twitter mediante ciencia de datos, correspondiente a la titulación Grado en Ciencia e Ingeniería de datos.

SOLICITA

que se inicie el procedimiento de defensa del mismo, para lo que se adjunta la documentación requerida, haciendo constar que

[X] se autoriza / [ ] no se autoriza la grabación en audio de la exposición y turno de preguntas.

Asimismo, con respecto al registro de la propiedad intelectual/industrial del TFT, declara que:

[ ] Se ha iniciado o hay intención de iniciarlo (defensa no pública).

[X] No está previsto.

Y para que así conste firma la presente. (fecha en firma electrónica)

El/La estudiante

Fdo.————————

A rellenar y firmar **obligatoriamente** por el/la/los/las tutores

En relación con la presente solicitud, se informa:(firmar donde corresponda)

Positivamente (en caso de detección de copia, esta firma quedará invalidada)	Negativamente (justificación en TFT05)
Fdo.————————	Fdo.————————

DIRECTOR DE LA ESCUELA DE INGENIERÍA INFORMÁTICA

Agradecimientos

En primer lugar, quiero expresar mi profundo agradecimiento a mi familia y amigos, por su apoyo incondicional a lo largo de este proceso. A mi tutor, José Miguel Santos Espino, gracias por sus valiosas sugerencias de mejora y por brindarme siempre una nueva perspectiva. Finalmente, extiendo mi agradecimiento a la Escuela de Ingeniería Informática de la Universidad de Las Palmas de Gran Canaria y a todo el profesorado del Grado en Ciencia e Ingeniería de Datos, por proporcionarme las herramientas y conocimientos necesarios para dar forma y culminar con éxito este proyecto.

Resumen

Este Trabajo Fin de Título aplica técnicas de ciencia de datos y procesamiento del lenguaje natural para analizar la comunicación de políticos españoles en la red social X (anteriormente Twitter). El estudio incluye la recopilación, limpieza, traducción automática y etiquetado de mensajes y comentarios, así como el reentrenamiento de modelos BERT para clasificar tanto el tono como la temática de los contenidos. El análisis final, complementado con visualizaciones, permitió identificar patrones en el estilo del discurso, en las respuestas ciudadanas y en las diferencias entre partidos o regiones. Este trabajo combina enfoques técnicos y sociales con el objetivo de contribuir a una comprensión más profunda, cuantitativa y matizada del discurso político digital en lengua española.

Abstract

This Final Degree Project applies data science and natural language processing techniques to analyze the communication of Spanish politicians on the social network X (formerly Twitter). The study includes the collection, cleaning, automatic translation, and annotation of posts and comments, as well as the fine-tuning of BERT models to classify both the tone and topic of the content. The final analysis, complemented by visualizations, enabled the identification of patterns in discourse style, citizen responses, and differences between parties or regions. This work combines technical and social approaches with the aim of contributing to a deeper, quantitative, and more nuanced understanding of Spanish-language digital political discourse.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

En los últimos años, las redes sociales se han consolidado como uno de los principales canales de comunicación política junto a medios más tradicionales como la radio o la televisión. Plataformas como Twitter, recientemente renombrada como X, permiten a los representantes públicos establecer un diálogo rápido y directo con la ciudadanía, lo que ha generado nuevas formas de interacción y participación política. Este entorno digital, caracterizado por su inmediatez, plantea nuevas dudas sobre los estilos comunicativos, la recepción de los mensajes y las diferencias en función de factores como el contexto geográfico o la orientación política.

El uso de redes sociales y de dispositivos móviles como fuentes principales de información ha superado ampliamente al de los medios tradicionales. Así, tal como se observa en la Ilustración 1.1, el uso de Internet en móviles, redes sociales y ordenadores supera a día de hoy el 90 % en España, mientras que otros canales como la televisión, la radio o la prensa impresa presentan porcentajes de uso considerablemente más bajos.

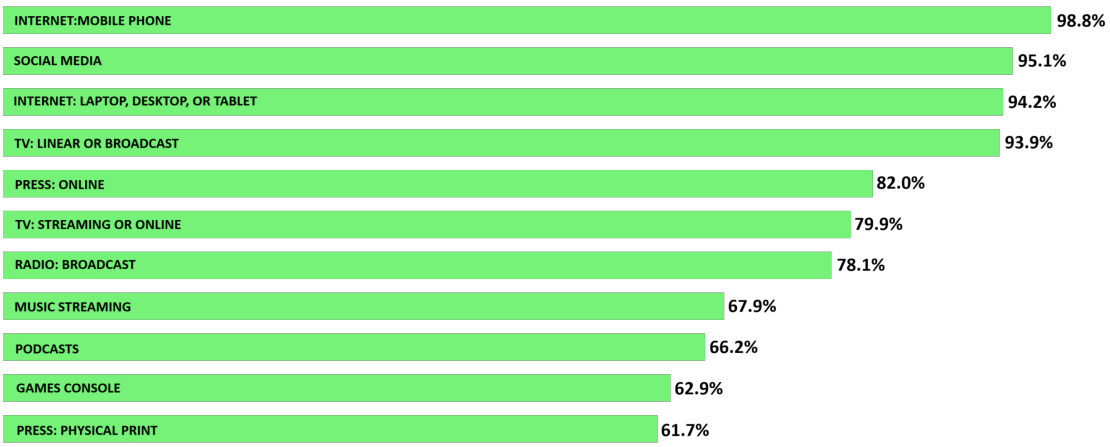

Ilustración 1.1: Porcentaje de personas en España que consumen cada tipo de contenido digital. Fuente: DataReportal, 2025.

No obstante, los medios tradicionales no han perdido del todo su influencia en la comunicación política. Según la II Encuesta Nacional de Polarización Política (CEMOP, 2022), la televisión sigue siendo el medio más utilizado para informarse sobre política: un 51,7 % de

los ciudadanos declara verla diariamente con esa finalidad, frente a un 37,2 % que recurre a redes sociales y un 35,1 % que consulta periódicos digitales [13].

En este contexto, una encuesta realizada en la Unión Europea en 2018 mostró que aproximadamente el 60 % de la población afirmaba que las redes sociales podían aumentar el interés en la política [44]. Esto refuerza la idea de que plataformas como X/Twitter tienen un papel relevante en la configuración de la agenda pública, facilitando el acceso a información institucional y amplificando el impacto de los discursos políticos.

El presente trabajo se desarrolla en el ámbito de la ciencia de datos y tiene como objetivo analizar el comportamiento comunicativo de los políticos españoles, tanto a nivel autonómico como nacional, en redes sociales. Mediante técnicas de procesamiento de lenguaje natural (PLN), se evaluarán distintos aspectos de su actividad en X/Twitter: popularidad e interacción de los mensajes, tono del discurso, temáticas predominantes y respuesta de la audiencia. El PLN es un área de investigación y aplicación que estudia cómo los ordenadores pueden entender y manipular texto o habla en lenguaje natural para realizar tareas útiles [14]. Esta disciplina resulta especialmente útil para analizar grandes volúmenes de datos lingüísticos, como los generados en redes sociales, y extraer de ellos información relevante para la interpretación del discurso político. Asimismo, se estudiará la influencia de variables como el partido político, la edad, el género o la comunidad autónoma de la que proviene o representa el emisor.

Para contextualizar este análisis, resulta clave considerar el perfil demográfico de los usuarios de X/Twitter. Así, en 2025, la plataforma contaba con 10,4 millones de usuarios en España, lo que representa el 21,7 % de la población total y el 25,5 % de los adultos mayores de 18 años [45]. Sin embargo, dicha base de usuarios no es homogénea: el 65,5 % son hombres y solo el 34,5 % mujeres. Si se traslada este análisis a nivel mundial, tal y como se aprecia en la Ilustración 1.2, predominan los hombres jóvenes: los varones de entre 25 y 34 años constituyen el 24,5 % de la audiencia publicitaria, seguidos por los de entre 18 y 24 años (18,9 %) [15].

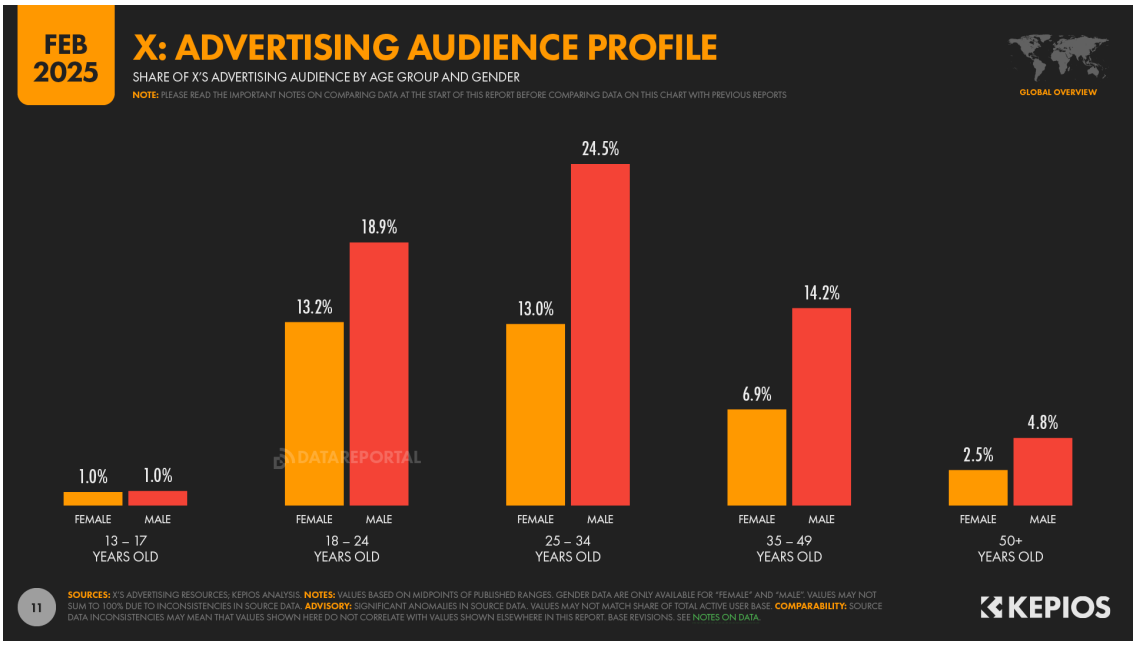

Ilustración 1.2: Distribución por edad y género de la audiencia publicitaria de X/Twitter a nivel mundial. Fuente: DataReportal, febrero de 2025.

Es importante considerar estas diferencias de representación entre grupos de edad y género, ya que pueden condicionar la forma en la que se recibe e interpreta el discurso político. De hecho, investigaciones recientes sobre campañas electorales han revelado que la percepción de interacción por parte de los partidos varía según el género, la edad y el lugar de residencia de los usuarios. Por ejemplo, en X/Twitter, las mujeres jóvenes (18–34 años) y los residentes de las capitales de provincia son quienes más perciben que los partidos buscan interactuar con ellos [7]. Además, estudios actuales han detectado una brecha de género creciente en el voto juvenil, con un mayor apoyo a partidos de extrema derecha entre varones jóvenes, fenómeno que se ha asociado a los hábitos de uso y exposición en redes sociales [38]. Estas observaciones sugieren que tanto el contenido como la forma de los mensajes pueden estar estratégicamente adaptados a audiencias específicas.

Estos factores refuerzan la importancia de estudiar la comunicación política en redes sociales considerando variables demográficas y contextuales. Para ello, este proyecto sigue el enfoque CRISP-DM, que se desarrollará en las siguientes fases:

1. **Comprensión del negocio y de los datos:** selección de políticos activos en X/Twitter a nivel nacional y autonómico; recopilación de sus publicaciones y metadatos relevantes (fecha, tipo de publicación, interacción, etc.).

2. **Preparación de los datos:** limpieza, normalización del texto y etiquetado de datos, junto con el añadido de información de contexto (partido político, comunidad autónoma, etc.).

3. **Modelado:** aplicación de modelos de lenguaje como BERT en español para la clasificación del tono emocional de los mensajes y análisis de sentimiento.

4. **Evaluación:** análisis cruzado de resultados para identificar patrones según partido político, geografía o tipo de contenido, así como correlación entre tono, temática y nivel de interacción.
5. **Despliegue:** creación de visualizaciones interactivas mediante bibliotecas como Plotly o herramientas como Streamlit, con el fin de facilitar la exploración de los resultados por cualquier tipo de usuario que pueda tener o no conocimientos técnicos.

A pesar de los avances en la aplicación de técnicas de análisis de sentimientos, el estudio del discurso político en lengua española aún se encuentra en una etapa inicial, con un desarrollo considerablemente más lento que en inglés [17]. Esto se debe, entre otras razones, a la complejidad del idioma y a la falta de diccionarios de términos emocionales adecuados [21, 22]. Además, fenómenos como el discurso de odio tienen matices de significado particulares que exigen métodos de análisis diferentes [17]. A pesar de estas dificultades, la introducción de modelos de lenguaje como BERT ha abierto nuevas posibilidades para llevar a cabo el análisis de textos políticos complejos y sensibles.

Por tanto, la motivación principal de este trabajo es mostrar el potencial del análisis de sentimientos y del procesamiento del lenguaje natural en castellano, aplicándolo a un caso de gran relevancia social: la comunicación política de los representantes españoles en X/Twitter. Con ello se busca reducir la brecha entre los avances tecnológicos actuales y su aplicación real al análisis del discurso político en español.

A partir de esta motivación, surgen dos preguntas de investigación que guían el desarrollo del trabajo:

- **¿Existen diferencias significativas en el estilo, el tono o el nivel de interacción de los mensajes políticos según el partido o la comunidad autónoma del emisor?**
- **¿Qué relación existe entre el tono y la temática de los mensajes y las métricas de interacción obtenidas, así como el tono de las respuestas recibidas?**

Responder a estas cuestiones permitirá arrojar luz sobre cómo se adaptan los discursos políticos a distintos contextos ideológicos y geográficos, y cómo se generan dinámicas de recepción, aceptación o polarización en redes sociales.

Para dar respuesta a estos objetivos, este trabajo contribuye aplicando técnicas de análisis de sentimientos y procesamiento del lenguaje natural en español, analizando de forma conjunta el tono, los temas y la interacción en redes sociales. A diferencia de estudios centrados únicamente en contextos anglosajones o en objetivos específicos, esta investigación aborda varias dimensiones del discurso político digital, con un enfoque dirigido al entorno hispanohablante.

Además, este Trabajo de Fin de Título se alinea con estudios recientes que destacan el valor creciente de aplicar inteligencia artificial y minería de datos al análisis de redes sociales, tanto desde una perspectiva comercial como institucional [35]. Estos enfoques permiten detectar actitudes y reacciones del público con un alto nivel de detalle, reforzando la conveniencia de trasladar estas técnicas al análisis de la comunicación política en España.

Finalmente, para facilitar la comprensión del lector, este documento se estructura en seis capítulos. El Capítulo 1 introduce el contexto general del trabajo y su relevancia. Por su parte, el Capítulo 2 aborda el estado actual del campo, la motivación y los objetivos iniciales. El Capítulo 3 presenta las principales aportaciones, competencias desarrolladas y alineamiento con los Objetivos de Desarrollo Sostenible. Seguidamente, en el Capítulo 4 se describe la metodología aplicada y las herramientas utilizadas, así como el proceso de trabajo y resultados intermedios que se hayan podido obtener en cada paso. El Capítulo 5 expone los resultados obtenidos, mientras que el Capítulo 6 recoge las conclusiones y plantea líneas futuras de trabajo.

Capítulo 2

Estado actual y objetivos iniciales

2.1. Estado del arte

Desde la entrada al nuevo milenio se ha estado gestando una nueva forma de comunicación, de compartir, de opinar, de criticar y de relación entre personas: las redes sociales. Según un estudio de We Are Social [52], en el último año la cifra de usuarios en redes sociales se ha incrementado en 206 millones hasta llegar a los 5,24 mil millones de personas interconectadas, lo que equivale al 63,9 % de la población total del planeta. En específico, a febrero de 2025, España supera esta media mundial con un 82,9 % de su población o, visto de otra manera, casi 40 de los 48 millones de habitantes del país forman parte de este entorno digital.

Las redes sociales han supuesto un cambio en la velocidad y en la forma en la que la población recibe la información. Noticias que antes tardaban horas en procesarse y contrastarse previo a su publicación, actualmente llegan al público final de manera directa y, en muchos casos, altamente sesgada. Ahora más que nunca, es el consumidor de estas noticias el responsable de discernir la verdad y elaborar una opinión propia con respecto al incesante flujo de información proveniente de las redes.

Sin embargo, la idea de que los propios testigos de la noticia adopten un papel de informante ha servido para dar voz y poner caras a situaciones que pasarían desapercibidas en la prensa tradicional. A este respecto, como señalan Bonilla y Rosa [8], "antes de que los medios tradicionales se pusieran al día con lo que estaba ocurriendo, la masa de tweets con hashtags fue una forma de llamar la atención sobre un incidente poco informado."

Este creciente interés por las redes sociales es lo que ha motivado a la clase política a hacer un debate interno sobre el gran potencial informativo e influyente de esta nueva herramienta. Las redes sociales han transformado la forma en que las personas se informan, interactúan y participan en debates públicos. En el caso concreto de la política, el punto de partida del uso de las redes sociales para conectar con el electorado apunta a la primera década del 2000. Este periodo de tiempo coincide con las campañas electorales a la presidencia de Barack Obama. Tal y como señala Boulianne [10], "la Primavera Árabe en 2011, así como las campañas de Obama en 2008 y 2012, han impulsado el interés en cómo las redes sociales pueden afectar la participación ciudadana en la vida cívica y política." En España, el uso de redes sociales en política comenzó a cobrar importancia en el mismo periodo, reflejado en momentos clave como el debate televisado entre Alfredo Pérez Rubalcaba (PSOE) y Mariano Rajoy (PP) en

2011. Este fue considerado el primer debate donde las redes sociales tuvieron una presencia significativa en el desarrollo, difusión y generación de contenidos por parte de los espectadores [16].

Como señala Boulianne [10], la participación política en redes sociales no debe entenderse exclusivamente en términos de comportamiento electoral, sino también como una forma de implicación cívica o expresiva. Las interacciones digitales, como comentarios, retweets o menciones, constituyen una forma relevante de participación que puede influir en la formación de opinión y en la movilización ciudadana, aunque no necesariamente en el voto. Por ello, este estudio se centra en el análisis del discurso político y la respuesta del electorado como manifestaciones de participación digital.

En esta línea, Boulianne y Larsson [11] exploran cómo distintas características del contenido de los mensajes afectan el nivel de interacción en redes sociales. Su estudio revela que las publicaciones con tono negativo o ataques directos a adversarios políticos tienden a generar mayor cantidad de comentarios, "likes" y compartidos, en comparación con aquellas centradas en propuestas políticas o menciones neutras. Asimismo, encuentran que etiquetar a otros usuarios, lo cual es considerado como una forma explícita de interacción, puede disminuir el alcance y la participación, especialmente en X/Twitter. Estas evidencias refuerzan la idea de que la interacción digital no es neutra ni homogénea, sino que está fuertemente condicionada por el tipo de discurso empleado por los actores políticos.

Sin embargo, aunque las redes sociales fueron utilizadas para informar, su capacidad para influir en el electorado seguía siendo mínima. Así, Abejón, Sastre y Linares [1] sostienen que "los políticos españoles han entendido la importancia de estar en las redes, pero no han entendido su verdadero uso. El movimiento 15-M, a través de las redes sociales, monopolizó la información de los medios sobre la campaña, pero las redes no modificaron el comportamiento electoral." Esta reflexión se refuerza en otras partes del mismo estudio, donde los autores concluyen que "los políticos han entendido la importancia de estar en Internet, pero no han entendido su verdadero uso, ya que lo utilizan más como herramienta de campaña que para relacionarse con el ciudadano" [1]. Este enfoque unidireccional y de entender las redes sociales como una simple herramienta limita el potencial deliberativo de las redes, alejando su uso político de modelos más interactivos o participativos observados en otros contextos.

Esto llevó a la conclusión inicial de que, aunque fueran excelentes herramientas para hacerse escuchar, las redes sociales no tenían la capacidad de influir en procesos electorales. Sin embargo, los éxitos de las redes sociales en otras campañas como la de la Asamblea Nacional de Francia [12], el Movimiento 5 Estrellas en Italia [3] y el Partido Republicano con Donald Trump a la cabeza [46] pronto demostraron cómo de efectivas pueden llegar a ser las redes sociales y, sobre todo X/Twitter, si se usan de manera adecuada. Estas tres campañas fueron grandes exponentes de un discurso más populista, caracterizado por mensajes personales y sensacionalistas, el cual llegó al público en general gracias en gran medida a las redes sociales, las cuales sirvieron como un vínculo directo con el electorado, permitiéndoles eludir las trabas del periodismo tradicional [19]. Es importante destacar cómo ninguna de estas campañas fue llevada a cabo por el partido gobernante, reforzando los resultados de varios estudios en los que se concluye que los partidos en la oposición son más activos en X/Twitter que los partidos en el gobierno [28].

En este contexto, el análisis de Stolee y Caton [46] sobre la campaña de Donald Trump ayuda a comprender la efectividad de las redes sociales cuando se tiene un objetivo claro. Los autores argumentan que X/Twitter no solo funcionó como canal de difusión de mensajes, sino que permitió a Trump consolidar una forma de discurso político centrado exclusivamente en su base electoral, rompiendo con la tradición de dirigirse a amplios sectores del electorado. Esta estrategia comunicativa, caracterizada por un estilo directo y emocional, facilitó un mayor acercamiento con sus seguidores, los cuales valoraban más la autenticidad percibida frente al hecho de que dichos mensajes fuesen ciertos. Además, los autores destacan cómo las características principales de X/Twitter, como la inmediatez y la brevedad con la que un post se puede volver tendencia, contribuyeron a moldear una nueva forma de comunicación presidencial que favorece la viralidad sobre la deliberación.

La plataforma más utilizada para extender dicho mensaje ha sido, desde hace ya muchos años, X/Twitter, la cual se presentó como un espacio ideal en el cual, tal y como explican Weller et al. [53], "la sociedad puede y ha llegado a estar conectada de formas nunca antes imaginadas." Ya desde sus inicios, X/Twitter ha sido vista como una herramienta innovadora para la comunicación, destacando en el campo del microblogging, lo cual es una forma de comunicarse con el público mediante mensajes cortos [40], [53]. También, y al igual que otras plataformas, ha servido como medidor de efectividad para las campañas de imagen de los propios políticos. A este respecto, Díaz Cuervo y Barbera González [18] ilustran su crecimiento en la comunicación política con el caso de Obama, indicando que, frente a los 157 retweets que consiguió en sus primeras elecciones, la imagen publicada para celebrar su reelección logró superar los 800.000 retweets en menos de tres días.

Por estas razones, elementos más tradicionales en la comunicación política, como es el caso de los debates televisados o la propia cobertura periodística, han empezado a incorporar las reacciones del público a aquellos contenidos que generan reacciones en X/Twitter [28].

El éxito político en redes sociales y, especialmente en X/Twitter, viene dado por mensajes cortos, sensacionalistas y altamente personalizados. Dichas publicaciones se desmarcan de las grandes campañas propagandísticas, las cuales muchas veces fallan por no tener un mensaje claro o con declaraciones demasiado extendidas como para interesar al público objetivo, por discursos concretos, directos y emocionalmente impactantes, que buscan generar interacción inmediata y viralidad en la plataforma. Este tipo de mensajes apelan a la identidad del votante, refuerzan sus creencias y facilitan la movilización digital, diferenciándose de las estrategias tradicionales basadas en argumentaciones extensas y detalladas.

Este nuevo punto de vista deriva de la utilización del análisis de sentimientos en distintos estudios de la literatura, ya sea a nivel europeo [3] o nacional [42]. Sin embargo, la mayoría de estos estudios realizan el análisis de sentimientos de las publicaciones por partido político, como si se tratase de una entidad única e individual. Además, aún son escasos los trabajos que tratan el análisis de sentimientos y el discurso de odio en el lenguaje español, tal y como concluyen del Valle y de la Fuente [17].

La revisión de del Valle y de la Fuente [17] permite además identificar las principales metodologías empleadas en el análisis de sentimientos aplicado al discurso político en español. Entre estas destacan las técnicas basadas en reglas léxicas (como SentiStrength o AFINN),

los enfoques clásicos de aprendizaje automático (Naive Bayes, SVM) y, más recientemente, los modelos de lenguaje preentrenados como BERT y BETO. Estos últimos han demostrado un muy buen rendimiento en tareas de clasificación de tono y detección de emociones, especialmente en textos complejos o contextualmente cargados. No obstante, los autores subrayan que el PLN en español presenta limitaciones importantes, como la ambigüedad semántica, la escasa disponibilidad de ejemplos etiquetados para tareas políticas, y las dificultades añadidas por el uso de ironía, sarcasmo o eufemismos.

Por eso, con el fin de dar respuesta a esta problemática aún no estudiada en la literatura, en este trabajo se plantea abordar el estudio de las interacciones de los miembros de los partidos con su electorado de manera individual utilizando el análisis de sentimientos y del lenguaje. Esto permitirá entender las características comunicativas de cada político estudiado y cómo responde el electorado a dichos mensajes. También servirá para analizar la influencia de la directiva de un partido en sus miembros a la hora de presentar un mensaje.

2.2. Motivación y antecedentes

El uso de redes sociales como canal principal de comunicación política ha transformado el modo en que los representantes públicos interactúan con la ciudadanía. Esta transformación ha generado nuevas formas de participación y ha abierto un amplio campo de estudio sobre los efectos de esta interacción en la opinión pública. Si bien existen análisis generales sobre la estrategia comunicativa de partidos y líderes políticos en X/Twitter, siguen siendo escasos los trabajos centrados en el análisis individualizado del discurso y su recepción en el contexto español.

Desde una perspectiva personal, el presente trabajo surge del interés por comprender si ciertos comportamientos de los representantes políticos —como la adopción de discursos individualistas, el tono confrontativo o el uso estratégico de temáticas nacionalistas— condicionan la respuesta de la ciudadanía. En particular, se busca analizar si existen patrones que se repiten entre partidos de ámbito nacional y aquellos con una agenda autonómica o independentista, y cómo estas diferencias afectan al estilo comunicativo, al nivel de interacción y al tono de las respuestas recibidas por parte del electorado. Esta curiosidad por entender las dinámicas reales del discurso político digital motivó el uso de técnicas de ciencia de datos y procesamiento del lenguaje natural.

Estudios recientes confirman que el tratamiento automático del lenguaje en español presenta desafíos específicos, como la capacidad que tienen las palabras de cambiar su forma dependiendo de su función gramatical en la oración, la ambigüedad semántica y la escasa disponibilidad de corpus etiquetados para tareas políticas [22, 17]. Además, se observa una clara falta de consenso metodológico en esta área de estudio[17]. Aunque algunos trabajos han comenzado a aplicar técnicas avanzadas de análisis de sentimientos en el contexto electoral español, como el estudio de Rodríguez-Ibáñez (2021) [42], estos se centran en tendencias globales por partido, sin profundizar en la caracterización individual de los actores ni en la respuesta ciudadana diferenciada.

Otros trabajos han demostrado que el tono emocional del mensaje influye directamente en el tipo de interacción recibida. Svetlov y Platonov (2019) [47], por ejemplo, identifican que los mensajes negativos generan mayor volumen de comentarios y reposts, mientras que los positivos reciben más likes y visualizaciones. Estas dinámicas se ven complementadas por estudios como el de Habibi y Sunjana (2019)[25], quienes aplican conjuntamente análisis de sentimiento y análisis de redes para estudiar la polarización política en Indonesia, identificando actores influyentes mediante métricas de centralidad y comunidades.

Desde una perspectiva aplicada, investigaciones recientes en marketing digital y minería de datos reafirman el valor de utilizar tecnologías de inteligencia artificial para interpretar grandes volúmenes de datos sociales y anticipar reacciones del público ante distintos tipos de mensajes [35]. Este enfoque metodológico permite abordar el discurso político desde múltiples dimensiones —tono, contenido y recepción— y extraer información estructurada para su posterior análisis.

En este contexto, este trabajo propone un estudio detallado del comportamiento comunicativo de los políticos españoles en X/Twitter, utilizando técnicas de procesamiento de lenguaje natural y visualización de datos. Se pretende con ello avanzar en la comprensión de cómo se construyen los discursos políticos digitales en el entorno hispanohablante y qué impacto tienen sobre una audiencia altamente segmentada por factores ideológicos, territoriales y demográficos.

2.3. Objetivos

El objetivo principal de este trabajo consiste en desarrollar un proceso de análisis basado en una metodología de ciencia de datos que permita estudiar y tratar de darle un sentido al estilo de comunicación de la clase política española en redes sociales, particularmente en la plataforma X/Twitter. Tal como se expuso en la Introducción, este trabajo busca dar respuesta a dos preguntas de investigación: (1) si existen diferencias significativas en el estilo, el tono o el nivel de interacción de los mensajes políticos según el partido o la comunidad autónoma del emisor, y (2) qué relación existe entre el tono y la temática de los mensajes y las métricas de interacción obtenidas, así como el tono de las respuestas recibidas.

A partir de esta meta general, se han definido los siguientes objetivos específicos para abordar dichas cuestiones:

1. Calcular índices de popularidad e interacción de políticos y partidos en X/Twitter, con el fin de evaluar el alcance y la respuesta generada por sus mensajes.
2. Identificar patrones de actividad regional mediante mapas que permitan visualizar las diferencias comunicativas entre las distintas comunidades autónomas españolas.
3. Realizar un análisis de sentimientos aplicado tanto a los mensajes emitidos por los políticos como a las respuestas recibidas por parte del público y las posibles réplicas que estos reciban.

4. Extraer palabras clave y temas más recurrentes en los mensajes y comentarios, clasificándolos según su tono emocional y temática predominante.
5. Como objetivo opcional, se contempla la posibilidad de implementar una aplicación web básica que permita explorar de forma dinámica los datos obtenidos, siempre que las condiciones técnicas y temporales lo permitan.

Estos objetivos están alineados con los principios del análisis exploratorio de datos y con las recomendaciones metodológicas para el tratamiento del lenguaje natural aplicado a textos políticos en español.

Capítulo 3

Aportaciones del trabajo

3.1. Principales aportaciones

Este Trabajo de Fin de Título (TFT) realiza una aportación significativa al análisis de tono y tema de la comunicación política en redes sociales en el idioma español, mediante la aplicación conjunta de técnicas de procesamiento del lenguaje natural y ciencia de datos. El valor de este trabajo viene dado a través de un proceso completo de recopilación, tratamiento y análisis de datos reales de actividad política en la red X/Twitter, centrado en representantes políticos españoles a nivel nacional y autonómico.

Desde el punto de vista técnico, se ha desarrollado un sistema de adquisición y enriquecimiento de datos que combina información institucional (procedente de fuentes como Wikipedia y la página oficial del Congreso de los Diputados), metadatos de perfil y actividad en redes sociales, y atributos lingüísticos. Este sistema incluye la detección y traducción automática de textos multilingües, así como la clasificación manual de una muestra de publicaciones y respuestas. A partir de esta base, se ha procedido al reentrenamiento supervisado de un modelo basado en BERT específicamente adaptado para el análisis de sentimiento y de temática en español, incorporando técnicas de ajuste por clases desbalanceadas como *class weighting* y *focal loss*, las cuales serán tratadas en profundidad en la Sección 4.5.

En términos científicos, el trabajo contribuye al estudio del discurso político digital, permitiendo explorar preguntas relacionadas con el tono emocional, la estrategia temática y la interacción del público. El enfoque adoptado permite también analizar el posible efecto de variables como el nivel educativo del emisor, su área de formación, la región a la que representa o el tipo de partido al que pertenece (nacional, regional o independentista).

Finalmente, se implementó una herramienta de visualización, logrando representar los resultados de forma clara y accesible a través de un panel interactivo desarrollado con Streamlit y Plotly. Esta herramienta permite explorar los datos de manera sencilla, aplicando filtros y navegando entre distintos tipos de gráficas. De este modo, el trabajo no solo aporta al desarrollo de modelos de PLN en español, sino que también ayuda a entender mejor cómo se comunican los políticos y cómo responden los ciudadanos en redes sociales.

3.2. Competencias específicas

Este TFT ha permitido poner en práctica conocimientos y habilidades propias del ámbito de los sistemas inteligentes y la ciencia de datos. En específico, se han abordado las siguientes competencias:

ED1: Capacidad para conocer los fundamentos, paradigmas y técnicas propias de los sistemas inteligentes y analizar, diseñar y construir sistemas, servicios y aplicaciones informáticas que utilicen dichas técnicas en cualquier ámbito de aplicación.

Esta competencia se ha desarrollado al aplicar modelos avanzados de inteligencia artificial, como BERT, para analizar publicaciones políticas. Para ello, se han recopilado datos reales desde distintas fuentes, se han limpiado y transformado, y se ha entrenado un modelo con técnicas especializadas para tratar desequilibrios en las clases. Todo este proceso ha permitido construir un sistema capaz de clasificar mensajes según su tono o temática.

ED8: Capacidad de concebir sistemas, aplicaciones y servicios basados en tecnologías de la información y las comunicaciones para Ciencia e Ingeniería de Datos, incluyendo Internet, web, comercio electrónico, multimedia, servicios interactivos y computación móvil.

Esta competencia se ha trabajado al diseñar todo el proceso de tratamiento de datos desde redes sociales y páginas institucionales. El proyecto incluye tanto la recogida como el análisis de datos, además de la creación de una aplicación interactiva que permite visualizar los resultados del análisis. Para ello, se han usado herramientas como Streamlit o Plotly, que permiten presentar los datos de forma visual y accesible para distintos tipos de usuarios.

3.3. Alineamiento con los objetivos de desarrollo sostenible

En esta sección se justificó el alineamiento del TFT con aquellos **ODS** con los que presentaba un mayor grado de relación, según lo indicado en el Cuadro 3.1.

Cuadro 3.1: Grado de relación del TFT con los objetivos de desarrollo sostenible.

ODS	Grado de relación			
	0 No procede	1 Bajo	2 Medio	3 Alto
1 Fin de la Pobreza	×			
2 Hambre cero	×			
3 Salud y Bienestar	×			
4 Educación de calidad		×		
5 Igualdad de género		×		
6 Agua limpia y saneamiento	×			
7 Energía Asequible y no contaminante	×			
8 Trabajo decente y crecimiento económico	×			
9 Industria, Innovación e Infraestructuras			×	
10 Reducción de las desigualdades		×		
11 Ciudades y comunidades sostenibles	×			
12 Producción y consumo sostenibles	×			
13 Acción por el clima	×			
14 Vida submarina	×			
15 Vida de ecosistemas terrestres	×			
16 Paz, justicia e instituciones sólidas				×
17 Alianzas para lograr objetivos		×		

Este TFT está relacionado con varios Objetivos de Desarrollo Sostenible (ODS), en especial los que promueven la transparencia de las instituciones, la mejora de las competencias digitales y el uso de tecnologías avanzadas para el beneficio de la sociedad. A continuación se señalan los ODS más directamente vinculados con este trabajo:

- **ODS 16 – Paz, justicia e instituciones sólidas (3 - Alto):** Este trabajo contribuye a mejorar la transparencia institucional al analizar cómo se comunican los representantes políticos en redes sociales. El uso de datos accesibles públicamente y de métodos de análisis bien documentados permite que otros investigadores puedan replicar, verificar o ampliar los resultados, lo que favorece mejor debate público.
- **ODS 9 – Industria, innovación e infraestructuras (2 - Medio):** Este trabajo aplica tecnologías emergentes como el procesamiento del lenguaje natural, el aprendizaje profundo y el análisis automatizado de redes sociales, en un contexto social y político relevante.
- **ODS 4 – Educación de calidad (1 - Bajo):** Este proyecto ayuda a que más personas comprendan cómo se usan los datos para analizar temas sociales y políticos. Al aplicar herramientas digitales a la comunicación política, se fomenta el aprendizaje y la comprensión tanto a nivel universitario como entre la ciudadanía en general.
- **ODS 5 – Igualdad de género (1 - Bajo):** Este trabajo tiene en cuenta el género de los políticos analizados y permite observar si hombres y mujeres reciben o generan

respuestas diferentes en redes sociales. Esto ayuda a detectar posibles desigualdades en la forma en que se participa y se reacciona en el debate político en redes sociales.

- **ODS 10 – Reducción de las desigualdades (1 - Bajo):** Al estudiar cómo se comunican los políticos según su territorio, partido o perfil personal, este proyecto permite identificar si existen diferencias en la visibilidad o el impacto de sus mensajes, lo que puede reflejar desigualdades en el entorno digital.
- **ODS 17 – Alianzas para lograr los objetivos (1 - Bajo):** El trabajo se apoya en herramientas, modelos y datos que están disponibles públicamente. Esto facilita que otros investigadores o desarrolladores puedan reutilizar y mejorar lo ya hecho, promoviendo así la colaboración y el avance conjunto en el análisis de datos sociales.

Capítulo 4

Desarrollo

4.1. Metodología

Este trabajo sigue la metodología CRISP-DM (Cross Industry Standard Process for Data Mining), modelo ampliamente usado en proyectos de ciencia de datos por su estructura adaptable a distintas áreas de aplicación.

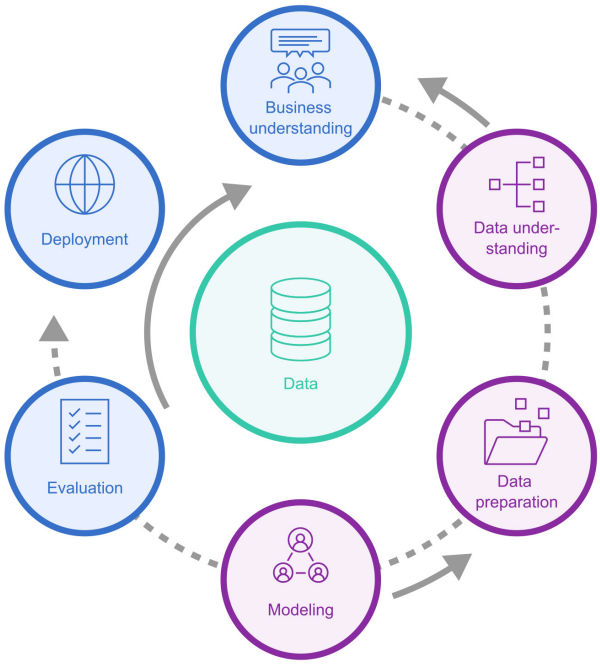

Ilustración 4.1: Fases del modelo CRISP-DM. Fuente: Fraunhofer Institute for Surface Engineering and Thin Films IST [20].

Tal y como se muestra en la Ilustración 4.1, el modelo CRISP-DM se compone de seis fases principales que guían el desarrollo del proyecto de forma cíclica y estructurada.

A continuación, se describen en detalle las fases tal y como se han aplicado en este trabajo:

4.1.1. Comprensión del negocio

En esta primera etapa se definió el objetivo general del proyecto: analizar el comportamiento comunicativo de los políticos españoles en la red social X/Twitter, con especial atención al tono de los mensajes, los principales temas tratados, la respuesta generada entre los usuarios y las posibles réplicas de los políticos a ciertos comentarios. Para ello, se seleccionaron tres grupos clave: diputados del Congreso, presidentes de comunidades autónomas y líderes de la oposición en dichas comunidades. Además, se identificaron variables relevantes (edad, formación, partido, territorio) para el análisis posterior.

4.1.2. Comprensión de los datos

En esta fase se recopiló información desde múltiples fuentes: perfiles oficiales de X/Twitter, Wikipedia y el portal del Congreso de los Diputados. A partir de estos recursos se extrajo información relevante como el número de seguidores, fechas de publicación, respuestas y metadatos de contexto. También se incorporaron variables derivadas como el área académica de formación o el nivel educativo de cada político y, finalmente, se aplicaron técnicas de detección de idioma y traducción para el contenido de los mensajes.

4.1.3. Preparación de los datos

Una vez recopilada la información, se procedió a su limpieza, transformación y añadido de información de contexto. Se eliminaron duplicados, publicaciones sin texto (como aquellas que contenían solo enlaces) y símbolos irrelevantes. Además, se detectó el idioma del contenido mediante langdetect, y se tradujeron automáticamente los textos no españoles utilizando la librería deep-translator con el motor de Google.

Los datos se estructuraron en tres bloques:

- **Metadata:** información del político (nombre, edad, comunidad autónoma, formación, partido, número de legislaturas, seguidores, etc.).
- **Posts:** publicaciones originales en X/Twitter con sus metadatos (fecha, interacciones, idioma, texto original y traducido).
- **Comentarios:** respuestas a cada publicación, incluyendo enlaces, número de likes, si recibieron respuesta del político, idioma y texto traducido.

Más adelante se llevó a cabo una tarea de etiquetado manual sobre un subconjunto representativo del corpus: 900 publicaciones y 1600 comentarios fueron clasificados por *tono* (positivo, negativo o neutro) y, en el caso de los posts, también por *temática* (p.ej., 'Gestión Pública e Instituciones", Economía y Empleo", Sociedad e Igualdad").

4.1.4. Modelado

A partir del conjunto etiquetado, se procedió al reentrenamiento del modelo mencionado previamente, SaBERT-Spanish-Sentiment-Analysis, disponible en la plataforma Hugging Face. Este modelo, basado en BERT para español, fue ajustado con los nuevos datos mediante técnicas de balanceo como *class weighting* y *focal loss*, con el objetivo de mejorar el rendimiento en clases menos representadas.

Se entrenaron dos modelos independientes:

- Un modelo de clasificación de **tono** para posts y comentarios.
- Un modelo de clasificación de **temática** para los posts.

Ambos modelos fueron validados mediante precisión, recall y F1-score.

4.1.5. Evaluación

Los modelos entrenados fueron aplicados sobre el corpus completo de publicaciones, comentarios y respuestas. A partir de las predicciones de tono y clasificación temática, se analizaron correlaciones con métricas de interacción (número de likes, retweets, comentarios) y se compararon patrones entre diferentes partidos políticos y comunidades autónomas.

Asimismo, se exploraron las relaciones entre la orientación del discurso (positivo, negativo, neutro) y las respuestas generadas por los usuarios, incluyendo la identificación de entidades y términos clave más frecuentes según cada tono o temática.

4.1.6. Despliegue

Como fase final, se diseñó una herramienta para la visualización exploratoria de resultados. Este entorno emplea tecnologías como Plotly para la generación de gráficos dinámicos y Streamlit para la construcción de la interfaz web.

La herramienta ofrece visualizaciones en forma de gráficos de tarta (pie chart), mapas, así como la capacidad de alterar dichas gráficas mediante la aplicación de filtros (partido, comunidad, rango temporal, tono, tema). El objetivo es poner los resultados a disposición de usuarios sin tantos conocimientos técnicos, facilitando un análisis interactivo y reproducible de los patrones encontrados en los datos.

La estructura iterativa de CRISP-DM permite revisar y ajustar cada fase a medida que avanzan los resultados, lo que ha resultado especialmente útil en este proyecto debido a la diversidad de los datos y al enfoque exploratorio del análisis.

En el siguiente Cuadro 4.1 se ha elaborado un resumen en el que se detallan los distintos pasos de la fase de desarrollo, junto con las técnicas y herramientas que se han usado para llevarlos a cabo.

Cuadro 4.1: Resumen de técnicas y herramientas empleadas en cada fase del proyecto

Apartado	**Técnicas utilizadas**	**Herramientas software**
4.2. Recolección y organización de datos	Diseño del dataset, extracción estructurada de datos, filtrado de contenidos (por fechas, tipo de publicación), navegación automática con seguimiento de relaciones post-comentario, obtención de metadatos, almacenamiento en tablas	Python (Selenium, BeautifulSoup, pandas), scripts propios de scraping, Microsoft Excel para organización de tablas
4.3. Limpieza y traducción	Filtros de calidad textual (retweets, duplicados, publicaciones sin texto), detección automática de idioma con Naïve Bayes y n-gramas, traducción automática	Python (pandas), langdetect, deep_translator con Google Translate
4.4. Etiquetado y preparación del corpus	Etiquetado manual supervisado, definición de dimensiones (tono, temática), creación de corpus de entrenamiento	Microsoft Excel, revisión manual de anotaciones, estrategia de clasificación supervisada
4.5. Entrenamiento y ajuste de modelos	Transfer learning, fine-tuning, balanceo de clases, class weighting, focal loss, evaluación de precisión, ajuste de hiperparámetros	HuggingFace Transformers (PyTorch), modelo SaBERT-Spanish-Sentiment-Analysis, scikit-learn para métricas
4.6. Aplicación del modelo y análisis exploratorio	Clasificación automática de nuevos textos, etiquetado masivo de mensajes, generación de nuevas columnas y gráficas para análisis	pandas, modelos entrenados con Transformers, visualización con plotly, geopandas, etc.
4.7. Despliegue	Creación de un panel interactivo de análisis exploratorio, integración de filtros y visualizaciones dinámicas	Streamlit para la interfaz web, plotly para las gráficas dinámicas, Python

Además de las técnicas y herramientas recogidas en el Cuadro 4.1, se contó con el apoyo puntual de herramientas de Inteligencia Artificial como fue el caso de **ChatGPT** en distintas fases del proyecto. Su uso se centró en:

- Mejorar la claridad y la fluidez de los textos, corrigiendo errores ortográficos, eliminando repeticiones innecesarias y facilitando la reorganización de tablas y secciones.
- Optimizar y limpiar fragmentos de código, agrupando funciones de forma más organizada y añadiendo elementos visuales como iconos o emojis a la interfaz para mejorar su usabilidad.
- Proponer librerías y recursos técnicos relevantes que complementaran el desarrollo de la parte de análisis y visualización de datos.
- Traducir, de forma puntual, citas bibliográficas al español.

El uso de ChatGPT se limitó a estas tareas de apoyo, manteniendo el control y la validación final de los resultados en manos del autor del proyecto.

4.2. Recolección y organización de datos

El motivo por el que se decidió obtener los datos de forma manual, en lugar de utilizar una base de datos ya existente, fue la escasa disponibilidad y actualización de los datasets disponibles. Las bases de datos accesibles carecían de muchos de los perfiles políticos que se querían analizar, y no incluían toda la información necesaria para los objetivos del trabajo. Además, aquellas que sí estaban disponibles solían estar desactualizadas y, en muchos casos, incluían formaciones políticas que ya no tienen representación relevante en la política española, como es el caso del partido político Ciudadanos.

La intención de este trabajo era obtener información que permitiera no solo realizar comparaciones contextuales entre los diferentes políticos, sino también aplicar técnicas de procesamiento del lenguaje natural (PLN) sobre sus publicaciones. Por ello, además de recopilar datos técnicos como la edad, los estudios o el partido al que pertenece cada político, se consideró imprescindible extraer una cantidad representativa de publicaciones en X/Twitter. De este modo, se podría analizar en mayor profundidad el tipo de comunicación que se emplea en la política digital contemporánea y cómo esta varía entre perfiles e instituciones.

El siguiente paso consistió en determinar qué políticos serían objeto de análisis. Para este proyecto se establecieron dos grandes grupos de estudio:

Presidentes y líderes de la oposición en las comunidades autónomas: Estos perfiles representan un punto intermedio entre la política nacional y la política local. Al tratarse de responsables de gobiernos regionales y de sus principales opositores, su comunicación en redes sociales suele dirigirse a un electorado más reducido y específico, lo que les obliga a mantener un tono generalmente más cercano y contextualizado. Los mensajes que publican suelen estar vinculados a la actualidad de su comunidad autónoma, abordando temas concretos como problemas sociales o económicos, logros institucionales o mensajes de carácter regional.

Al ser líderes visibles de un territorio determinado, las respuestas que generan también tienden a proceder, al menos teóricamente, de ciudadanos de esa misma comunidad, aun-

que esta relación no pueda verificarse con seguridad al no contar con metadatos fiables de localización geográfica en las redes sociales.

Este estudio no incluye a representantes de las ciudades autónomas de Ceuta y Melilla, ya que su escala territorial no es equiparable a la de una comunidad autónoma. Incorporarlas habría requerido considerar también ciudades relevantes y de mayor tamaño como Madrid, Barcelona o Valencia, lo que se saldría del marco de análisis.

La inclusión de los líderes de la oposición, además de equilibrar la representación política, permite contrastar si estos adoptan estrategias distintas en sus discursos a las de los presidentes autonómicos. En particular, se busca explorar si dirigen su actividad comunicativa hacia una crítica constante a sus contrapartes o si centran sus mensajes en sus propias propuestas. Este tipo de comportamiento puede identificarse mediante el análisis de hashtags, menciones directas y el tono de los mensajes emitidos.

La información detallada sobre cada uno de estos representantes autonómicos se resume en el siguiente cuadro, que recoge los presidentes en ejercicio a fecha de 09/01/2025 y la comunidad autónoma a la que representan:

Cuadro 4.2: Presidentes autonómicos seleccionados

Nombre	**Comunidad Autónoma**
Juanma Moreno	Andalucía
Jorge Azcón	Aragón
Fernando Clavijo	Canarias
María José Sáenz de Buruaga	Cantabria
Emiliano García-Page	Castilla-La Mancha
Alfonso Fernández Mañueco	Castilla y León
Salvador Illa	Cataluña
Isabel Díaz Ayuso	Comunidad de Madrid
María Chivite	Comunidad Foral de Navarra
Carlos Mazón	Comunidad Valenciana
María Guardiola	Extremadura
Alfonso Rueda	Galicia
Marga Prohens	Islas Baleares
Gonzalo Capellán	La Rioja
Imanol Pradales	País Vasco
Adrián Barbón	Principado de Asturias
Fernando López Miras	Región de Murcia

El siguiente cuadro recoge los líderes de la oposición en cada comunidad autónoma seleccionados para este estudio. Estos políticos desempeñan un papel clave en el contraste del discurso frente a los presidentes autonómicos, y su análisis permite observar dinámicas de confrontación o propuesta dentro del ámbito regional. La selección se basa en los principales partidos de la oposición en cada comunidad y en la existencia de perfiles activos en la red

social X/Twitter.

Cuadro 4.3: Líderes de la oposición autonómicos seleccionados

Nombre	**Comunidad Autónoma**
María Jesús Montero Cuadrado	Andalucía
Pilar Alegría	Aragón
Ángel Víctor Torres	Canarias
Miguel Ángel Revilla	Cantabria
Paco Núñez	Castilla-La Mancha
Carlos Martínez Mínguez	Castilla y León
Albert Batet	Cataluña
Mónica García Gómez	Comunidad de Madrid
Cristina Ibarrola	Comunidad Foral de Navarra
Diana Morant Ripoll	Comunidad Valenciana
Miguel Ángel Gallardo	Extremadura
Ana Pontón	Galicia
Francina Armengol	Islas Baleares
Javier García Ibáñez	La Rioja
Pello Otxandiano	País Vasco
Álvaro Queipo	Principado de Asturias

Como se puede apreciar, no se ha incluido al líder de la oposición en la Región de Murcia. Esto se debe a que el político Francisco Lucas Ayala, actual Secretario General del PSOE en dicha comunidad, no dispone de una cuenta identificable en la red social X/Twitter a fecha de consulta. Cabe destacar además el caso particular de María Jesús Montero, quien aparece tanto como diputada nacional como líder de la oposición en Andalucía. Esta duplicidad se abordará más adelante con mayor detalle, ya que su perfil comunicativo se inclina más hacia el ámbito estatal que hacia el autonómico.

En lo que respecta a los representantes del Congreso de los Diputados, se optó por seleccionar únicamente a aquellos diputados que además ocuparan cargos relevantes dentro de la estructura interna de sus partidos. Esta decisión fue tomada por dos motivos fundamentales: por un lado, reducir el volumen total de perfiles a analizar (350 diputados resultan excesivos para una tarea de análisis individualizado); y por otro, garantizar que los perfiles seleccionados tuvieran una actividad destacable en redes sociales, al combinar funciones representativas e institucionales.

Se descartaron los senadores por su escasa presencia mediática y la dificultad para acceder a información relevante. En los casos de partidos con una representación muy reducida, como el PNV (5 escaños) o EH Bildu (6 escaños), se aplicó una media ponderada basada en el número de diputados seleccionados por partido, eligiendo finalmente un único representante por cada uno.

Una vez se tuvo la lista inicial elaborada, se revisaron manualmente las cuentas de X/T-

witter asociadas a cada político, eliminando aquellos perfiles que no mostraban actividad frecuente, que dicha actividad hubiese cesado mucho tiempo antes de la realización de este trabajo o que, por sus características técnicas, pudieran dificultar la posterior extracción de publicaciones y comentarios.

El cuadro siguiente recoge los diputados seleccionados, incluyendo su cargo dentro del partido y la provincia por la que ejercen su representación en el Congreso:

Cuadro 4.4: Diputados del Congreso seleccionados con cargos en la ejecutiva de sus partidos

Nombre	**Cargo en el partido**	**Provincia diputación**
Cristina Narbona Ruiz	Presidenta del PSOE	Madrid
Pedro Sánchez Pérez-Castejón	Secretario General del PSOE	Madrid
María Jesús Montero Cuadrado	Vicesecretaria General del PSOE	Sevilla
Santos Cerdán León	Secretario de Organización del PSOE	Navarra
Esther Peña Camarero	Portavoz de la Ejecutiva Federal del PSOE	Burgos
Alfonso Rodríguez Gómez de Celis	Sec. de Política Institucional y Formación PSOE	Sevilla
Alejandro Soler Mur	Secretario de Política Municipal del PSOE	Alicante
Elisa Garrido Jiménez	Secretaria de Medio Rural del PSOE	La Rioja
Félix Bolaños García	Secretario de Justicia del PSOE	Madrid
Francisco Lucas Ayala	Secretario de Reforma Constitucional del PSOE	Murcia
Javier Alfonso Cendón	Secretario de Ciencia, Innovación y Universidades PSOE	León
Montse Mínguez García	Secretaria de Trabajo, Economía Social PSOE	Lleida
Víctor Gutiérrez Santiago	Secretario LGTBI del PSOE	Madrid
Luc Andre Diouf Dioh	Secretario de Migraciones y Refugiados del PSOE	Las Palmas
Óscar Puente Santiago	Vocal de la Ejecutiva del PSOE	Valladolid
Patxi López Álvarez	Portavoz del Grupo Socialista en el Congreso	Bizkaia
Víctor Camino Miñana	Secretario General de JSE	Valencia
Alberto Núñez Feijóo	Presidente del PP	Madrid
Concepción Gamarra Ruiz-Clavijo	Secretaria General del PP	La Rioja
Elías Bendodo Benasayag	Coordinador General del PP	Málaga
Carmen Fúnez de Gregorio	Vicesecretaria de Políticas Sociales del PP	Ciudad Real
Juan Bravo Baena	Vicesecretario de Economía del PP	Sevilla
Ana Isabel Alós López	Vicesecretaria de Igualdad del PP	Huesca
Borja Sémper Pascual	Portavoz nacional del PP	Madrid
Noelia Núñez González	Vicesecretaria Movilización y Reto Digital del PP	Madrid
Miguel Tellado Filgueira	Vicesecretario Coordinación Autonómica del PP	A Coruña
Carmen Navarro Lacoba	Vicesecretaria Política Social del PP	Albacete
Óscar Clavell López	Vicesecretario provincial PP Castellón	Castellón
Jaime Miguel De los Santos González	Secretario de Cultura del PP	Madrid
Santiago Abascal Conde	Presidente de VOX	Madrid
Rocío De Meer Méndez	Vocal del Comité Ejecutivo Nacional de VOX	Almería
Francisco Javier Ortega Smith-Molina	Vicepresidente de VOX	Madrid
María José Rodríguez de Millán Parro	Portavoz VOX en el Senado	Madrid
María de los Reyes Romero Vilches	Vicepresidenta de VOX	Sevilla
Lander Martínez Hierro	Sec. técnico y finanzas Sumar	Bizkaia
Txema Guijarro García	Sec. general GP Sumar	Alicante
Rafael Cofiño Fernández	Secretario de Sanidad Sumar	Asturias
Carlos Martín Urriza	Portavoz Economía GP Sumar	Madrid
Yolanda Díaz Pérez	Líder de Sumar	Madrid
Gabriel Rufián Romero	Portavoz de ERC en el Congreso	Barcelona
Inés Granollers Cunillera	Secretaria de Política Territorial, Vivienda y Seguridad	Lleida
Míriam Nogueras i Camero	Portavoz de Junts en el Congreso	Barcelona
Josep Maria Cruset Domènech	Secretario de Industria y Turismo	Tarragona
Aitor Esteban Bravo	Portavoz del PNV en el Congreso	Bizkaia
Oskar Matute García de Jalón	Portavoz adjunto de EH Bildu	Bizkaia

4.2.1. Creación del conjunto de datos

Para organizar toda la información recopilada, se tomó la decisión de utilizar la herramienta *Microsoft Excel* como entorno de trabajo inicial. Esta elección viene dada por la facilidad de uso de esta herramienta, su capacidad para estructurar grandes volúmenes de información en hojas independientes y su compatibilidad con formatos como `.csv` o `.sql`, lo que facilitó la posterior carga de datos desde entornos de programación como Python.

El conjunto de datos creado se dividió en tres hojas principales, cada una centrada en un tipo de entidad o interacción diferente:

- **Metadata:** contiene información general sobre los políticos seleccionados, incluyendo datos personales, trayectoria política y detalles de su perfil en X/Twitter. Esta hoja sirvió como punto de unión entre el resto del contenido, facilitando la adición de métricas y la posterior búsqueda de patrones según variables como edad, comunidad autónoma, partido político o nivel educativo.
- **Posts:** en esta hoja se almacenaron las publicaciones extraídas de las cuentas oficiales de X/Twitter de cada político. Incluye campos como el enlace al post, contenido textual, número de *likes*, *retweets*, fecha de publicación, una columna con la detección del idioma y su correspondiente traducción al castellano.
- **Comentarios:** en esta última hoja se incluyeron los comentarios realizados por otros usuarios en respuesta a los posts anteriores, junto con los posibles comentarios de réplica por parte del propio político. Además de los campos anteriores, se añadió información sobre la fecha de publicación del comentario, el número de interacciones, el idioma original y su traducción al español.

Esta forma de estructurar la información permitió un desarrollo más ágil y un control más preciso de la calidad de los datos, facilitando tanto la exploración inicial como las fases posteriores de análisis y modelado.

Metadata

Como se ha mencionado anteriormente, la hoja *Metadata* contiene la información descriptiva de los políticos incluidos en este TFT. Las variables almacenadas inicialmente en esta hoja son las siguientes:

- **ID_Político:** Identificador numérico único asignado a cada político para facilitar su referencia en otros conjuntos de datos.
- **Nombre:** Nombre completo del político, siguiendo la forma en la que se muestran en las fuentes oficiales.
- **Cargo:** Clasificación del político según su posición institucional: *Presidente* (de comunidad autónoma), *Diputado* (del Congreso) o *Líder_Oposición* (en comunidades autónomas).

A fecha de 09/01/2025, cabe destacar que María Jesús Montero es la única persona incluida que ocupa dos categorías simultáneamente. Sin embargo, tras revisar su perfil de X/Twitter, se concluyó que su actividad está más relacionada con su rol de diputada y vicepresidenta del Gobierno que con el de líder del PSOE andaluz.

- **Twitter:** Usuario oficial en la red X/Twitter, utilizado para identificar y acceder al perfil del político.
- **Género:** Clasificación binaria entre "Hombre" y "Mujer", adoptada para simplificar el tratamiento de los datos.
- **Estudios:** Nivel máximo de estudios completado, categorizado como:
  - *Secundaria*
  - *Bachillerato*: Incluye Bachillerato y Formación Profesional.
  - *Grado*: Carrera universitaria de primer ciclo.
  - *Postgrado*: Másteres, doctorados o títulos equivalentes de formación avanzada.
- **Campo:** Área de conocimiento general asociada a los estudios o profesión del político, como "Ingeniería_Tecnología" o "Ciencias_Sociales".
- **Comunidad Autónoma:** Comunidad a la que representa el político. En el caso de los diputados, esta se ha determinado en función de la provincia por la que salieron elegidos al Congreso.
- **Partido:** Nombre del partido político por el que se presentó el político. En algunos casos se han considerado federaciones regionales (como PSC, PSN, PSE-EE) como parte del PSOE.
- **Legislaturas:** Lista de años de inicio de cada legislatura en la que el político ha ocupado cargo institucional.
- **Número de Legislaturas:** Valor numérico total de legislaturas ejercidas, derivado de la variable anterior.
- **Edad:** Edad del político calculada a fecha 27 de abril de 2025.
- **Posts:** Número total aproximado de publicaciones en su cuenta, incluyendo tanto publicaciones originales como retweets.

  Es importante señalar que esta cifra puede estar redondeada, ya que X/Twitter utiliza abreviaciones como "mil" o "M" a partir de ciertas cifras.
- **Seguidores:** Número de seguidores de la cuenta del político, también sujeto a las abreviaciones señaladas en la variable anterior.
- **Comienzo en X/Twitter:** Año en el que se creó la cuenta.
- **Descripción:** Texto descriptivo añadido por el político en su perfil.

Posts

La segunda hoja del dataset contiene las publicaciones extraídas de las cuentas de X/Twitter de los políticos seleccionados, junto con información complementaria relevante para el análisis. No incluye los comentarios de los usuarios, los cuales se tratan en una hoja independiente. Las variables incluidas al inicio del proyecto son:

- **ID_Político:** Referencia cruzada con la hoja *Metadata*, utilizada para asociar cada publicación con el político correspondiente.
- **Enlace_Post:** URL completa de la publicación en la red X/Twitter.
- **Contenido:** Texto original del mensaje publicado por el político.
- **Fecha_Publicación:** Fecha y hora exacta en la que se realizó la publicación.
- **Comentarios_Extraidos:** Número de comentarios asociados al post que han sido efectivamente extraídos y almacenados en la hoja correspondiente.
- **Comentarios_Totales:** Número total de comentarios recibidos por la publicación, contabilizados a fecha 27 de abril de 2025.
- **Retweets:** Número de veces que el mensaje fue compartido por otros usuarios.
- **Likes:** Número total de usuarios que marcaron la publicación con un "me gusta".
- **Idioma:** Idioma original del contenido publicado.
- **Contenido_Traducido:** Versión traducida al castellano del texto del post, si fue necesario. En el caso de publicaciones ya escritas en español, se copia directamente el contenido original.
- **Tono:** Grado de intencionalidad del post del político dividido en tres niveles; Positivo, Neutro y Negativo
- **Tema:** Asunto tratado en el mensaje del político dividido en cuatro categorías; "Gestión Pública e Instituciones", "Economía, Empresa, Empleo e Infraestructuras", "Sociedad, Igualdad y Derechos" y "Otros".

Comentarios

La tercera y última hoja del dataset contiene los comentarios públicos realizados por otros usuarios en respuesta a las publicaciones de los políticos. En ella también se incluyen, en su caso, las réplicas dadas por los propios políticos. Las variables consideradas son:

- **Enlace_Post:** URL de la publicación original a la que corresponde el comentario.
- **Enlace_Comentario:** URL completa del comentario individual.
- **Contenido:** Texto completo del comentario realizado por un usuario.
- **Fecha_Publicación:** Fecha y hora en la que se publicó el comentario.

- **Comentarios:** Número de respuestas que recibió el comentario.
- **Retweets:** Número de veces que el comentario fue compartido por otros usuarios.
- **Likes:** Número de "me gusta" recibidos por el comentario.
- **Respuesta:** Texto de la respuesta emitida por el político, si la hubo.
- **Idioma_Comentario:** Idioma original del comentario del usuario.
- **Idioma_Respuesta:** Idioma original de la posible respuesta del político.
- **Comentario_Traducido:** Comentario traducido al castellano, en caso de no estar originalmente en este idioma.
- **Respuesta_Traducida:** Versión en español de la respuesta del político, si procede.
- **Tono:** Grado de intencionalidad del comentario dividido en tres niveles; positivo, neutro y negativo.
- **Tono_Respuesta:** Grado de intencionalidad de la respuesta del político, en caso de que la hubiera.

4.2.2. Web scraping

Para completar las hojas de datos previamente descritas —*Metadata*, *Posts* y *Comentarios*— fue necesario recurrir a técnicas automatizadas de extracción de información directamente desde las plataformas en las que se encuentra publicada de forma accesible, pero no descargable. En concreto, ni Wikipedia ni el portal del Congreso de los Diputados ofrecen APIs que permitan acceder de forma estructurada a los datos requeridos, como legislaturas, edad de los políticos, perfiles de diputados o cargos institucionales. Por esta razón, se optó por emplear técnicas de *web scraping*.

Esta decisión también se aplicó a la red social X/Twitter, cuya API gratuita no permite la descarga de publicaciones ni comentarios, y cuyas versiones de pago presentan importantes restricciones en cuanto a volumen de acceso a los datos y costes asociados. Aplicar técnicas de *scraping* no solo permitió superar estas limitaciones, sino que posibilitó estructurar los datos de manera personalizada y adaptada a las necesidades específicas del proyecto, permitiendo así obtener toda la información necesaria para las variables ya definidas en el conjunto de datos.

El web scraping en sí es una técnica automatizada que permite extraer información estructurada a partir del contenido visible en páginas web. Se basa en simular la navegación de un usuario humano para acceder a la información publicada en un sitio, procesar el código HTML de la página y extraer aquellos datos que resultan relevantes con el objetivo de almacenarlos y analizarlos posteriormente.

Esta estrategia resulta especialmente útil cuando no se dispone de una API (Interfaz de Programación de Aplicaciones) pública, gratuita o completa que proporcione acceso directo a los datos, o bien cuando las disponibles son de pago, presentan limitaciones de uso (en

número de peticiones, tiempo o volumen), o no incluyen todos los atributos necesarios para el estudio.

El proceso de scraping suele implicar varias acciones automatizadas: cargar páginas web de forma dinámica, recorrer enlaces o botones (paginación, despliegue de comentarios, etc.) y extraer información desde etiquetas HTML específicas. Esta automatización requiere el uso de herramientas capaces de simular un navegador real, interactuar con los elementos de la página y recuperar el contenido en bruto para su posterior tratamiento [29].

4.2.3. Obtención de metadatos

Objetivo del proceso

El objetivo de este apartado fue obtener automáticamente información básica de cada político incluido en el estudio, como su nombre, partido, comunidad autónoma, edad o número de legislaturas que ha ejercido. Todos estos datos fueron almacenados en la hoja *Metadata* del archivo principal de la base de datos.

Para ello, se diseñaron distintos scripts en Python que accedían a fuentes públicas como el portal del Congreso, Wikipedia o los perfiles de X/Twitter. Estos scripts se encargaban no solo de extraer los datos, sino también de organizarlos y validarlos antes de incorporarlos al archivo final, para así evitar tener que realizar funciones posteriores de limpieza. El uso de estas herramientas permitió completar la hoja *Metadata* sin necesidad de hacerlo a mano, lo que habría supuesto un esfuerzo muy elevado y con multitud de errores, especialmente al tratarse de decenas de perfiles.

Estrategia de obtención de datos

Dado que los datos provenían de diferentes fuentes y presentaban formatos diversos, se dividió el proceso de obtención en varios bloques:

- Un bloque específico para diputados del Congreso, con información extraída del portal oficial y datos adicionales de redes sociales.
- Un segundo bloque centrado en presidentes autonómicos, donde se combinaron fuentes como Wikipedia con información institucional.
- Y un último bloque destinado a líderes de la oposición, donde sólo se pudo automatizar la descarga de información de sus perfiles de X/Twitter. El resto de sus datos fue incorporado manualmente, ya que no existía una fuente fiable y centralizada desde la cual extraerlos de forma automatizada.

Fuentes utilizadas

- Portal del Congreso de los Diputados: para los diputados, se extrajeron automáticamente nombre completo, partido, provincia por la que salieron representados, comunidad autónoma y número de legislaturas. Además, a partir de la fecha de nacimiento se pudo calcular sus edades.
- Wikipedia: en el caso de los presidentes autonómicos, se utilizó la estructura de tablas disponibles para extraer los periodos de gobierno, partidos y datos biográficos.
- X/Twitter: se consultaron los perfiles oficiales para completar información relacionada con sus perfiles (número de seguidores, de posts, descripción y fecha en la que se creó la cuenta).

Almacenamiento y validación

Todos los datos recogidos se almacenaron inicialmente en archivos intermedios (en formato .json) que sirvieron como respaldo en caso de interrupciones o fallos durante la ejecución y también como una posibilidad para la adición de más políticos en caso de quererlo. Posteriormente, una vez validados, se integraron en la hoja *Metadata* del archivo Excel principal.

Herramientas empleadas

Para asegurar la calidad de los datos obtenidos, se aplicaron varias validaciones automáticas (como el formateo de fechas, la detección de duplicados y la comprobación del número de legislaturas) y se realizó una revisión manual final de la información para corregir posibles errores o inconsistencias. El lenguaje principal utilizado fue Python, debido a su sencillez y fácil uso en contextos de *data scraping*. Dichos procesos fueron divididos en varias clases específicas que trataran por separado la información de diputados, presidentes autonómicos y sus actividades en la red X/Twitter. También se hizo uso de una serie de librerías/herramientas que facilitaron la realización de los objetivos previstos, algunas de las cuales fueron empleadas en distintos procesos dentro de la fase 4.2 Recolección y organización de datos . Entre las distintas librerías empleadas en el desarrollo de este apartado de obtención de los metadatos destacan las siguientes:

- **Selenium WebDriver:** librería fundamental en este proyecto. Permite simular la navegación en un navegador real, interactuar con páginas dinámicas (como X/Twitter) y acceder a contenido que requiere scroll (paginación) o contenido que se genera dinámicamente con JavaScript. Fue utilizada para extraer datos tanto del portal del Congreso como de Twitter.
- **Requests y BeautifulSoup:** se usaron para recuperar, analizar y estructurar el contenido estático de la página web, como las tablas de presidentes autonómicos en Wikipedia. Permiten descargar HTML y extraer bloques de texto o enlaces mediante estructuras del tipo DOM (árbol jerárquico que organiza los elementos del HTML).

- **pandas:** empleada para manipular estructuras tabulares de datos, especialmente hojas de cálculo en formato Excel. Permitió cargar, actualizar y consolidar la información extraída desde múltiples fuentes.
- **openpyxl:** utilizada junto con pandas para guardar cambios y mantener el formato de los archivos Excel empleados como repositorio principal del dataset.
- **undetected-chromedriver:** versión modificada del controlador de Chrome que evita ser detectado por mecanismos antibot de sitios como X/Twitter. Se utilizó junto con Selenium para acceder a perfiles mediante sesiones autenticadas.
- **re (expresiones regulares):** empleada para identificar patrones en el texto, como fechas de nacimiento, años legislativos o formatos numéricos. Fue clave para estructurar datos extraídos de HTML y descripciones en X/Twitter.

Selenium WebDriver

Selenium WebDriver fue una de las herramientas principales utilizadas en este trabajo para automatizar el acceso y la recolección de datos en páginas web dinámicas. Esta librería permite simular el comportamiento de un usuario real en un navegador: abrir páginas, desplazarse por el contenido, hacer clic en elementos interactivos o extraer información del HTML generado mediante JavaScript. A diferencia de otras opciones que sólo leen el contenido HTML estático inicial, Selenium puede esperar a que la página se haya cargado por completo y luego extraer la información deseada, incluso en entornos complejos y con contenido que aparece progresivamente.

Esta capacidad de manejar el dinamismo de ciertas páginas web fue esencial para acceder a plataformas como X/Twitter o al portal del Congreso de los Diputados, donde la información no está disponible directamente en el HTML fuente y sólo aparece tras ejecutar scripts o interactuar con la interfaz. Selenium permitió acceder a este contenido de forma automática, sin depender de APIs oficiales que suelen tener limitaciones de acceso o estar sujetas a cambios frecuentes.

En un estudio comparativo, Lotfi et al. (2021) destacan Selenium como una de las herramientas más eficaces para proyectos de scraping en sitios con carga dinámica, gracias a su flexibilidad para navegar, extraer y controlar elementos generados en tiempo real. Subrayan que es especialmente adecuada cuando se necesita recuperar contenido interactivo o cargado por JavaScript, algo cada vez más común en la web moderna [34].

Asimismo, Gojare et al. (2015) señalan que Selenium WebDriver ha demostrado ser una solución robusta y versátil para la automatización web, con ventajas como su compatibilidad con múltiples navegadores y sistemas operativos, y su capacidad para manejar aplicaciones Ajax (aquellas que actualizan contenido sin recargar la página completa) [23]. Frente a enfoques manuales o basados en scripts independientes, Selenium facilita un flujo de trabajo más integrado, menos propenso a errores y mucho más escalable.

Motivos para su elección frente a otras herramientas:

- A diferencia de librerías como *BeautifulSoup* o *Scrapy*, que sólo funcionan con contenido HTML estático, Selenium permite interactuar con páginas que cargan información de forma dinámica mediante JavaScript, como ocurre en X/Twitter.
- Herramientas como *Puppeteer* o *Playwright* también permiten automatizar la navegación web, pero están pensadas principalmente para usarse con Node.js. En cambio, Selenium se adapta de forma directa a Python, el lenguaje empleado en este proyecto, lo que hizo más fácil su uso y configuración.

Requests y BeautifulSoup

Para obtener información de páginas web con contenido estático, se utilizaron las librerías Requests y BeautifulSoup, conocidas por ser fáciles de usar y de integrar en proyectos. Esta combinación fue especialmente útil en aquellos casos donde el contenido aparece directamente incrustado en el HTML al cargar la página, sin necesidad de ejecutar scripts adicionales, como ocurre con las tablas de Wikipedia que recopilan información sobre presidentes autonómicos.

La librería Requests permitió realizar peticiones HTTP a los servidores web y recuperar el código fuente de las páginas de manera rápida y fiable. Seguidamente, BeautifulSoup procesó y estructuró el código HTML, lo que facilitó localizar los datos relevantes mediante etiquetas, clases o rutas específicas. De esta manera, fue posible extraer, por ejemplo, los nombres de los presidentes, sus partidos o los periodos en los que ejercieron el cargo.

Según Abodayeh et al. (2023), esta combinación es especialmente recomendable para proyectos pequeños o moderadamente complejos, ya que permite realizar extracciones de datos con bajo consumo de recursos y sin necesidad de herramientas más pesadas como navegadores automatizados [2]. En su estudio, muestran cómo BeautifulSoup puede procesar múltiples estructuras HTML en cuestión de segundos, lo que la convierte en una opción ideal tanto para quienes se inician en el scraping como para quienes buscan soluciones reutilizables y mantenibles.

De forma complementaria, Lotfi et al. (2021) destacan en su revisión que Requests y BeautifulSoup siguen siendo una de las opciones más usadas en tareas de extracción puntual de información desde sitios web estáticos. También destacan que son fáciles de aprender, compatibles con otros módulos de análisis de datos y eficaces al integrarse en flujos de trabajo automatizados [34].

En este trabajo, esta solución fue más que suficiente para las fuentes públicas que no requerían interacción con el usuario a la hora de cargar datos, como las páginas de Wikipedia, permitiendo desarrollar scripts ágiles, ordenados y fácilmente escalables para nuevos casos.

pandas

La librería pandas fue la herramienta principal en este proyecto, ya que permitió trabajar de forma organizada y eficiente con las hojas del conjunto de datos. pandas es una de las bibliotecas más utilizadas en Python para la manipulación de datos en tablas, como los que se encuentran en hojas de cálculo, bases de datos o archivos .csv. Su estructura principal,

el DataFrame, permite manejar grandes volúmenes de datos de manera simple, aplicando operaciones complejas en pocas líneas de código.

En este trabajo, pandas fue fundamental para automatizar la gestión de las diferentes hojas del archivo Excel, permitiendo actualizarlas con información extraída de varias fuentes: la web del Congreso de los Diputados, Wikipedia y perfiles de X/Twitter. En esta etapa, se utilizó, entre otras cosas, para ubicar las filas correspondientes a cada político, completar campos vacíos con los datos correspondientes, corregir errores y guardar los cambios en el mismo archivo sin intervención manual.

Además de la lectura y escritura de hojas de Excel, pandas facilitó tareas como filtrar por cargos, combinar tablas intermedias con la hoja principal, y transformar datos de texto (como convertir listas de legislaturas a cadenas o normalizar nombres).

Según McKinney (2011), creador de la librería, pandas fue diseñada para ofrecer a los analistas una forma flexible y potente de trabajar con datos reales, permitiendo transformarlos, combinarlos o limpiarlos sin depender de herramientas externas como Excel o SQL [37]. De forma complementaria, Lotfi et al. (2021) también destacan su presencia en numerosos estudios de scraping, donde se emplea tras la recolección de datos para estructurarlos, analizarlos o exportarlos a formatos reutilizables [34].

Motivos para su elección frente a otras herramientas:

- Permite leer y escribir archivos Excel con gran facilidad, lo cual fue fundamental para mantener actualizada la hoja *Metadata*.
- Tiene una sintaxis muy clara que simplifica operaciones comunes como filtrar, rellenar, unir o agrupar datos.
- Es una herramienta ampliamente adoptada en proyectos de análisis de datos y scraping, lo que viene acompañado de abundante documentación.

openpyxl

La librería openpyxl se utilizó en este proyecto para guardar automáticamente los cambios realizados en el archivo Excel donde se almacenaban los resultados. Aunque pandas fue la herramienta principal para procesar y manipular los datos, no era suficiente por sí sola para escribir archivos Excel modernos sin perder formato. Por eso se combinó con openpyxl, una librería especializada en la lectura y escritura de archivos .xlsx, compatible con Microsoft Excel.

Su función principal en esta parte fue permitir la actualización automática de la hoja *Metadata*, que contenía información estructurada sobre cada político. Una vez ejecutados los distintos scripts de extracción desde fuentes externas (como el portal del Congreso, Wikipedia o X/Twitter), los datos obtenidos se insertaban directamente en las celdas correspondientes. Esto incluía campos como el partido político, la comunidad autónoma, la edad o el número de legislaturas, entre otros.

Gracias a openpyxl, el sistema pudo abrir el archivo Excel existente, modificar sólo la hoja

necesaria sin alterar el resto, conservar el formato original (como estilos, bordes o fórmulas) y guardar los cambios sin intervención manual. Esta automatización fue clave para mantener el conjunto de datos actualizado de forma continua y eficiente, reduciendo errores y evitando tareas repetitivas que, de hacerse a mano, habrían supuesto un aumento significativo del tiempo de trabajo. Esto encaja con los planteamientos de Lotfi et al. (2021) en los que se mencionan que, en muchos proyectos de scraping, el paso final suele implicar la exportación a formatos legibles y compartibles como Excel o CSV, donde herramientas como openpyxl y pandas juegan un papel fundamental para organizar y almacenar los datos extraídos de manera estructurada [34].

Undetected-Chromedriver

Undetected-Chromedriver es una herramienta diseñada para esquivar las restricciones que imponen muchas páginas web con el fin de bloquear el acceso automatizado mediante bots. A diferencia de los navegadores controlados por automatización tradicional, esta librería modifica ciertos comportamientos y aspectos de Chrome para hacerlo menos identificable como un navegador automatizado. De este modo, consigue sobrepasar mecanismos de detección basados en la inspección del *user-agent*, la ejecución en modo sin interfaz gráfica (*headless*) o el análisis de propiedades del entorno que suelen delatar a herramientas como Selenium.

En este proyecto, su uso fue especialmente relevante para acceder a plataformas como X/Twitter sin ser detectado ni bloqueado. Mientras que Selenium por sí solo puede ser identificado con facilidad cuando se realizan muchas acciones automatizadas en un corto periodo de tiempo, undetected-chromedriver permitió simular un comportamiento más humano y mantener sesiones activas durante más tiempo. Esto fue clave para recopilar metadatos como número de seguidores, publicaciones o fecha de creación del perfil de los políticos, tareas que se realizaban sobre una interfaz web dinámica y protegida.

Según Bale et al. (2022), este tipo de enfoque demostró ser uno de los más eficaces en su estudio comparativo sobre técnicas de scraping, donde se probaron distintos métodos sobre más de un centenar de sitios web. Lograron extraer contenido con éxito en 112 de los 120 sitios analizados, destacando por su capacidad para evitar bloqueos, mantener sesiones independientes y reducir errores en la extracción [6].

Además, aunque Lotfi et al. (2021) no hacen mención explícita a esta herramienta, sí enfatizan la importancia de adoptar técnicas que permitan respetar las políticas de acceso automatizado y sortear barreras técnicas impuestas por las páginas, especialmente en entornos donde el contenido se protege contra bots. En este sentido, undetected-chromedriver aporta una solución práctica y eficiente para superar estas barreras, manteniendo la compatibilidad con Selenium y los navegadores modernos [34].

Expresiones regulares (RegEx)

Las expresiones regulares, o *regex*, son una herramienta fundamental para buscar y extraer patrones específicos dentro de texto plano, como puede ser el código HTML de una página web. En este proyecto, se utilizaron principalmente para identificar y extraer fragmentos

concretos de texto estructurado, como fechas de nacimiento, años de legislatura o etiquetas HTML con clases determinadas, que resultaban necesarias para completar los datos de cada político en la hoja *Metadata*.

Este enfoque destaca por su velocidad y simplicidad. Según Uzun et al. (2018), el uso de expresiones regulares es una de las técnicas más eficientes en términos de tiempo de ejecución cuando se busca un único resultado bien definido, con un promedio de tan solo 0.071 milisegundos por operación [50]. Esto las convierte en una opción especialmente recomendable cuando el formato de los datos es predecible y no varía entre instancias.

Lotfi et al. (2021) también reconocen la utilidad de las expresiones regulares en procesos de scraping, particularmente cuando se trabaja con contenidos que no presentan una estructura totalmente definida. En su revisión, señalan que aunque otras técnicas pueden llegar a ser más apropiadas para estructuras complejas, las expresiones regulares destacan por su rapidez, bajo consumo de memoria y capacidad de adaptación en escenarios donde se necesita extraer información precisa y repetitiva de forma directa [34].

Motivos para su elección:

- Ofrecen una solución rápida y eficaz para localizar patrones conocidos en estructuras HTML claras y repetitivas.
- No es necesario analizar todo el árbol del documento, a diferencia de lo que ocurre con otras herramientas de análisis de HTML.
- Funcionan bien en páginas donde los datos deseados —como nombres, fechas o enlaces— siempre aparecen con la misma etiqueta o formato.

Además de las librerías ya mencionadas y descritas en detalle, se utilizaron otras herramientas complementarias, cuya función específica es clara y no requiere una explicación extensa:

- **webdriver-manager:** facilitó la gestión del controlador de Chrome, asegurando que Selenium fuera compatible con la versión del navegador instalada en el sistema.
- **os:** se utilizó para manejar rutas de archivos, verificar la existencia de carpetas o archivos, y crear directorios necesarios para guardar los resultados del scraping.
- **datetime:** permitió calcular edades a partir de fechas de nacimiento obtenidas en la web, lo que ayudó a enriquecer el conjunto de datos con información temporal.
- **time y random:** se emplearon para introducir pausas aleatorias entre solicitudes o acciones, simulando un comportamiento humano y reduciendo el riesgo de ser bloqueado por los servidores.
- **json:** permitió guardar y recuperar datos estructurados, como listas de políticos o resultados de análisis, en archivos intermedios. Gracias a su sintaxis simple y a que puede transformarse fácilmente en listas o diccionarios de Python sin necesidad de añadir librerías adicionales, resultó muy útil para tratar datos entre procesos, almacenarlos de forma organizada y analizarlos con otras herramientas como pandas.

Como apunte final, es importante mencionar que las herramientas mencionadas sirvieron para descargar la información solo para los presidentes de comunidades autónomas y los miembros del Congreso de los Diputados seleccionados. En el caso de los líderes de la oposición, al no existir una base de datos estructurada ni un repositorio unificado con su información institucional, solo se pudo automatizar la extracción de su perfil en X/Twitter. El resto de los datos asociados a estos perfiles se completaron de forma manual.

4.2.4. Obtención de posts

Objetivo del proceso

El objetivo de este proceso fue recoger de forma automática las publicaciones de los políticos en la red X/Twitter. Estos mensajes se guardaron en la hoja llamada *Posts*, que forma parte del archivo Excel general con toda la base de datos del proyecto.

Cada fila de esa hoja representa un tweet concreto, escrito por alguno de los políticos incluidos en el estudio. Para que esta asociación fuera posible, era importante que en la hoja *Metadata* estuviera registrado correctamente el nombre de usuario en X/Twitter de cada político. A partir de esa información, los distintos scripts creados accedían a sus perfiles y descargaban automáticamente sus publicaciones.

El objetivo final era tener un registro organizado de estos mensajes, incluyendo no solo el texto, sino también otros datos útiles como la fecha de publicación, el número de comentarios, retweets o "me gusta", y el enlace directo al tweet. Todo esto se guardó sin tener que hacerlo a mano, lo que permitió trabajar con muchos perfiles a la vez y mantener la hoja de *Posts* actualizada de forma rápida y eficiente.

Modos de ejecución

Durante la recolección de publicaciones en X/Twitter, se implementaron dos modos principales de ejecución para adaptarse a distintos escenarios de análisis:

- **Modo individual**: este modo permitía descargar los posts publicados por un solo político. Era útil para pruebas puntuales, tratamiento de errores o casos en los que se requería actualizar o volver a procesar un único perfil. Al tratarse de un único usuario, el proceso se ejecutaba en una sola instancia y sin necesidad de paralelización.
- **Modo grupal**: se utilizó cuando era necesario extraer información de varios perfiles al mismo tiempo. En este caso, el conjunto de políticos se dividía entre varios procesos paralelos, lo que permitía reducir significativamente el tiempo de ejecución. Cada proceso trabajaba con un subconjunto de usuarios y generaba su propio archivo de resultados temporal, los cuales se fusionaban de manera ordenada una vez terminaban todos los subprocesos.

Además, cada uno de estos modos podía funcionar bajo dos estilos distintos de extracción:

- **Por número de publicaciones**: se descargaba un número definido de tweets recientes por cada político, hasta alcanzar el límite indicado en la configuración. Este enfoque fue útil cuando se quería limitar la cantidad de datos recolectados o simplemente acceder a los tweets más actuales.
- **Por intervalo de fechas**: se especificaba una fecha de inicio y una de fin, y el sistema extraía únicamente las publicaciones que se encontraban dentro de dicho periodo. Este criterio resultó especialmente útil para realizar análisis centrados en campañas, eventos específicos o periodos de legislatura concretos.

  Para asegurar que sólo se descargaran publicaciones dentro del rango deseado, el proceso se encargaba de comparar la fecha de cada tweet con las fechas límite establecidas. Si se detectaba que el siguiente tweet era anterior al rango objetivo, se detenía el desplazamiento automático (scroll) y se finalizaba la extracción. No obstante, se incorporó una excepción importante: los tweets fijados.

  En muchos casos, los perfiles políticos mantenían en la parte superior de su cronología un tweet fijado que podía haber sido publicado muchos meses —o incluso años— antes del intervalo indicado. Si no se excluía correctamente este tipo de publicaciones, el sistema podía interpretar erróneamente que todas las publicaciones eran anteriores a la fecha mínima y, como consecuencia de esta interpretación errónea, interrumpir la recolección en ese mismo momento, sin llegar a haber descargado nada. Por ese motivo, se añadió una lógica específica que permitía detectar y omitir los tweets fijados cuando su fecha estuviera fuera del rango, asegurando así que no interfirieran en la descarga del resto del contenido.

Gracias a estas opciones de configuración y a los ajustes implementados para mejorar la precisión del filtrado por fecha, el sistema ofreció gran flexibilidad y permitió adaptar la recolección de posts a las necesidades concretas de cada fase del análisis.

Paralelización, automatización y gestión de sesiones

Para poder extraer publicaciones desde múltiples perfiles políticos en X/Twitter de forma rápida y continua, se diseñó un sistema completamente automatizado y paralelizado. El objetivo era reducir los elevados tiempos de ejecución y evitar bloqueos por parte de la plataforma.

Primero, se optó por una estrategia de ejecución en paralelo que permitiera repartir la carga de trabajo entre varios procesos simultáneos. Cada proceso se encargaba de un grupo diferente de políticos, asignados de forma equilibrada para mantener tiempos de descarga similares. Esto permitió ejecutar múltiples extracciones al mismo tiempo, acelerando significativamente el proceso de recolección de posts. Al finalizar cada subproceso, los resultados se almacenaban en archivos temporales que luego eran fusionados automáticamente en un único documento. Este archivo consolidado se integraba finalmente en la hoja *Posts*, evitando duplicados y manteniendo el formato estructurado previamente definido.

Para la navegación automatizada, se utilizó la librería Selenium WebDriver junto con

undetected-chromedriver, una herramienta previamente mencionada en la sección *Obtención de metadatos*, que permite evitar los mecanismos de detección de actividad automatizada implementados por plataformas como X/Twitter. Esta combinación hizo posible simular el comportamiento de un usuario real dentro del navegador: abrir perfiles, moverse a través de las publicaciones, cargar contenido generado de manera dinámica mediante JavaScript, y hacerlo sin ser bloqueado ni interrumpido por medidas de protección contra bots.

Con el objetivo de mejorar aún más la autenticidad del comportamiento simulado, se incorporaron acciones más humanas como pausas aleatorias y desplazamientos por la cronología mediante scrolls. Además, se evitó el uso intensivo del modo sin interfaz gráfica (*headless*), ya que su uso puede ser fácilmente detectado por ciertos sitios web como señal de automatización.

Uno de los aspectos clave para mantener la sesión activa y evitar restricciones fue la incorporación de cookies de sesión reales. Estas cookies, almacenadas previamente en archivos .json, permitieron simular accesos reales de usuarios autenticados. De esta manera, el sistema pudo visualizar contenido que normalmente estaría oculto para usuarios no registrados. Cada proceso paralelo utilizó un conjunto distinto de cookies, lo que ayudó a distribuir la actividad entre múltiples identidades y redujo el riesgo de ser bloqueado por exceso de tráfico desde una sola cuenta.

Gracias a esta combinación de paralelización, navegación automatizada y gestión de sesiones con cookies reales, fue posible ejecutar el scraping de publicaciones de forma estable, segura y eficiente, manteniendo la fidelidad de los datos obtenidos y reduciendo al mínimo las interrupciones o restricciones impuestas por la plataforma.

Filtrado y extracción de contenido

Durante la recolección de publicaciones de los políticos, se aplicaron varias medidas para filtrarlas con el fin de asegurar la calidad y la relevancia de los datos extraídos. Este proceso incluyó tanto el análisis del contenido visible en la interfaz como la validación de ciertos elementos clave en cada tweet antes de almacenarlo.

En primer lugar, se excluyeron de forma sistemática los retweets, tanto aquellos hechos por el propio político a sus publicaciones anteriores como los que correspondían a otras cuentas. El objetivo era centrarse exclusivamente en contenido original producido por los perfiles analizados. Además, también se descartaron los tweets que no incluían ningún texto —es decir, publicaciones compuestas únicamente por imágenes, vídeos, emojis o GIFs sin acompañamiento textual—, ya que no aportaban información útil desde un punto de vista lingüístico para el análisis posterior. Finalmente, se descartaron aquellas publicaciones cuyo contenido consistía exclusivamente en un enlace, ya que no aportaban información útil para el análisis textual.

De cada publicación que sí superaba estos filtros, se extrajeron los siguientes elementos:

- La **fecha de publicación**, utilizada tanto como campo informativo como elemento obligatorio para tratar con los límites temporales en las extracciones por periodo.

- El **contenido del tweet** que pasaba el filtrado.
- Las principales **métricas públicas**: número de comentarios, número de retweets y número de "me gusta".
- El **enlace directo a la publicación**, que permite acceder a ella desde cualquier navegador para su verificación manual o revisión posterior.

En los casos donde se activaba la extracción por intervalo temporal, se añadía una condición extra a la extracción. Esta consistía en detener automáticamente el desplazamiento por la cronología en cuanto se identificaba una publicación cuya fecha era anterior a aquella que se había especificado como límite inferior. No obstante, fue necesario tener en cuenta una excepción importante: los tweets fijados (o *pinned tweets*). Como ya se mencionó anteriormente, estas publicaciones, aunque aparecen al principio del perfil, pueden pertenecer a fechas muy anteriores y, si no se filtraban correctamente, provocaban una falsa condición de finalizado del proceso de extracción que impedía obtener los tweets válidos del intervalo deseado de tiempo. Por tanto, se implementó una detección específica para omitirlos.

Gracias a estos mecanismos de filtrado y control, se consiguió que el conjunto de publicaciones recopiladas respondiera únicamente a los criterios establecidos, evitando posts duplicados, contenido irrelevante y errores de interpretación temporal.

Salida y consideraciones finales

Una vez finalizado el proceso de recolección, las publicaciones extraídas se almacenaron de forma estructurada en archivos temporales en formato Excel, uno por cada subproceso en el caso de ejecuciones paralelas. Estos archivos fueron posteriormente fusionados en una única hoja denominada *Posts*, integrada dentro del archivo principal que recoge toda la base de datos del proyecto.

Cada fila de esta hoja representa un tweet individual y contiene información similar a esta:

- **ID_Político**: identificador numérico único para cada uno de los políticos, siendo este el mismo que se utilizó en la hoja *Metadata*.
- **Enlace_Post**: URL directa al tweet original en X/Twitter.
- **Contenido**: texto del tweet extraido.
- **Fecha_Publicación**: fecha exacta en la que fue publicado el tweet.
- **Métricas**: número de comentarios, retweets y "me gusta" en el momento de la extracción.

Además de las anteriores variables que se han mencionado, para cada tweet obtenido se le adjuntó el valor inicial 0 a la columna Comentarios_Extraidos. Con esto se pretendía mantener un conteo del número de comentarios que se obtuvieran en sus procesos de extracción pertinentes.

Desde el inicio del proceso, surgieron diversas dificultades técnicas que fue necesario resolver para asegurar un uso responsable de la automatización. En primer lugar, se evitó sobrecargar los servidores de X/Twitter mediante pausas aleatorias, tiempos de espera entre acciones y una ejecución escalonada de los procesos. Además, al acceder como usuarios autenticados mediante cookies previamente generadas, se respetaron las restricciones públicas de la plataforma sin comprometer sus mecanismos de seguridad.

También se asumieron ciertas limitaciones relacionadas a este tipo de extracción:

- El uso de cookies implicaba una caducidad temporal, por lo que algunas sesiones pudieron perder validez con el tiempo, lo que requería la actualización manual de estas.
- El sistema no accedía a contenidos que requerían interacción compleja o elementos cargados fuera del alcance del scroll simulado. Debido a esto, se podía dar el caso de que el programa pudiera toparse con un "límite" de posts que se podían cargar. Por lo tanto, fue necesario establecer un contador que fuese aumentando por cada deslizamiento de pantalla que no devolviera nuevas publicaciones hasta llegar a un límite. Una vez se llegase a dicho tope, se terminaría la extracción del perfil que se estuviese tratando en ese momento.

En conjunto, estos mecanismos y precauciones permitieron llevar a cabo la recolección de forma robusta, siguiendo con las restricciones presentadas y adaptada a los márgenes técnicos realistas que impone una plataforma como X/Twitter.

Herramientas adicionales empleadas

Además de las librerías ya detalladas en el apartado de *Obtención de metadatos*, durante el proceso de obtención de publicaciones se incorporaron otras herramientas complementarias que facilitaron la realización de tareas específicas:

- **multiprocessing:** permitió ejecutar varios procesos simultáneamente en paralelo. Esta librería estándar de Python facilita dividir una tarea en procesos independientes que pueden ejecutarse de manera simultánea. A diferencia de los hilos *threads*, los procesos no están limitados por el Global Interpreter Lock (GIL). Este mecanismo permite que solo un hilo *thread* ejecute código Python a la vez, por lo que, al no estar limitado por esto, se consigue mejorar la eficiencia en tareas intensivas. Según Aziz et al. (2021), la librería multiprocessing ofrece una solución efectiva y escalable para acelerar tareas computacionales mediante subprocesos independientes, lo cual resulta ideal en entornos con múltiples núcleos de CPU y permite mejorar significativamente el rendimiento en recolección de datos o procesamiento masivo [5].
- **unicodedata:** utilizada para normalizar texto antes de convertir abreviaciones numéricas (como "2,5 mil" o "3.2M") en valores enteros. Gracias a esta librería, se pudieron interpretar correctamente diferentes formatos de escritura y evitar errores en la conversión de texto a valores numéricos.
- **typing (List):** facilitó la definición de las estructuras de datos utilizadas, como las listas de cookies de sesión. Esta librería permitió una gestión más segura de los datos

durante su manipulación.

Estas herramientas ayudaron en la realización de procesos claves en etapas puntuales de procesamiento, extracción y control de datos.

4.2.5. Obtención de comentarios

Objetivo del proceso

El objetivo principal de este proceso fue obtener los comentarios que otras personas dejaron en respuesta a los tweets publicados por los políticos incluidos en el estudio. Estos comentarios se almacenaron en una hoja específica llamada *Comentarios*, dentro del mismo archivo Excel que recoge toda la base de datos del proyecto.

Cada fila de esta hoja representa un comentario individual relacionado con un tweet concreto. Para poder asociarlos correctamente, el sistema se basó en los enlaces a los tweets que ya estaban registrados en la hoja *Posts*. A partir de ahí, los distintos scripts automatizados accedían a cada publicación y recopilaban los comentarios disponibles más recientes.

La extracción de estos comentarios no solo permitió conocer las reacciones del público a los mensajes políticos, sino que también hizo posible detectar si el propio autor del tweet (el político) había respondido a alguno de ellos. Este dato fue especialmente útil para analizar interacciones directas entre políticos y ciudadanos.

En resumen, este proceso permitió ampliar la información disponible para cada tweet, añadiendo un nivel de análisis más profundo sobre la conversación que generaron y las respuestas que provocaron. Todo ello se hizo de manera automatizada, lo que facilitó la recolección de grandes cantidades de datos en poco tiempo y sin necesidad de intervención manual.

Paralelización y distribución de tareas

Dado que cada publicación política podía tener decenas o incluso cientos de comentarios, extraer esta información de forma secuencial habría llevado demasiado tiempo. Para resolver este problema y, al igual que como se hizo con la obtención de los posts, se diseñó un sistema que divide la tarea entre varios procesos que trabajan en paralelo. Cada uno de ellos se encargaba de un grupo distinto de tweets, permitiendo que se analizaran muchos enlaces al mismo tiempo y acelerando así todo el proceso.

Este reparto se hizo de forma equilibrada: se tomaron todos los enlaces a los tweets recopilados previamente y se dividieron en partes iguales, una para cada proceso. De este modo, cada subproceso tenía una carga de trabajo similar y se aprovechaban mejor los recursos del ordenador, especialmente en equipos con varios núcleos de procesamiento.

Una vez que cada proceso terminaba de extraer los comentarios asignados, guardaba los resultados en un archivo temporal. Al finalizar toda la ejecución, esos archivos fueron

fusionados automáticamente en una sola hoja llamada *Comentarios*, donde se integraron todos los datos recogidos. Finalmente, se actualizó la columna Comentarios_Extraidos de la hoja *Posts* con el nuevo número de comentarios obtenidos para cada post.

Esta estrategia no solo hizo posible trabajar de manera más rápida y eficiente, sino que también permitió mantener el control sobre qué comentarios ya habían sido procesados y evitar repeticiones innecesarias.

Navegación automática y acceso a los comentarios

Para poder acceder a los comentarios de cada publicación, el sistema necesitaba visitar directamente cada tweet dentro del navegador, igual que lo haría una persona. Esto se hizo mediante navegación automatizada, utilizando herramientas que permiten simular la interacción de un usuario real con la página.

Una vez más se emplearon dos librerías: Selenium WebDriver y undetected-chromedriver. La primera permitió controlar el navegador automáticamente (abrir páginas, hacer scroll, hacer clic, etc.), mientras que la segunda ayudó a evitar que la web detectara que no se trata de una persona real, sino de un sistema automatizado.

De la misma manera que en la extracción de los posts, para que la navegación fuera más creíble y evitara bloqueos por parte de la plataforma, se añadieron pausas aleatorias entre acciones y desplazamientos por la página que imitaban el comportamiento humano. Además, el sistema cargaba previamente cookies reales de sesión, lo que le permitía entrar como si se tratara de un usuario autenticado. Esto era importante porque algunos comentarios solo están visibles cuando se accede con una cuenta registrada.

Gracias a esta estrategia, la cual presenta la misma estructura que en la obtención de posts, el sistema pudo explorar cada tweet, cargar su contenido dinámico y acceder a los comentarios sin interrupciones. Así, se consiguió que el proceso fuera más estable, evitando limitaciones impuestas por la plataforma.

Extracción y filtrado del contenido

Una vez que el sistema accedía al tweet, comenzaba la extracción de los comentarios visibles. Sin embargo, no se trataba simplemente de guardar cualquier respuesta: se aplicaron una serie de filtros y condiciones para asegurar que la información recopilada fuera realmente útil para el análisis.

En primer lugar, se descartaban aquellos comentarios que no tuvieran texto. Por ejemplo, si un usuario respondía solo con un vídeo, una imagen o un emoji, ese comentario no se consideraba relevante y no se guardaba. Además, también se evitaban duplicados, es decir, comentarios que ya hubieran sido extraídos anteriormente. También, al igual que en la extracción de posts, se omitieron los comentarios cuyo contenido consistía únicamente en un enlace, ya que no aportaban información significativa para el análisis.

El objetivo era obtener un conjunto concreto de respuestas por cada publicación —por ejemplo, 2 o 3 comentarios representativos—, y siempre se daba prioridad a los primeros comentarios visibles. En algunos casos, si el autor del tweet respondía directamente a alguien (por ejemplo, aclarando o contradiciendo algo), el sistema intentaba identificar esa respuesta para poder guardarla junto con el comentario original. Si no había respuesta del autor, se ponía directamente "No". Cabe mencionar que aquellos comentarios que recibían respuestas por parte de los políticos se mostraban como los primeros, lo que facilitó la extracción de esta clase de mensajes.

Para cada comentario seleccionado, se extrajo la siguiente información:

- El enlace directo al comentario.
- El contenido textual de la respuesta.
- La fecha en la que se publicó.
- Las métricas visibles: número de comentarios, retweets y "me gusta".
- Una etiqueta indicando si el autor original del tweet había respondido a ese comentario.

El sistema realizaba desplazamientos automáticos por la página (scroll) para ir cargando nuevos comentarios si aún no se había alcanzado la cantidad deseada. Sin embargo, se estableció un límite de intentos para evitar que el sistema quedara bloqueado en publicaciones con pocos comentarios disponibles.

Gracias a este proceso, se pudo construir una hoja de datos con respuestas reales de los usuarios, bien organizadas, filtradas y listas para su análisis posterior.

Salida y cierre del proceso

Una vez finalizada la extracción de comentarios, cada proceso guardaba los resultados de forma local en un archivo temporal. Este archivo contenía todos los comentarios válidos obtenidos por ese subproceso, con las columnas ya organizadas para facilitar su integración posterior.

Al terminar todos los procesos, el sistema reunía automáticamente esos archivos temporales en la hoja *Comentarios*, integrada en el archivo general del proyecto. Durante esta fusión se revisaban posibles duplicados y se mantenía la estructura definida previamente.

Cada fila de la hoja *Comentarios* representa un comentario único, con los siguientes campos:

- **Enlace_Post**: enlace del tweet original al que responde el comentario.
- **Enlace_Comentario**: enlace directo al comentario extraído.
- **Contenido**: texto del comentario, limpio y legible.
- **Fecha_Publicación**: fecha en la que se publicó la respuesta.

- **Comentarios**, **Retweets**, **Likes**: métricas asociadas al comentario en el momento de su extracción.
- **Respuesta**: campo que indica si el autor del tweet respondió a ese comentario. En el caso de que hubiera respondido se añadía el contenido de su respuesta, mientras que para aquellos que no recibieron respuesta se puso un "No" en la celda.

Además, se actualizó automáticamente la hoja *Posts* para reflejar cuántos comentarios habían sido extraídos de cada publicación. Esto se hizo gracias a un conteo que revisaba cuántas veces aparecía cada tweet como referencia en la hoja *Comentarios*.

Para que todo este proceso funcionara correctamente, también se tuvieron en cuenta algunos aspectos técnicos importantes:

- Se utilizaron pausas aleatorias y navegación realista para evitar bloqueos por parte de X/Twitter.
- Se trabajó con sesiones de usuario autenticadas mediante cookies, lo que permitió acceder a contenido visible solo para cuentas registradas.
- Se establecieron límites en el número de desplazamientos y en el tiempo de espera, para no forzar innecesariamente las cargas de contenido y evitar errores.

Por último, cabe señalar que para la obtención de comentarios no fue necesario incorporar herramientas adicionales a las ya utilizadas en los procesos previos de obtención de metadatos y publicaciones. La estructura, los mecanismos de paralelización y la navegación automatizada fueron los mismos, lo que facilitó la integración del sistema y redujo su complejidad general.

4.3. Limpieza y traducción

Limpieza de los datos extraídos

A diferencia de muchos proyectos de análisis textual donde es necesario llevar a cabo uno o más procesos de limpieza de los datos antes de poder utilizarlos, en este caso esa etapa resultó mínima gracias al diseño previo del sistema de recolección. Desde un principio, a los scripts encargados de extraer publicaciones y comentarios de X/Twitter se les añadieron múltiples filtros que permitieron garantizar que el contenido obtenido fuera útil, relevante y apto para su análisis directo.

Por ejemplo, durante la descarga de publicaciones se filtraron automáticamente los tweets que eran retweets (ya fuera de otros usuarios o del propio político), así como aquellos que no contenían texto, es decir, publicaciones formadas únicamente por imágenes, vídeos, GIFs o enlaces externos. De este modo, se garantizaba que cada entrada recogida incluyera suficiente contenido lingüístico para poder ser analizada posteriormente.

En el caso concreto de la extracción por intervalo de fechas, también se aplicó un filtro específico para evitar descargar tweets fijados cuya fecha real estuviera fuera del rango temporal indicado. Esto fue necesario porque muchos perfiles mantenían publicaciones ancladas en la parte superior, que podían inducir a error si no se descartaban adecuadamente. Solo se permitía conservar estos tweets si su fecha coincidía efectivamente con el período de interés.

Por su parte, en la extracción de comentarios también se omitieron respuestas vacías, repetidas o que no añadían valor al análisis. Además, el sistema descartaba automáticamente los comentarios que no contenían texto alguno, como aquellos que solo incluían vídeos o emojis sin acompañamiento textual.

A pesar de todas estas medidas preventivas, se incluyó un único paso adicional de limpieza posterior. Este consistió en la detección y eliminación de posibles duplicados tanto en los comentarios como en los posts, en el remoto caso de que alguna entrada hubiera logrado pasar los filtros previos. Para ello, se crearon procedimientos que revisaban los enlaces únicos de cada publicación y comentario, eliminando automáticamente cualquier fila duplicada dentro de las hojas correspondientes del archivo Excel.

Gracias a todas estas precauciones incorporadas desde la fase de recolección, y a la verificación final mediante el control de duplicados, el conjunto de datos llegó a la etapa de análisis prácticamente limpio, sin necesidad de aplicar más transformaciones sobre el contenido textual original.

Detección del idioma original

Dado que los textos analizados provenían de políticos de diferentes regiones de España —algunas con varias lenguas cooficiales— fue necesario identificar el idioma en el que estaba escrito cada mensaje antes de proceder a su traducción. Esta identificación evitó traducir textos que ya estaban en español y permitió seleccionar correctamente aquellos que sí requerían ser traducidos, como los escritos en catalán, gallego, vasco o inglés.

Para llevar a cabo esta tarea se utilizó la librería langdetect, una herramienta de Python que detecta automáticamente el idioma de un texto. Su funcionamiento se basa en modelos estadísticos (*Naïve Bayes*) entrenados con patrones de letras y palabras (*n-gramas*).

Naïve Bayes

Naïve Bayes es un modelo probabilístico que se usa con frecuencia para tareas de clasificación automática, como detectar el idioma de un texto. Se basa en el *teorema de Bayes*, el cual devuelve la probabilidad de que un texto pertenezca a una clase c sabiendo las evidencias observadas E:

$$P(c \mid E) = \frac{P(E \mid c) \cdot P(c)}{P(E)}$$

Para hacerlo más sencillo y rápido de calcular, se parte de la idea de que las características del texto (como palabras o fragmentos) son independientes entre sí cuando se conoce la clase. Aunque en realidad estas características no sean totalmente independientes, asumirlo facilita mucho los cálculos y, además, suele dar buenos resultados, ya que lo que importa es encontrar la clase más probable y no tanto conocer la probabilidad exacta [41].

En este proyecto se usó este modelo a través de la librería langdetect, que aprovecha Naïve Bayes combinado con patrones de *n-gramas* para identificar rápidamente el idioma de textos cortos, sin necesidad de entender el significado completo.

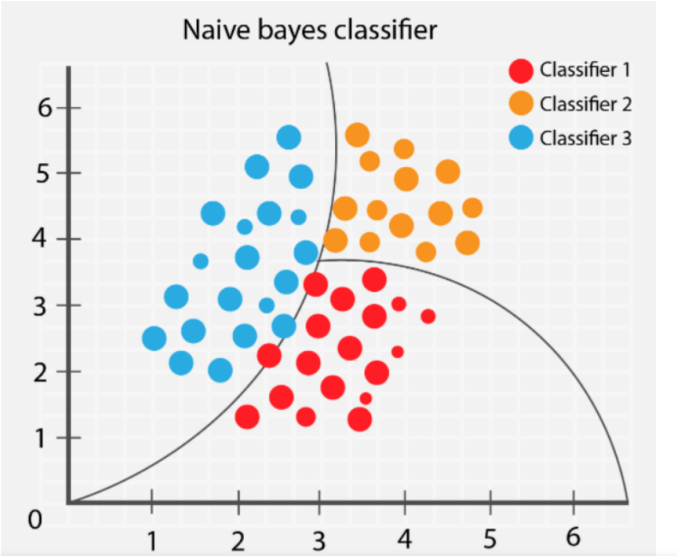

Ilustración 4.2: Ejemplo de clasificación multiclase con Naïve Bayes. Fuente: Tovar (2022), Huawei Forum.

n-gramas

Los modelos de *n-gramas* sirven para estimar la probabilidad de que aparezca una palabra (o un grupo de caracteres) basándose en las palabras anteriores. Normalmente se asume que solo dependen de las $n-1$ palabras previas. Por ejemplo, en un trigrama se tiene en cuenta el contexto de las dos palabras anteriores:

$$P(w_i \mid w_{i-2}, w_{i-1})$$

Este enfoque ayuda a capturar secuencias típicas de cada idioma (como “th” en inglés o “ll” en español) sin analizar toda la gramática de forma profunda. Gracias a esta estrategia, herramientas como langdetect pueden reconocer el idioma de textos breves de forma sencilla y eficaz.

En este proyecto, langdetect permitió clasificar publicaciones y comentarios en más de 50 idiomas, identificando aquellos distintos al español para traducirlos posteriormente. Se valoró

su facilidad de integración y velocidad, así como un rendimiento equilibrado en textos breves [26].

Traducción automática del contenido

Después de identificar el idioma en que estaba escrito cada texto, el siguiente paso fue traducir todos aquellos que no estaban en español. El objetivo de esta traducción fue tener todos los textos en una misma lengua para poder analizarlos de forma más sencilla y coherente. Si no se hiciera esto, trabajar con fragmentos en distintos idiomas podría generar confusión a la hora de clasificar el contenido de los posts y comentarios, derivando así en resultados finales negativos.

La herramienta utilizada para llevar a cabo esta tarea fue deep_translator, una librería de Python que permite acceder de manera sencilla al traductor de Google. Gracias a ella, fue posible traducir automáticamente los textos desde su idioma original al español, sin necesidad de usar directamente la API oficial de Google Translate. La librería soporta una amplia variedad de idiomas y está especialmente pensada para tareas como esta, donde se requiere traducir grandes cantidades de texto de manera automatizada.

La lógica fue simple: si el texto ya estaba en español, se dejaba tal cual; si estaba en otro idioma, se traducía. En algunos casos concretos —como el asturiano o el romaní—, la traducción no fue posible porque esos idiomas no están disponibles en el servicio. En esas situaciones, se decidió conservar el contenido original.

A pesar del buen rendimiento del traductor de Google, cabe destacar que la traducción automática puede no ser perfecta, ya que factores como los "hashtags", diminutivos o palabras mal escritas afectan a la hora de entender el contexto de la frase. Aún así, se consideró lo suficientemente precisa para los fines del análisis. En general, este proceso permitió unificar los datos bajo un mismo idioma, evitando problemas derivados de trabajar con varias lenguas a la vez y facilitando los pasos posteriores del estudio.

4.4. Etiquetado y preparación del corpus

4.4.1. Objetivo del etiquetado

El proceso de etiquetado se llevó a cabo con el objetivo de preparar un subconjunto de mensajes lo suficientemente grande como para que pudiera ser utilizado en fases posteriores de entrenamiento del modelo de clasificación. Dado que muchas técnicas de aprendizaje automático requieren conjuntos de datos previamente anotados para aprender a reconocer patrones, era necesario clasificar manualmente una parte del conjunto de publicaciones y comentarios. Esta clasificación permitió establecer una pequeña base de datos confiable a partir de la cual entrenar y evaluar modelos capaces de categorizar automáticamente el resto de posts y comentarios.

Además, disponer de datos etiquetados facilitó el estudio de la distribución de tonos y temáticas en el discurso político en redes sociales, permitiendo explorar no solo qué se dice, sino cómo se dice, y sobre qué se dice.

4.4.2. Dimensiones del etiquetado

Para llevar a cabo el análisis supervisado, se procedió al etiquetado manual de una parte del corpus. Dicho etiquetado se estructuró en torno a varias dimensiones fundamentales:

- **Tono**: Esta dimensión clasifica el contenido en función de su carga emocional o valorativa. Dicho etiquetado fue utilizado tanto para los posts como para los comentarios y las posibles respuestas que estos obtuvieran. Se establecieron tres categorías principales: *positivo*, *negativo* y *neutro*. Un texto se considera positivo si expresa aprobación, apoyo o entusiasmo; negativo si transmite crítica, rechazo o malestar; y neutro si se limita a informar o describir sin llegar a expresar una postura u opinión clara.

  Una de las decisiones más importantes a la hora de etiquetar el corpus fue incluir la categoría *neutro* dentro de la clasificación por tono. En muchas ocasiones, especialmente en redes sociales, los mensajes no expresan ni emociones positivas ni negativas, sino que simplemente informan, describen o hacen afirmaciones sin mostrar una postura clara. Ignorar esta clase de contenidos puede dar lugar a interpretaciones forzadas o incluso a errores en el análisis.

  Varios estudios han señalado esta misma limitación del enfoque tradicional binario. Por ejemplo, González-Pizarro et al. (2021) advierten que muchas interacciones en redes tienen una polaridad difícil de clasificar, lo que justifica la necesidad de una categoría intermedia. Además, investigaciones más técnicas sobre análisis de sentimientos subrayan que la neutralidad no solo es común, sino que puede afectar a la precisión de los modelos si no se tiene en cuenta correctamente [24, 48].

  Por esta razón, se optó por añadir esta etiqueta como su propia categoría, con el objetivo de reflejar con mayor grado de detalle la variedad de tonos presentes en los mensajes, y evitar que el modelo "forzara" una clasificación emocional errónea a falta de esta tercera categoría.

- **Temática**: Esta etiqueta recoge el tema principal del mensaje. Para definir estas categorías, se revisaron diferentes métodos descritos en otros estudios. Algunos de estos han utilizado enfoques automáticos como LDA o BERTopic para agrupar tweets de forma no supervisada [49, 55], pero estos métodos suelen generar grupos difíciles de interpretar y dependen demasiado de las palabras concretas del corpus, lo que hace complicado compararlos a lo largo del tiempo. Sin embargo, otros trabajos optan por definir categorías temáticas fijas y asignarlas manualmente [32, 27, 4], lo que facilita entender y analizar los datos de forma más consistente.

  En este proyecto se siguió esta segunda estrategia, partiendo de un conjunto más amplio de categorías, pero se observó que algunas solo contenían 3 o 4 publicaciones, lo que resultaba poco útil y podía generar problemas para entrenar modelos automáticos.

Por eso se decidió agrupar temas similares hasta quedarse con cuatro grandes bloques, que recogen bien la variedad de contenidos políticos, pero evitan etiquetas demasiado pequeñas.

- **Gestión Pública e Instituciones**: Incluye publicaciones centradas en el discurso político, los posicionamientos ideológicos, los eventos de los partidos, las relaciones institucionales o la defensa de las leyes. Esta categoría recoge el relato político e institucional de los políticos.
- **Economía, Empresa, Empleo e Infraestructuras**: Agrupa mensajes relacionados con datos económicos, actividad empresarial, inversiones y mejoras en infraestructuras. Refleja tanto medidas económicas a gran escala como proyectos con impacto local o regional.
- **Sociedad, Igualdad y Derechos**: Trata temas como igualdad de género, derechos sociales, inclusión, memoria histórica, sanidad, educación o cultura entendidas como derechos. Es el apartado humano y ético del discurso político.
- **Otros**: Recoge publicaciones que no se relacionan con ninguna temática clara. Incluye mensajes personales, emotivos, anecdóticos, humorísticos o sin un contenido político definido.

4.4.3. Tamaño y formato del corpus anotado

Durante el proceso de etiquetado se clasificaron manualmente unos 900 posts y alrededor de 1700 comentarios. Esta anotación se hizo siguiendo las dos dimensiones principales ya explicadas: el tono del mensaje y el tema del que trata.

En cuanto al tono, la mayoría de los textos fueron clasificados como negativos (1133 ejemplos), seguidos por los positivos (933 ejemplos) y, en menor cantidad, los neutros (505 ejemplos).

Con respecto a la temática, solo se usaron los posts de los políticos, los cuales se agruparon en cuatro grandes bloques: *Gestión Pública e Instituciones* fue la más común (342 ejemplos), seguida por *Economía, Empresa, Empleo e Infraestructuras* (238), *Sociedad, Igualdad y Derechos* (203) y, por último, *Otros* (136), que recoge mensajes difíciles de clasificar en los anteriores.

Todos estos datos se almacenaron en dos archivos de Excel: uno con la clasificación por tono, que incluyó tanto publicaciones como comentarios, y otro solo con las publicaciones para la clasificación por tema. En ambos casos, el formato era simple: cada fila contenía el texto original y su etiqueta (de tono o temática), lo que facilitaba su uso para entrenar modelos de análisis automático.

4.5. Entrenamiento y ajuste de modelos

4.5.1. Selección del modelo base

El modelo seleccionado como punto de partida fue SaBERT-Spanish-Sentiment-Analysis, un clasificador de sentimiento en español basado en la arquitectura BERT. Fue desarrollado como parte de un proyecto de Ingeniería Informática en la Universidad de Buenos Aires (UBA), y entrenado sobre un conjunto de 11.500 tweets en español procedentes de varias regiones. En su configuración original, el modelo clasificaba únicamente entre sentimiento positivo y negativo, y alcanzó un 86,47 % de precisión en su clasificación.

Se optó por este modelo debido a su especialización en textos en español y su buen rendimiento en tareas similares (análisis de sentimiento en redes sociales). Al haber sido entrenado con un corpus que contenía una gran variedad de ejemplos reales de tweets, se consideró un punto de partida adecuado para expandir su uso a tareas de clasificación más desarrolladas y con más etiquetas, como las abordadas en este trabajo, que también parten de mensajes, por lo general cortos y poco formales, sobre todo en el caso de los comentarios.

No se consideró necesario probar distintas arquitecturas o variantes de modelos preentrenados por dos motivos principales. Por un lado, el procesamiento de lenguaje natural en español es todavía un campo en desarrollo, con menos opciones disponibles en comparación con el inglés. Por otro lado, se prefirió seguir un enfoque práctico, eligiendo directamente un modelo que ya estuviera entrenado y ofreciera una buena base para ajustarlo a las necesidades concretas de este proyecto. El modelo seleccionado —SaBERT-Spanish-Sentiment-Analysis— estaba diseñado específicamente para detectar sentimientos en tweets en español, lo que encajaba perfectamente con el tipo de textos que se quería analizar. Esta semejanza entre el modelo preentrenado y los que se pretendían obtener lo convirtió en una opción muy adecuada para empezar a trabajar.

BERT

BERT (*Bidirectional Encoder Representations from Transformers*) es un modelo de lenguaje desarrollado por Google que revolucionó el procesamiento de lenguaje natural (PLN) al introducir una comprensión profunda del contexto de las palabras dentro de una oración. Su principal innovación consiste en analizar el texto en ambas direcciones —de izquierda a derecha y de derecha a izquierda— a la vez, lo cual permite entender con mayor precisión el significado de cada palabra según las que la rodean [56, 30].

Este modelo se basa en una arquitectura llamada **Transformer**, introducida en 2017, que eliminó el uso de redes recurrentes y convolucionales [51].

Las redes recurrentes (RNN, por sus siglas en inglés) fueron durante mucho tiempo el enfoque más común para procesar texto. Funcionaban leyendo palabra por palabra, en orden, lo que les permitía recordar lo que venía antes y tener cierta idea del contexto. Pero ese proceso

era lento y poco eficiente, sobre todo cuando el texto era largo, ya que la memoria del modelo se iba diluyendo y perdía precisión con el tiempo.

Por otro lado, las redes convolucionales (CNN), que originalmente se usaron para analizar imágenes, también se probaron en el procesamiento de lenguaje. Eran buenas para detectar patrones o combinaciones frecuentes de palabras —por ejemplo, expresiones típicas— pero no entendían bien la relación entre palabras que estaban lejos unas de otras en la oración. Por eso, aunque aportaron ideas útiles, tampoco lograban capturar todo el significado del texto.

El Transformer resolvió esas limitaciones al usar un mecanismo conocido como *self-attention*. Este mecanismo permite que cada palabra tome en cuenta a todas las demás del texto, para entender mejor su significado según el contexto.

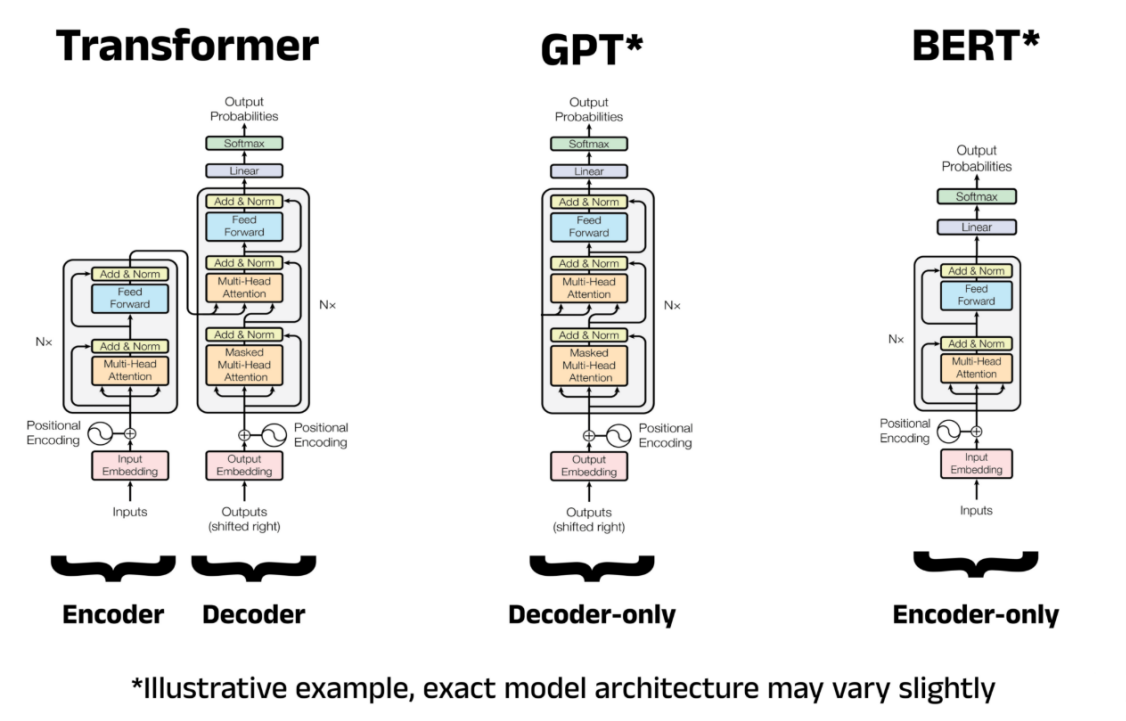

Ilustración 4.3: Comparativa entre la arquitectura Transformer completa, GPT (basado solo en decodificadores) y BERT (basado solo en codificadores). Imagen adaptada de Towards Data Science [43].

En la ilustración anterior puede observarse cómo BERT utiliza únicamente la parte izquierda del Transformer —el encoder o codificador—. A diferencia de modelos generativos como GPT, BERT no produce texto nuevo, sino que se enfoca en entender el texto de entrada para tareas como análisis de sentimiento, clasificación de frases o respuesta a preguntas.

Entrenamiento de BERT. Antes de poder resolver tareas concretas, BERT se entrena en dos tareas generales:

1. **Masked Language Modeling (MLM)**: Se seleccionan aleatoriamente algunas pa-

labras del texto y se reemplazan por un marcador especial (`[MASK]`). El modelo debe predecir cuál era la palabra original, basándose en las palabras que la rodean.

2. **Next Sentence Prediction (NSP)**: Se presentan dos oraciones al modelo, y este debe decidir si la segunda mantiene sigue teniendo sentido con respecto a la primera. Esto le ayuda a capturar relaciones entre frases enteras, no solo palabras.

Ambas tareas se combinan durante la etapa de preentrenamiento de BERT, lo que le permite aprender en profundidad las características del lenguaje, incluso antes de ser adaptado a tareas específicas como análisis de sentimientos o clasificación de frases.

Atención multi-cabeza. La clave del funcionamiento de BERT es el mecanismo de *self-attention*, que permite que cada palabra preste atención a otras partes del texto. Su funcionamiento básico se define mediante la fórmula:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^\top}{\sqrt{d_k}}\right)V$$

donde:

- Q (*queries*), K (*keys*) y V (*values*) son distintas formas de representar cada palabra para que el modelo pueda decidir en qué otras palabras debe fijarse. Se puede entender como que cada palabra hace una pregunta (Q), y luego compara esa pregunta con las "claves" (K) de todas las demás. Según lo bien que coincidan, toma algo de información (V) de esas palabras. Así, el modelo puede enfocarse en lo más importante del contexto para cada palabra.
- d_k es simplemente el tamaño de los vectores de las claves (K). Se usa para evitar que los números se vuelvan demasiado grandes cuando se comparan las preguntas (Q) con las claves (K). Al dividir por $\sqrt{d_k}$, se mantienen los valores en un rango razonable, lo que ayuda a que la función *softmax* funcione bien y no se descontrole.
- El resultado final es una mezcla de la información (V) de todas las palabras del texto, pero no todas pesan lo mismo. El modelo le da más importancia a las palabras que cree que son más útiles para entender el significado de la palabra que está analizando. Esta forma de "prestar atención" a las partes más relevantes del texto es lo que hace que el mecanismo de *self-attention* sea tan eficiente.

Este proceso se realiza de forma paralela en múltiples "cabezas" para captar distintos tipos de relaciones (atención multi-cabeza).

Función softmax

La función softmax es una herramienta matemática que convierte una lista de números en una distribución de probabilidades. Es decir, transforma cualquier conjunto de valores en otro conjunto donde todos los números están entre 0 y 1 y suman 1. Esto le permite al modelo decidir a qué palabra prestar más atención dentro del texto.

En el mecanismo de atención de BERT, softmax se aplica después de comparar las preguntas (Q) con las claves (K). De este modo, el modelo determina la relevancia de cada palabra en relación con la que está evaluando y distribuye la atención de manera proporcional.

$$\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

Como softmax utiliza exponentes (e^{x_i}), si los valores de entrada son demasiado grandes, incluso pequeñas diferencias se amplifican mucho. Esto puede hacer que el modelo se vuelva "demasiado seguro" y preste atención casi exclusivamente a una sola palabra, ignorando las demás. Para evitarlo, antes de aplicar softmax, se dividen los valores por $\sqrt{d_k}$, que es el tamaño de los vectores de las claves. Esta división reduce la escala de los números y ayuda a mantenerlos en un rango razonable, asegurando que la atención se reparta de forma más equilibrada. Gracias a esto, el modelo no se "descontrola" y puede considerar varias palabras relevantes al mismo tiempo.

En conclusión, gracias a su arquitectura basada en Transformers, su entrenamiento bidireccional y su capacidad para ser ajustado mediante *fine-tuning*, es por lo que se tomó la decisión de usar esta herramienta para el procesamiento de texto en lenguaje natural, especialmente en tareas como las abordadas en este trabajo.

4.5.2. Ajuste a nuevas etiquetas (fine-tuning)

Una vez seleccionado el modelo preentrenado, fue necesario ajustarlo a las tareas concretas de este proyecto, ya que se requería un número de clases distinto al que incluía originalmente. El modelo base, SaBERT-Spanish-Sentiment-Analysis, había sido entrenado para clasificar únicamente entre dos tonos (positivo y negativo). Sin embargo, en este caso se necesitaba una clasificación en tres categorías de tono (positivo, negativo y neutro) y una segunda tarea con cuatro etiquetas temáticas distintas.

Para ello se aplicó una técnica conocida como fine-tuning, que consiste en ajustar un modelo ya entrenado para que aprenda una nueva tarea con un conjunto de datos diferente. En lugar de entrenar el modelo desde cero, se parte de una red neuronal que ya ha aprendido representaciones útiles del lenguaje (por ejemplo, cómo se relacionan las palabras entre sí) y se le entrena unas pocas veces más con ejemplos concretos del nuevo problema. Esta técnica permite ahorrar tiempo, reducir recursos computacionales y obtener buenos resultados incluso con conjuntos de datos de otros tamaños.

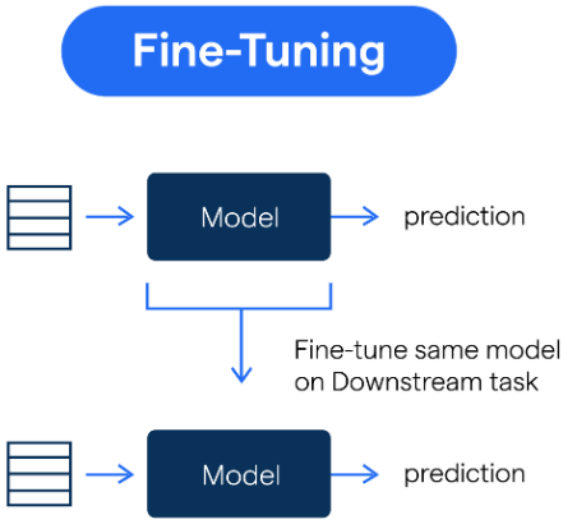

Ilustración 4.4: El proceso de *fine-tuning* permite adaptar un modelo general a tareas concretas, como la clasificación de sentimiento o la detección de temas. Imagen extraída de BotPenguin [9].

El principal cambio durante este ajuste fue la **modificación de la capa de salida**. Mientras que el modelo original estaba configurado con dos salidas (positivo/negativo), en este trabajo se sustituyó esa capa final por otra con:

- Tres salidas para la tarea de **clasificación por tono** (positivo, negativo, neutro).
- Cuatro salidas para la tarea de **clasificación temática** (Gestión pública, Economía, Sociedad y Otros).

Cada una de estas salidas produce una probabilidad y la clase predicha es aquella con mayor puntuación.

Transfer Learning: aprovechar lo ya aprendido

El *fine-tuning* es posible gracias a un concepto más general llamado aprendizaje por transferencia (*transfer learning*). Este enfoque se basa en la idea de que lo aprendido en una tarea puede ser útil para otras tareas similares. En el caso de BERT, el modelo ya había sido entrenado para entender cómo funciona el lenguaje (estructura, gramática, contexto), por lo que no necesitaba volver a aprenderlo desde cero. Lo único que se requiere es ajustar sus "conocimientos" para aplicarlos a una nueva situación.

Embeddings: representar el lenguaje como vectores

El conocimiento del modelo sobre el lenguaje se basa en lo que se conoce como embeddings. Los *embeddings* son vectores numéricos que representan palabras o frases. A través del entrenamiento, BERT aprende a asignar a cada palabra un vector que captura su significado y lo que representa. Esto quiere decir que palabras con sentidos parecidos (como "feliz" y "contento") tendrán vectores similares, mientras que palabras distintas (como "felicidad" y "tristeza") estarán mucho más alejadas entre sí en ese espacio vectorial.

Además, como BERT es bidireccional, el vector de una palabra cambia según el contexto en el que aparece. Por ejemplo, el embedding de "banco" será distinto en las frases "me senté en el banco" y "fui al banco a sacar dinero", porque el modelo entiende que en cada caso la palabra se refiere a cosas diferentes.

Estos embeddings, ya aprendidos durante el preentrenamiento, se reutilizan durante el *fine-tuning*, ajustándolos ligeramente para que se adapten mejor a la tarea concreta que se quiere resolver.

Esta capacidad de construir modelos eficaces sin necesidad de partir desde cero fue clave para poder adaptar con éxito un modelo ya funcional a las necesidades específicas de este trabajo, ampliando el número de etiquetas sin comprometer el rendimiento.

4.5.3. Clasificación por tono

Una vez se tuvo el dataset con los datos etiquetados listos para entrenar, se pasó a la redacción de los modelos. Aunque en el modelo de análisis del tono del mensaje se incluyó la etiqueta "Neutro", principalmente se buscaba maximizar el grado de acierto de aquellos con tonos positivos y negativos. Esta decisión se tomó bajo el razonamiento de que es mejor confundir un tweet neutro con uno positivo o negativo a confundir uno negativo con uno positivo por no darle prioridad a su buen clasificado. Aun así, se diseñaron cuatro variantes del modelo preentrenado para intentar obtener aquel que devolviera un buen porcentaje de acierto en el clasificado de mensajes neutros, sin que se llegue a afectar al etiquetado de negativos y positivos.

Los modelos probados fueron los siguientes:

1. **Básico:** Este primer modelo utilizó la versión preentrenada ya mencionada y se adaptó para clasificar los mensajes en tres categorías: negativo, neutro y positivo. En este caso no se aplicaron técnicas específicas para corregir el desbalance entre clases, ya que se quería comprobar lo bien que clasificaba el modelo sin modificar. Se mantuvo la función de pérdida original (entropía cruzada), que es la usada habitualmente en este tipo de tareas.

   Para encontrar la configuración más adecuada, se probaron diferentes combinaciones de épocas de entrenamiento. Se evaluó cada una en función de su precisión total y, en caso de empate, el equilibrio entre las etiquetas, con el objetivo de evitar que algunas clases tuvieran buenos resultados a costa de otras.

Después del entrenamiento, el modelo se evaluó con un conjunto de prueba y una función que permitía detectar cuándo un mensaje no es claramente positivo ni negativo, asignándole la etiqueta neutra si las predicciones no mostraban una diferencia clara entre ambas. Este sistema se ajustó para mejorar el rendimiento global.

2. **Class_balanced:** Este modelo aplicó una técnica de balanceo de clases (class balancing) para comprobar si un reparto más equitativo entre etiquetas podía mejorar el rendimiento del clasificador. Para ello, se seleccionó la misma cantidad de mensajes para cada categoría, igualando el número de ejemplos al de la clase menos representada (en este caso la neutra). Así, las tres clases tuvieron la misma presencia durante el entrenamiento y la evaluación. Aunque esta decisión implicó reducir el tamaño total del conjunto de datos —lo que supuso perder alrededor de mil muestras—, su intención fue mejorar la identificación de las clases más difíciles. Según el estudio de Laurikkala (2001) [31], equilibrar las clases puede ser útil para mejorar la capacidad del modelo al reconocer categorías poco frecuentes, especialmente cuando estas son más complejas de identificar que el resto.

   Más allá de este ajuste, el resto del proceso se mantuvo sin cambios: se exploraron diferentes combinaciones de épocas para encontrar el mejor resultado posible, evaluando tanto la precisión global como el equilibrio entre clases. Además, se mantuvo el mismo criterio para detectar mensajes neutros, permitiendo que el modelo los identificara cuando no existía una decisión clara entre los extremos positivo o negativo.

3. **Class_Weights:** En este caso, se buscó mejorar el equilibrio del modelo aplicando un ajuste automático que le diera más importancia a las categorías menos representadas. Esto se hizo asignando pesos diferentes a cada clase en la función de pérdida durante el entrenamiento, de forma que los errores en las clases minoritarias penalizaran más que los errores cometidos en las clases mayoritarias.

   Este tipo de ajuste, conocido como *class weighting* (ponderación de clases), es útil cuando algunas categorías aparecen con mucha menos frecuencia que otras, ya que ayuda a que el modelo no tienda a ignorarlas. Por ejemplo, si una clase aparece solo en un 10 % de los casos con respecto a otras etiquetas, sin este ajuste el modelo podría acertar en la mayoría de las predicciones simplemente evitando esa clase. Asignando un peso mayor a esa categoría menos frecuente, se obliga al modelo a prestarle más atención y a esforzarse más en aprender a identificarla correctamente.

   Este sistema de ponderación se calcula automáticamente en base a cuántos ejemplos hay de cada clase, dando mayor peso a las menos frecuentes. Así, el modelo aprende de forma más equilibrada, sin dejar de lado las clases más pequeñas.

   Aparte de este cambio, el procedimiento general fue el mismo: se probaron distintas combinaciones de épocas para encontrar la que ofreciera mejores resultados, siempre respetando la misma lógica de evaluación centrada tanto en la precisión global como en el equilibrio entre etiquetas.

4. **Class_Weights + Focal Loss:** Esta variante mantuvo el uso de pesos ajustados para compensar el desbalance entre clases, pero además incorporó una función de pérdida

distinta conocida como Focal Loss. Esta técnica permite centrar el aprendizaje del modelo en aquellos ejemplos que resultan más difíciles de clasificar, reduciendo la influencia de los ejemplos más fáciles.

El proceso de entrenamiento se realizó probando distintas combinaciones de épocas y un parámetro adicional llamado gamma, que controla el grado de enfoque hacia los casos más complicados. La elección del mejor modelo final se hizo siguiendo los mismos criterios que en el resto de variantes.

Función de pérdida: Cross Entropy

Para entrenar los modelos de clasificación se utilizó la función de pérdida CrossEntropyLoss, una herramienta ampliamente empleada en tareas de clasificación multiclase. Esta función permite al modelo aprender ajustando sus predicciones a las clases reales.

En términos simples, lo que hace es medir qué tan diferentes son las probabilidades predichas por el modelo (obtenidas tras aplicar la función *softmax*, que transforma los valores de salida en probabilidades) respecto a la etiqueta correcta. Cuanto mayor sea la diferencia, mayor será la penalización o "pérdida". Esta función da más peso a los errores cometidos con mucha seguridad, y menos a las predicciones dudosas que se acercan a la respuesta correcta.

Matemáticamente, se expresa como:

$$\text{CrossEntropy}(y, \hat{y}) = -\sum_{i=1}^{C} y_i \cdot \log(\hat{y}_i)$$

donde y es la representación de la clase correcta en formato *one-hot* (un vector con un 1 en la posición correspondiente a la clase real y ceros en el resto), $\hat{y}$ son las probabilidades que el modelo asigna a cada clase, y C es el número total de clases.

Esta función de pérdida funciona bien siempre que las etiquetas sean correctas, porque ayuda al modelo a mejorar sus predicciones paso a paso. Sin embargo, algunos estudios como el de Zhang y Sabuncu (2018) [54] han mostrado que puede ser sensible a errores en las etiquetas, ya que tiende a poner más atención en aquellos ejemplos en los que se equivoca con seguridad. Esto puede llevar a que el modelo aprenda mal cuando existen muchas etiquetas incorrectas.

Aun así, en este trabajo se optó por utilizar CrossEntropyLoss debido a que los datos fueron revisados manualmente y se consideraron suficientemente fiables.

Función de pérdida: Focal Loss

Para mejorar cómo el modelo maneja situaciones donde unas clases son mucho más comunes que otras, también se probó la función de pérdida FocalLoss. Esta función fue presentada originalmente por Lin et al. (2017) para la detección de objetos [33], y ha demostrado ser útil

cuando el modelo suele confiar demasiado en sus predicciones o cuando algunas categorías son mucho más frecuentes.

FocalLoss se basa en la conocida entropía cruzada, pero incluye un ajuste extra que hace que el modelo preste menos atención a los ejemplos que ya acierta con mucha seguridad. Así, le da más importancia a los ejemplos que le cuestan más trabajo. Esto puede ayudar a que el modelo sea más preciso y también a que sus predicciones sean más realistas (es decir, que las probabilidades que calcula se parezcan más a la realidad).

La fórmula general de esta función es:

$$\text{FocalLoss}(y, \hat{y}) = -\sum_{i=1}^{C} y_i \cdot (1 - \hat{y}_i)^{\gamma} \cdot \log(\hat{y}_i)$$

donde C es el número de clases, y_i indica si la clase i es la correcta (toma valor 1 para la clase real y 0 para las demás), $\hat{y}_i$ es la probabilidad que el modelo asigna a la clase i, y γ (gamma) es un número que se puede ajustar para decidir cuánto se quiere enfocar el aprendizaje en los errores más difíciles. Si se pone $\gamma = 0$, se obtiene simplemente la entropía cruzada clásica.

Ese término adicional, $(1 - \hat{y}_i)^{\gamma}$, actúa como un filtro: si el modelo ya está muy seguro de una respuesta, la pérdida será pequeña; si no lo está, la pérdida será mayor, y el modelo dedicará más esfuerzo a mejorar en esos casos.

Además, como explican Mukhoti et al. (2020) [39], esta función puede ayudar a que los modelos de redes neuronales no solo acierten más, sino que también sean más fiables a la hora de expresar lo seguros que están de sus respuestas, lo que es muy útil cuando esas predicciones se usan para tomar decisiones importantes.

4.5.4. Comparación de resultados entre modelos de clasificación por tono

Este apartado se centró en los resultados obtenidos por el clasificador del tono, es decir, el modelo encargado de identificar si un mensaje es negativo, positivo o neutro. Los modelos aplicados al clasificador por temática se analizarán más adelante, en una sección aparte.

En el Cuadro 4.5 se muestran los porcentajes de acierto obtenidos por cada variante del modelo según el tipo de entrenamiento aplicado. Se incluyen tanto los resultados promedio (media general) como los valores por clase (*negativo*, *positivo* y *neutro*).

El objetivo de esta comparación fue analizar cómo influyen distintas estrategias de ajuste —como el balanceo de clases o la asignación de pesos— en la capacidad del modelo para clasificar correctamente cada categoría, especialmente aquellas menos representadas. Las celdas sombreadas en verde destacan los mejores resultados obtenidos para el promedio y cada clase.

Cuadro 4.5: Porcentajes de acierto por clase y promedio general según el tipo de entrenamiento

Clase	**BÁSICO**	**C.BALANCED**	**C.WEIGHTS**	**FOCAL LOSS**
MEDIA	81,8	82,8	81,9	81,6
NEGATIVO	84,6	78,2	89,7	89,1
POSITIVO	87,7	88,1	91,3	90,4
NEUTRO	73,3	82,2	64,6	65,2

Como se puede observar en el Cuadro 4.5, los modelos que mejores resultados mostraron fueron aquellos en los que se incorporaron estrategias específicas para abordar el desequilibrio entre clases: el modelo con clases balanceadas y el modelo con ponderación de clases (class weights).

En el primer caso, se aplicó una técnica de submuestreo para equilibrar el número de ejemplos de cada clase, reduciendo la cantidad de muestras de las clases mayoritarias hasta igualarlas con la clase menos representada. Como resultado, este modelo fue entrenado únicamente con 1515 muestras (505 por clase), frente a las 2571 que se usaron en los demás modelos. A pesar de trabajar con un conjunto de datos más reducido, esta estrategia resultó ser muy efectiva en la detección de la clase neutro, la cual resultaba ser más difícil de clasificar que las otras. Gracias al balanceo, se logró un aumento notable en la precisión de esta clase (del 73,3 % en el modelo básico al 82,2 %), así como una mejora en el promedio general de aciertos, que alcanzó el valor más alto de todos los modelos evaluados (82,8 %).

Por otro lado, el modelo que utilizó pesos por clase durante el entrenamiento —es decir, que dio más importancia a los errores cometidos en las clases menos representadas— también obtuvo buenos resultados. Esta estrategia permitió mejorar notablemente el rendimiento en las clases positiva y negativa, alcanzando las puntuaciones más altas en esas categorías (91,3 % y 89,7 % respectivamente). Esto muestra que aplicar penalizaciones distintas a cada clase en la función de pérdida (en este caso, CrossEntropyLoss) ayuda al modelo a aprender de forma más equilibrada, incluso cuando unas clases tienen muchas más muestras que otras.

En cambio, el modelo entrenado con Focal Loss, aunque fue pensado para centrarse en los ejemplos difíciles y mejorar la calibración del modelo, no consiguió resultados tan buenos. En particular, tuvo problemas para clasificar correctamente las muestras de la clase neutro, donde obtuvo el segundo peor resultado. Esto sugiere que, en este conjunto de datos, las estrategias que ajustan directamente los datos o los pesos de cada clase son más efectivas que aquellas que intentan adaptarse dinámicamente durante el entrenamiento.

En resumen, los resultados muestran que prestar atención al desequilibrio entre clases —ya sea reduciendo la diferencia entre ellas o ajustando la forma en que se entrena el modelo— puede mejorar claramente su desempeño, sobre todo en aquellas clases que suelen tener menos ejemplos.

Cuadro 4.6: Diferencia entre la puntuación máxima por clase y los valores alcanzados por cada modelo. Se resalta la mayor diferencia por clase (amarillo) y el total menor (verde).

Clase	BÁSICO	C.BALANCED	C.WEIGHTS	FOCAL LOSS
MEDIA	1,00	0,00	0,90	1,20
NEGATIVO	5,10	11,50	0,00	0,60
POSITIVO	3,60	3,20	0,00	0,90
NEUTRO	8,90	0,00	17,60	17,00
TOTAL	18,60	14,70	18,50	19,70

Además de obtener el grado de acierto que tuvo cada modelo, también se calculó cuánto se alejaron sus resultados del mejor valor posible en cada clase. Esto se muestra en el Cuadro 4.6. En este, por cada clase (Media, Negativo, Positivo y Neutro), se restó el resultado de cada modelo al mejor valor alcanzado por cualquiera de ellos. Cuanto más bajo fuera ese número, mejor lo hizo el modelo en esa categoría.

Gracias a este enfoque, podemos ver no solo quién sacó la mejor nota, sino también quién estuvo más cerca de los mejores resultados de forma general. En este sentido, el modelo con clases balanceadas (C.BALANCED) fue el que tuvo el total de diferencia más bajo (14,70), lo que indica que fue el más constante y equilibrado entre todas las clases.

Por el contrario, aunque el modelo con pesos por clase (C.WEIGHTS) fue el mejor en las clases positiva y negativa, su puntuación en la clase "Neutro" fue la que más se alejó del mejor valor, lo que elevó su diferencia total. Algo parecido ocurrió con el modelo básico, que tuvo buenos resultados en algunas clases, pero se quedó muy atrás en otras. Sin embargo, el modelo con Focal Loss fue el que más lejos estuvo de los mejores resultados.

4.5.5. Modelo elegido

Tras evaluar los distintos modelos entrenados, se decidió elegir como modelo final aquel que utiliza ponderación de clases (class weights) junto con la función de pérdida **CrossEntropyLoss**. Esta elección vino dada porque el objetivo principal de este sistema no es solo obtener una buena puntuación media, sino también **clasificar correctamente los mensajes positivos y negativos**, ya que estos son mucho más importantes que los mensajes neutros en cuanto a los riesgos de un mal clasificado se refiere.

Como se explicó anteriormente, si un mensaje negativo se llegara a clasificar como positivo —o viceversa—, podría causar muchos problemas, sobre todo en contextos como el análisis político, donde una mala interpretación puede modificar el sentido del mensaje original. De hecho, tal y como señalan Matalon et al. [36], en debates políticos *es esencial sopesar explícitamente tanto los efectos positivos de las interacciones que están de acuerdo con la Fuente como los efectos negativos de las interacciones que están en desacuerdo con ella.*

Por tanto, era fundamental que el modelo supiera *clasificar correctamente el sentimiento expresado* por políticos y ciudadanos, especialmente cuando este era negativo o positivo.

El modelo con **ponderación de clases** destacó por mejorar el rendimiento en la clasificación del tono positivo y negativo, lo que garantizaba una mayor fiabilidad en las predicciones que realmente importaban. Este comportamiento resultó especialmente relevante cuando se trataba de mensajes que podrían modificar el sentido de una conversación o causar malentendidos, como ocurre en contextos políticos o sociales.

Aunque es cierto que el modelo con **clases balanceadas** obtuvo una puntuación media superior (casi un 1 % más alta), este rendimiento lo logró **sin haber utilizado aproximadamente mil ejemplos adicionales de mensajes positivos y negativos**, que sí se incluyeron en el entrenamiento del modelo con ponderación de clases. Esta diferencia sugiere que el modelo balanceado no llegó a enfrentarse a suficientes ejemplos como para toparse con mensajes difíciles de clasificar.

Por otro lado, si se compara con el modelo básico, es evidente que la mejora obtenida se debió precisamente al refuerzo del aprendizaje sobre esas clases críticas. Aunque el rendimiento del modelo final en la clase neutra no fue el mejor, **el equilibrio logrado en la clasificación de los mensajes positivos y negativos lo convirtió en la opción más adecuada** para cumplir con los objetivos de este trabajo.

4.5.6. Clasificación por tema

Además de clasificar los mensajes según su tono (positivo, negativo o neutro), también se entrenó un modelo para identificar el **tema principal** de cada mensaje. Esta tarea buscaba asignar a cada texto una categoría temática específica (por ejemplo, economía, sanidad, empleo, etc.), lo que permitía entender mejor de qué se hablaba en cada caso.

Dado que, en la tarea de clasificación por tono, el modelo con class weights y función de pérdida CrossEntropyLoss fue el que mejor rendimiento ofreció en las categorías clave, se decidió aplicar directamente esa misma estrategia al clasificador por tema. A diferencia de lo que se hizo con los modelos de tono, aquí no se repitió todo el proceso de comparación entre variantes (como el balanceo de clases o el uso de funciones de pérdida alternativas). En su lugar, se adaptó el modelo preentrenado original para que trabajara con etiquetas temáticas, utilizando la técnica de **ponderación de clases** en la función de pérdida.

Esta decisión funcionó bien: el modelo alcanzó una **precisión del 81,5 %** y ofreció un rendimiento equilibrado entre las diferentes categorías. Las clases con mayor presencia en el conjunto de datos, como *Gestión Pública* y *Economía*, lograron porcentajes de acierto en la clasificación superiores al 84 % (84,1 % y 84,8 % respectivamente). Incluso las clases más pequeñas, como *Sociedad, Igualdad y Derechos* (78,4 %) y *Otros* (68,4 %), mantuvieron resultados positivos. Esto indica que el modelo fue capaz de identificar correctamente la mayoría de los mensajes de cada categoría, sin necesidad de aplicar ajustes adicionales para compensar desequilibrios en los datos.

4.6. Aplicación del modelo y análisis exploratorio

Después de entrenar y validar los modelos de clasificación por tono y por tema, se procedió a su aplicación sobre el conjunto completo de mensajes recogidos. Para los **posts**, se aplicaron ambos modelos de tono y de tema sobre la columna *Contenido_Traducido*, ya que esta representa el mensaje traducido al castellano en caso de haber sido originalmente escrito en otro idioma.

En el caso de los **comentarios**, se utilizó la columna *Comentario_Traducido* y, para las posibles **respuestas de políticos** a esos comentarios, la columna *Respuesta_Traducida*. A diferencia de los posts, en estos dos casos únicamente se aplicó el modelo de tono. La razón es que tanto los comentarios ciudadanos como las respuestas de los representantes no siguen necesariamente una estructura temática definida, por lo que no resultaba adecuado aplicar un clasificador de temas a este tipo de contenido, habitualmente más espontáneo y poco estructurado.

En total, se clasificaron:

- **8.597 posts por temática**
- **37.970 mensajes por tono** (incluyendo posts, comentarios y respuestas)

4.6.1. Distribución de temas (en los posts)

Tras aplicar el modelo de clasificación temática, se obtuvo la siguiente distribución de temas en las publicaciones realizadas por los políticos:

- **Gestión Pública e Instituciones:** 4.415 mensajes
- **Economía, Empresa, Empleo e Infraestructuras:** 2.453 mensajes
- **Sociedad, Igualdad y Derechos:** 1.790 mensajes
- **Otros:** 861 mensajes

Estos resultados muestran el mismo orden en el número de elementos etiquetados que cuando fueron asignados manualmente en el Excel de entrenamiento, con los contenidos centrados en la gestión institucional y los asuntos económicos siendo los primeros, seguidos de cuestiones sociales y los temas indefinidos en menor medida.

4.6.2. Distribución del tono en los mensajes

En cuanto al tono, los resultados del modelo aplicado fueron los siguientes:

En los posts:

- Positivos: 5.643

- Negativos: 3.025
- Neutros: 851

En los comentarios:

- Negativos: 21.649
- Neutros: 5.371
- Positivos: 2.946

En las respuestas a comentarios:

- Negativas: 81
- Neutras: 60
- Positivas: 59

Estos datos reflejan una diferencia clara entre el tono empleado por los políticos en sus publicaciones —predominantemente positivo— y el tono más crítico o negativo de los comentarios de los ciudadanos. También destaca la variedad de tono en las respuestas de los políticos, a pesar de que el tono negativo sigue siendo el predominante.

Una vez obtenidas las predicciones del modelo, se procedió a realizar un análisis exploratorio con el objetivo de identificar patrones relevantes en los datos finales. Para ello, se exploraron aspectos como la distribución de temas, la evolución del tono en las publicaciones y respuestas, así como posibles diferencias en la interacción con la ciudadanía según características sociodemográficas de la audiencia, tales como edad, género, comunidad autónoma o nivel educativo.

Este trabajo exploratorio se apoyó en herramientas de análisis y visualización de datos como pandas, plotly, geopandas, y técnicas de procesamiento de lenguaje natural implementadas con librerías como nltk o spaCy. A través de ellas se realizaron limpiezas adicionales de texto, extracción de palabras clave, análisis de frecuencias, y representaciones gráficas interactivas para aportar una visión más profunda y comprensible del comportamiento discursivo de los actores políticos en redes sociales.

Los resultados específicos de esta fase de análisis se presentan en detalle en el capítulo de Resultados.

4.7. Despliegue

Despliegue de la aplicación

Todas las visualizaciones y análisis explicados en el capítulo Resultados se reunieron en una herramienta interactiva accesible desde una página web. Esta herramienta, creada con Streamlit, organizó los distintos gráficos generados con Plotly (barras, sectores, mapas,

comparativas de términos y entidades, análisis de tono, etc.) según los resultados obtenidos, para que cualquier persona pudiera explorarlos de forma sencilla y dinámica.

Para su desarrollo, se estructuró el código en varios archivos, según la función de cada parte (carga de datos, visualizaciones, análisis más avanzados, configuración de colores y filtros). Esto permitió tener el proyecto mejor organizado y facilitar su mantenimiento a futuro.

Durante las pruebas, se utilizó el servidor local de Streamlit, ejecutando la aplicación con el comando **streamlit run app.py**, lo que permitió revisar y ajustar el resultado en tiempo real desde el navegador. Más adelante, se preparó el proyecto para poder desplegarlo en la nube, valorando opciones como **Streamlit Cloud** o **Heroku**, para hacerlo más fácil de compartir con otras personas.

También se creó un archivo de dependencias (`requirements.txt`) con todas las librerías necesarias, de forma que cualquiera que tenga unas nociones básicas de Python pueda instalar y usar la aplicación, tanto en su ordenador como en un servidor remoto.

El objetivo de la herramienta fue que cualquier usuario, sin importar su nivel técnico, pudiera navegar por los resultados y analizarlos de forma intuitiva. Gracias a los filtros disponibles (por partido, comunidad autónoma, periodo de tiempo, tipo de análisis), la aplicación facilita descubrir nuevos patrones de comunicación política en redes sociales, sin necesidad de programar o usar herramientas complejas.

Finalmente, la aplicación se desplegó con Streamlit, lo que hace posible ejecutarla en local o en la nube y compartirla, aunque con algunas limitaciones en cuanto a recursos disponibles y escalabilidad.

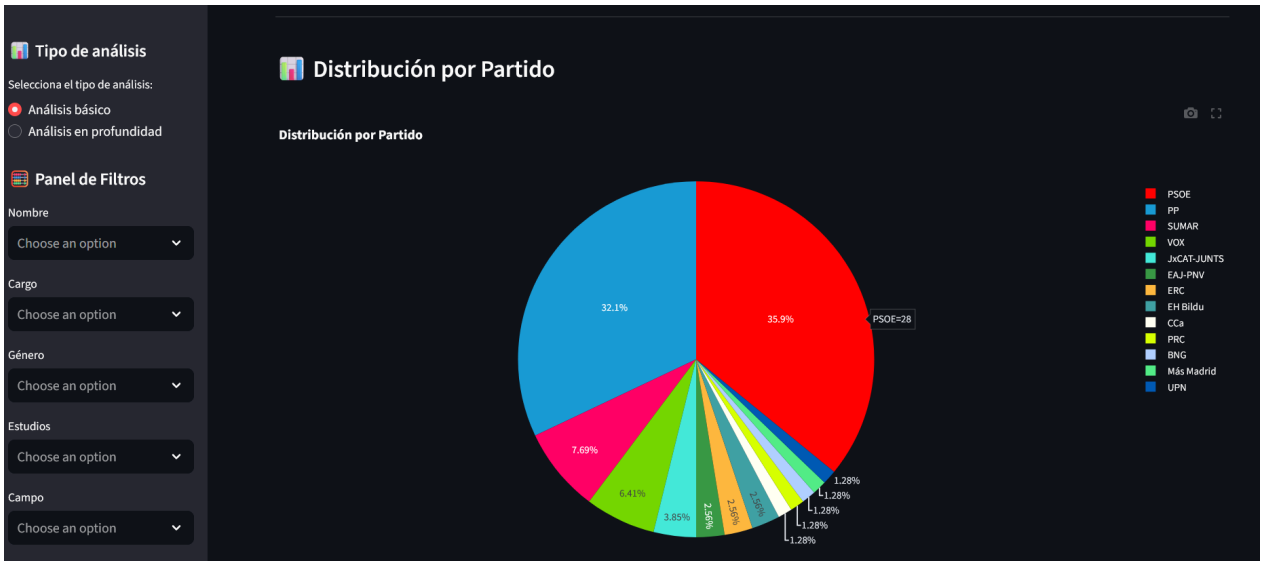

Ilustración 4.5: Vista previa de la aplicación creada.

Capítulo 5

Resultados

Esta sección tiene como objetivo responder, mediante visualizaciones y análisis estadísticos, a una serie de preguntas clave que podrían plantearse los partidos políticos en relación con su estrategia comunicativa en redes sociales. A partir de los datos recogidos y procesados en las fases anteriores del proyecto, se exploraron patrones de comportamiento, temáticas dominantes, variaciones en el tono del discurso y diferencias en las respuestas que recibieron en función de variables como la edad, la comunidad autónoma, el género o el nivel educativo.

5.1. Análisis descriptivo de *Metadata*

Antes de comenzar con el análisis en profundidad y la búsqueda de patrones, fue importante entender los datos básicos de los políticos con los que se pretendía llevar a cabo el estudio. Debido a esto, se realizó una visualización descriptiva de la tabla *Metadata* con el objetivo de mostrar las diferentes variables de cada columna y el peso de cada elemento de esta en el estudio.

Cuadro 5.1: Perfil de los políticos analizados (n=78). Parte 1: Variables personales, académicas y territoriales.

Variable	**Categoría**	**Frecuencia**	**%**
Cargo	Presidente	17	21,80
	Diputado	46	59,00
	Líder_Oposición	15	19,20
Género	Mujer	28	35,90
	Hombre	50	64,10
Estudios	Secundaria	1	1,28
	Bachillerato	8	10,30
	Grado	42	53,80
	Postgrado	27	34,60
Campo	Derecho_CienciasPolíticas	30	38,50
	Economía_Administración_Empresa	15	19,20
	Ingeniería_Tecnología	9	11,50
	CienciasSociales_Humanidades	7	8,97
	Medicina_Salud	5	6,41
	Educación	3	3,85
	Traducción	3	3,85
	Comunicación	3	3,85
	Psicología	2	2,56
	Artes_Diseño	1	1,28
Comunidad Autónoma	Comunidad de Madrid	16	20,50
	Andalucía	7	8,97
	Cataluña	7	8,97
	Comunidad Valenciana	6	7,69
	País Vasco	6	7,69
	Castilla y León	5	6,41
	Castilla-La Mancha	4	5,13
	La Rioja	4	5,13
	Aragón	3	3,85
	Principado de Asturias	3	3,85
	Canarias	3	3,85
	Galicia	3	3,85
	Comunidad Foral de Navarra	3	3,85
	Islas Baleares	2	2,56
	Cantabria	2	2,56
	Extremadura	2	2,56
	Región de Murcia	2	2,56

Cuadro 5.2: Perfil de los políticos analizados (n=78). Parte 2: Variables políticas y legislativas.

Variable	Categoría	Frecuencia	%
Partido	PSOE	28	35,90
	PP	25	32,10
	SUMAR	6	7,69
	VOX	5	6,41
	JxCAT-JUNTS	3	3,85
	EAJ-PNV	2	2,56
	ERC	2	2,56
	EH Bildu	2	2,56
	CCa	1	1,28
	PRC	1	1,28
	BNG	1	1,28
	Más Madrid	1	1,28
	UPN	1	1,28
Número de Legislaturas	1	34	43,60
	2	11	14,10
	3	19	24,40
	Más de 3	14	17,90
Edad	30–39	9	11,50
	40–49	24	30,80
	50–59	38	48,70
	60–69	5	6,41
	70+	2	2,56

5.2. Análisis general de la comunicación política

Tras haber descrito las características principales de los políticos incluidos en el estudio, el siguiente paso fue analizar cómo se comunicaban en la red social X/Twitter y qué tipo de impacto generaron con sus publicaciones.

Dado que no era posible abordar todas las variables disponibles por cuestiones de espacio, se optó por centrar el análisis en cinco grandes bloques que ayudaron a entender distintos aspectos de la comunicación digital:

- **Popularidad y alcance**: quiénes tuvieron más visibilidad y seguidores.
- **Actividad en X/Twitter**: con qué frecuencia publicaron y desde cuándo usan la red.

- **Interacción e impacto**: qué perfiles generaron más respuestas, likes o retweets.
- **Tonalidad del discurso**: si los mensajes fueron positivos, negativos o neutros.
- **Contenido**: sobre qué temas hablaron y qué palabras o nombres mencionaron más.

Estos cinco ejes permitieron identificar patrones en el comportamiento digital de los políticos y partidos, así como diferencias entre regiones o posicionamientos ideológicos.

Una vez explicados estos cinco bloques, se realizó un análisis final para detectar posibles relaciones entre visibilidad, frecuencia de publicación, tono o temáticas tratadas. Esta parte sirvió para extraer conclusiones más amplias y detectar estrategias comunes o diferencias significativas entre políticos y sus partidos.

5.2.1. Popularidad y alcance

Este primer bloque del análisis se centró en entender qué políticos y partidos tenían más visibilidad en la red social X/Twitter. Para ello, se analizaron el número total de seguidores, la velocidad con la que crecían sus cuentas y las diferencias entre comunidades autónomas.

En lo que respecta a los políticos más seguidos, Pedro Sánchez fue, con diferencia, el líder con mayor número de seguidores (casi dos millones), seguido por Isabel Díaz Ayuso y Gabriel Rufián, ambos con un millón. También destacaron perfiles como Santiago Abascal, Yolanda Díaz o Miguel Ángel Revilla. En concreto, el caso del secretario general del Partido Regionalista de Cantabria (PRC) es especialmente curioso por el hecho de que su número de seguidores (638 mil) actualmente supera a la población de la propia Cantabria (593 mil).

Otro aspecto a resaltar es el PP, único partido de ámbito estatal en el que el líder de dicha formación (Alberto Núñez Feijóo) no es el más popular en la red social. En su lugar, es la presidenta de la Comunidad de Madrid, Isabel Díaz Ayuso la que ocupa esa posición con más del cuádruple de seguidores.

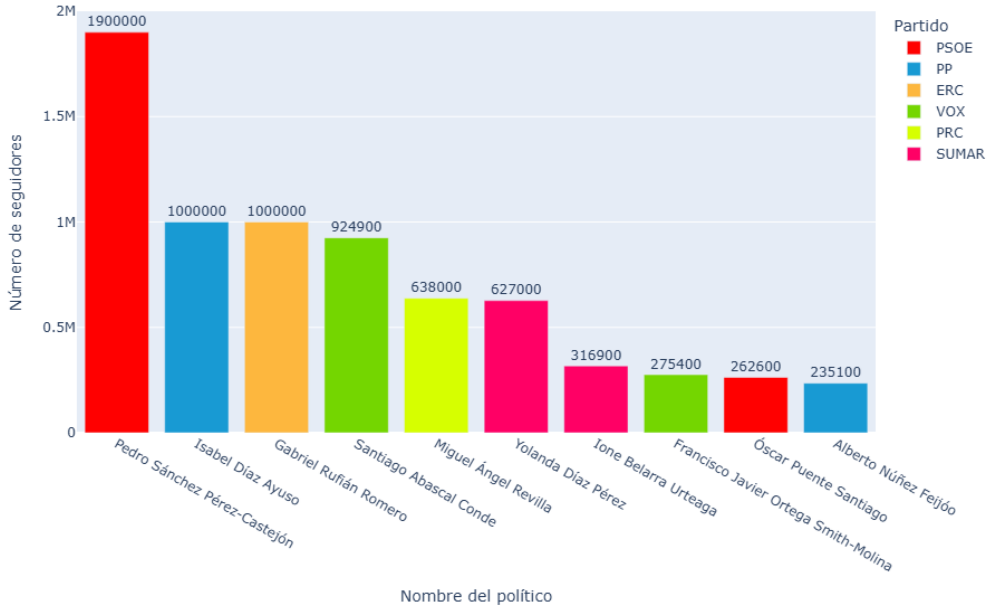

Ilustración 5.1: Top 10 políticos más populares en X/Twitter según número de seguidores.

Al agrupar los datos por partido, el PSOE y el PP encabezaron la lista con más de tres y casi dos millones de seguidores, respectivamente. Aun siendo partidos más pequeños en representación, formaciones como VOX, ERC o SUMAR también lograron reunir una cantidad de seguidores significativa. Destaca sobre todo el peso que tienen las figuras más importantes sobre sus respectivos partidos; Pedro Sánchez (PSOE), Santiago Abascal (VOX) y Yolanda Díaz (SUMAR) aportan alrededor de dos tercios de los seguidores totales de sus partidos, mientras que Ayuso supone más de la mitad de los usuarios para el PP y Gabriel Rufián casi la totalidad de Esquerra Republicana de Catalunya (ERC).

Con respecto a las diferencias entre los dos partidos principales (PSOE – PP) y el resto de formaciones, vale la pena mencionar cómo aquellos con menor representación en este estudio, como pueden ser SUMAR (6 políticos), VOX (5 políticos) o ERC (2 políticos), no presentan tanta distancia con el PP (25 políticos) como sí tiene esta formación con respecto al PSOE (28 políticos).

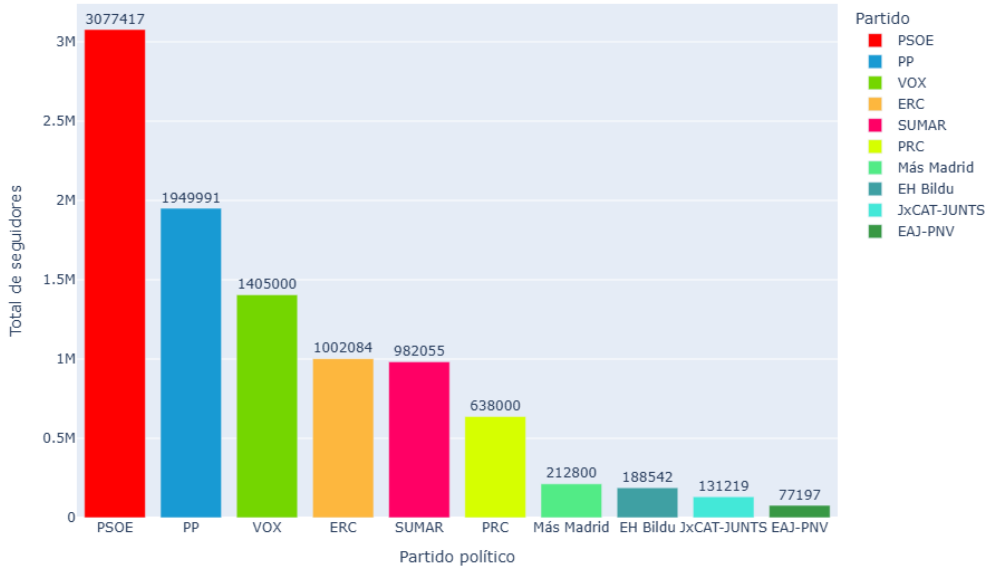

Ilustración 5.2: Top 10 partidos más populares por número total de seguidores.

Además del volumen total, también se calculó el ritmo de crecimiento de las cuentas, es decir, cuántos seguidores ganaban al año. Pedro Sánchez volvió a liderar esta métrica, seguido esta vez por Gabriel Rufián como segundo político con mejor crecimiento e Isabel Díaz Ayuso completando el top 3, todos con más de 60.000 seguidores anuales.

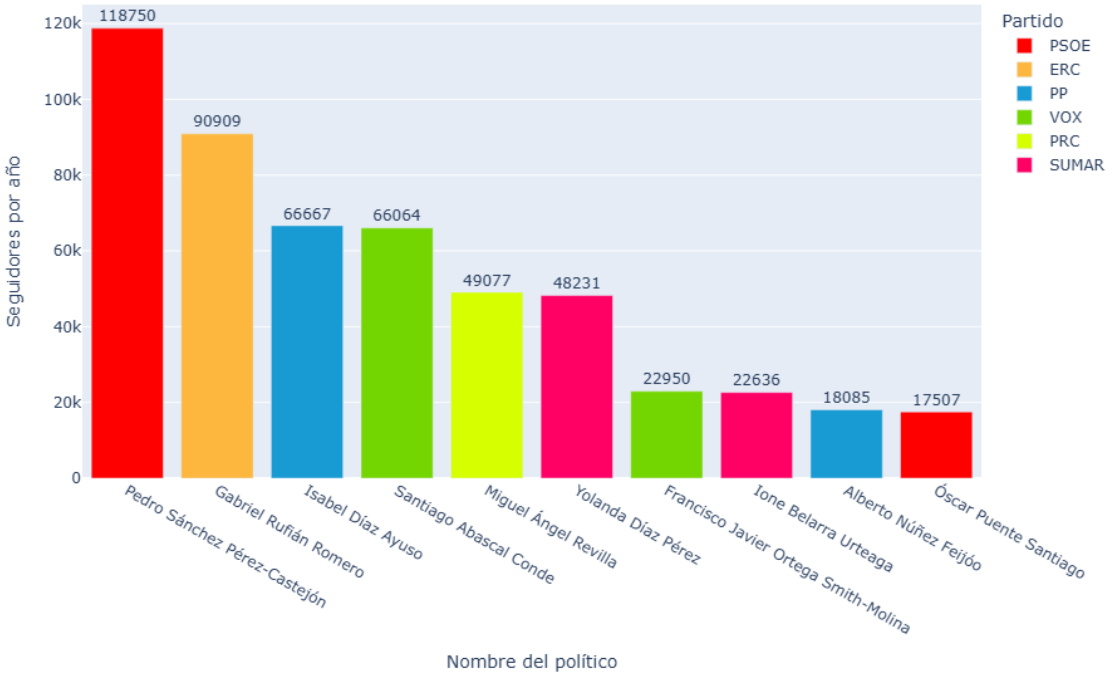

Ilustración 5.3: Top 10 políticos con mayor tasa anual de crecimiento en seguidores.

A nivel de partidos, los que más crecieron en promedio fueron ERC y el PRC, con más de 45.000 seguidores al año por perfil, con VOX, Más Madrid y SUMAR completando la mitad más positiva de esta gráfica. Esto puede deberse en parte a que dichos partidos no tienen tantos representantes que contrarresten la popularidad de sus miembros principales.

En cambio, partidos grandes como PSOE o PP, al tener más cuentas, presentaron tasas medias más bajas. Por su parte, los representantes de los partidos más tradicionales en la política regional, como son el PNV y JUNTS (antigua Convergència Democràtica de Catalunya) tampoco muestran mucha popularidad en las redes.

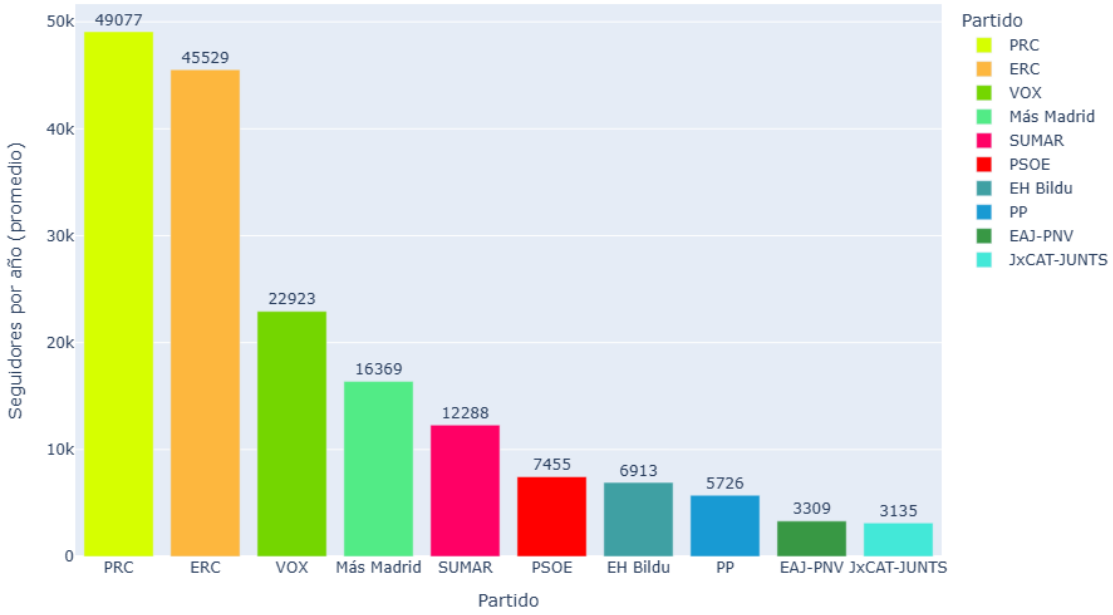

Ilustración 5.4: Tasa media de crecimiento anual por partido político.

Por último, se elaboró un mapa para visualizar las diferencias territoriales en popularidad. Las comunidades donde los políticos tenían, de media, más seguidores fueron Madrid, Cantabria y Cataluña. Esto coincidió con la presencia de figuras con mucha repercusión o relevancia tanto a nivel nacional como autonómico.

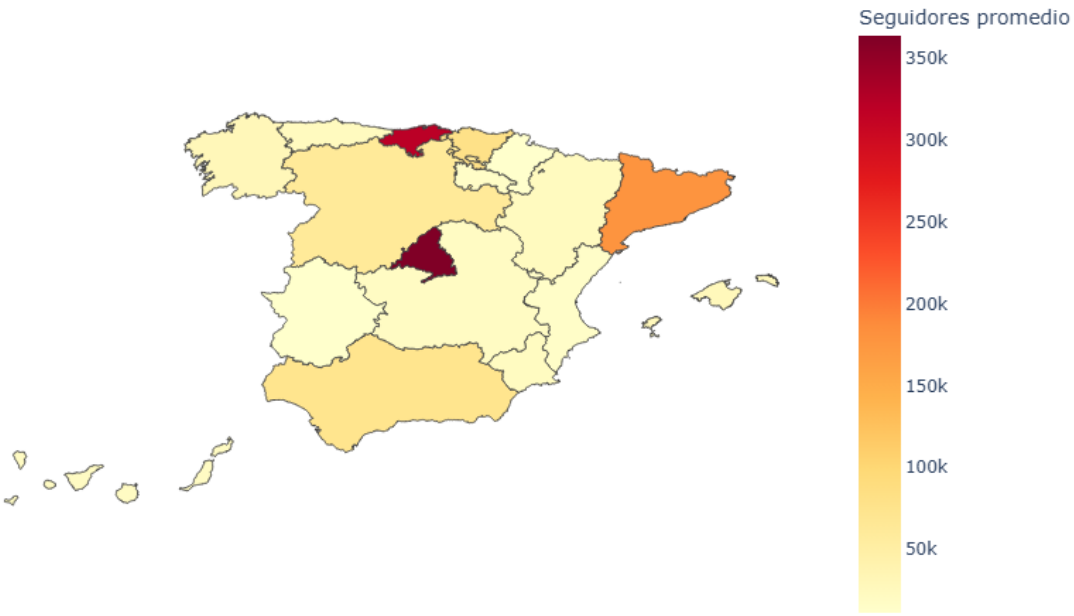

Ilustración 5.5: Popularidad media por comunidad autónoma.

Conclusión del bloque: Aunque los grandes líderes nacionales siguen siendo los más seguidos, casos como los de Rufián o Revilla demuestran que un perfil regional también puede tener un gran impacto. Además, se observó que el crecimiento en redes no siempre depende del tamaño del partido, y que algunas regiones destacan claramente por su visibilidad digital.

5.2.2. Actividad en X/Twitter

Este segundo bloque del análisis se centró en conocer con qué frecuencia publican los políticos en X/Twitter, tanto en términos totales como en la media anual de publicaciones. También se analizó la distribución territorial de la actividad para ver si algunas regiones son más activas que otras.

En cuanto al número total de publicaciones, los más activos fueron Oskar Matute (EH Bildu), Míriam Nogueras (JxCAT-JUNTS) y Cristina Narbona (PSOE), todos con cifras por encima de los 80.000 tweets.

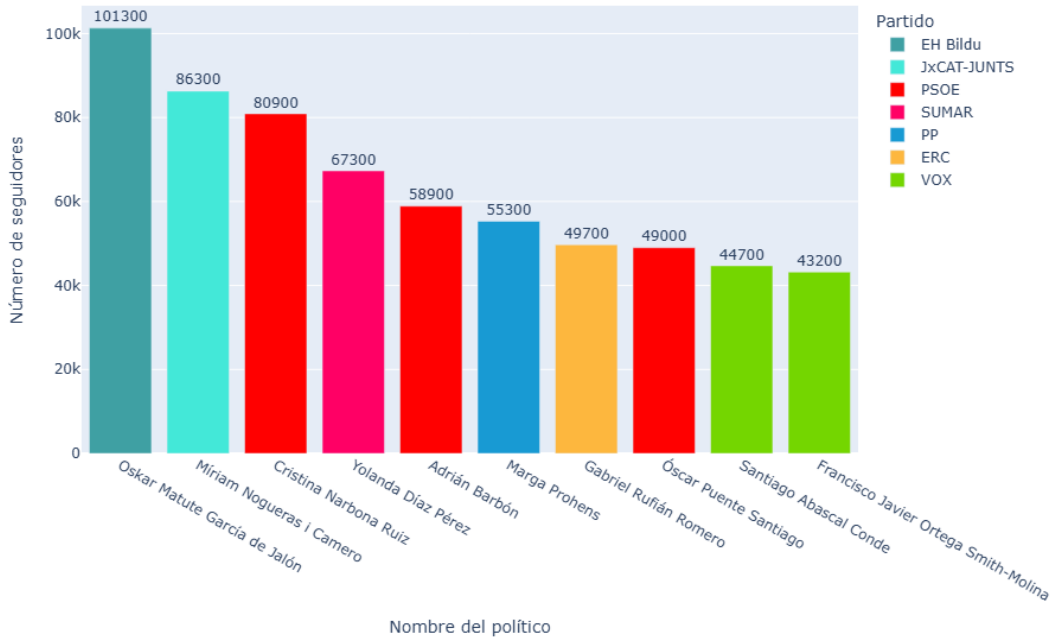

Ilustración 5.6: Top 10 políticos con mayor número de publicaciones en X/Twitter.

Cuando se midió la actividad anual, Oskar Matute volvió a encabezar el ranking con más de 6.000 publicaciones por año, seguido de Nogueras, Narbona, Yolanda Díaz y Rufián, todos ellos con estrategias muy activas. En comparación con lo obtenido en la Ilustración 5.1, la mitad de los políticos con mayor número de seguidores no aparecen entre los 10 políticos con más posts. Destacan sobre todo las ausencias de Pedro Sánchez e Isabel Díaz Ayuso.

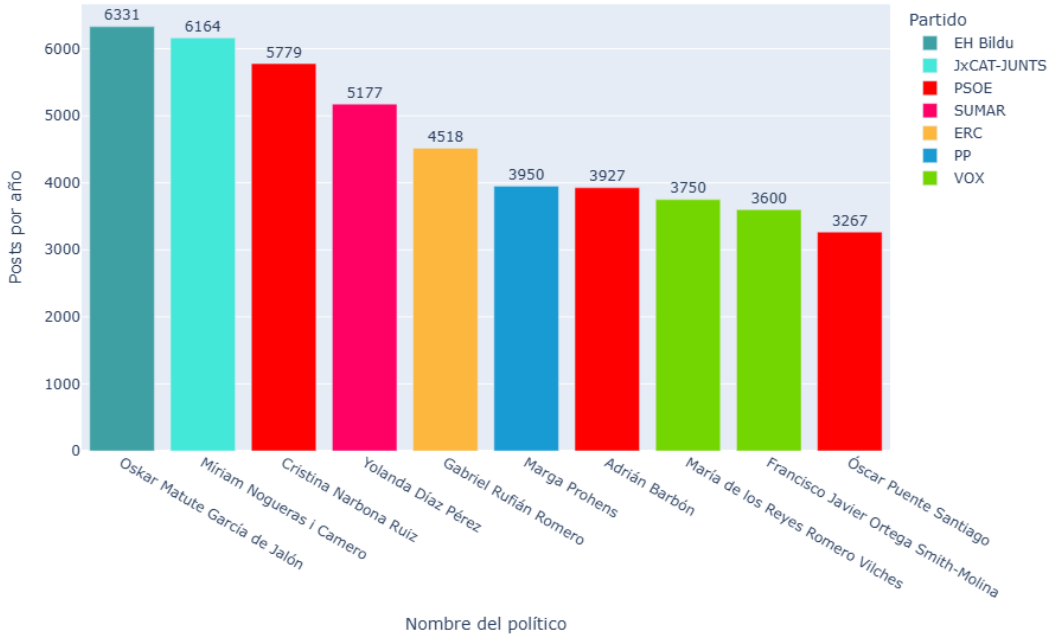

Ilustración 5.7: Top 10 políticos con mayor tasa anual de publicaciones.

A nivel de partidos, los más activos en volumen fueron el PSOE y el PP, seguidos por VOX y SUMAR. Sin embargo, partidos como EH Bildu y JxCAT-JUNTS, los cuales mostraron una masa de seguidores muy inferior con respecto a otros partidos regionalistas/independentistas, también destacaron por su alta actividad.

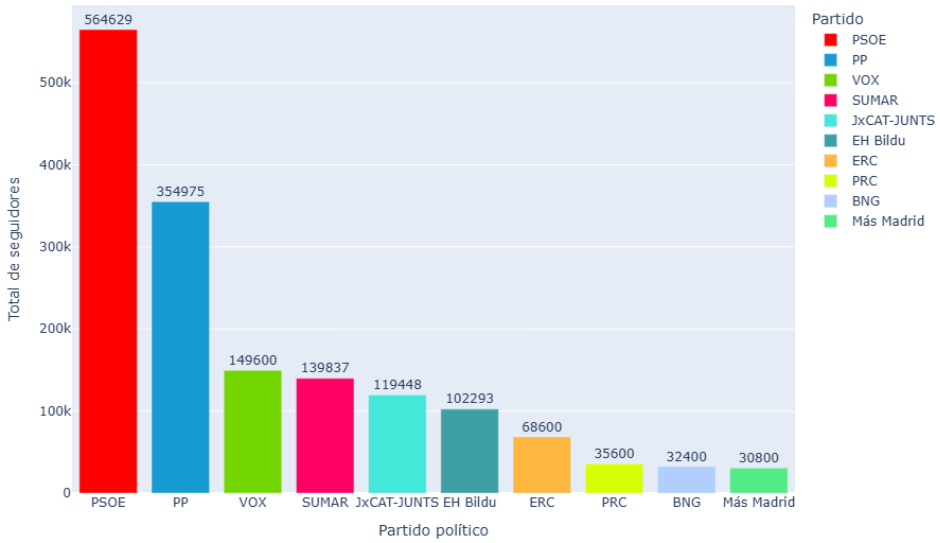

Ilustración 5.8: Top 10 partidos con mayor volumen de publicaciones.

En cuanto a la tasa media de publicaciones por año, los más activos fueron los partidos independentistas vascos (EH Bildu) y catalanes (ERC y JxCAT-JUNTS). Por el contrario, el PSOE, a pesar de su tamaño, mostró una media baja de publicaciones por cuenta, mientras que el PP se quedó fuera de los diez primeros.

Ilustración 5.9: Tasa media anual de publicación por partido.

Por último, se representó la actividad media por comunidad autónoma. Las regiones con mayor número medio de publicaciones fueron Baleares, Asturias, Cataluña y Madrid. Los casos de Baleares y Asturias son especialmente llamativos, ya que superan a regiones que tradicionalmente ocupan la narrativa nacional, como son Madrid y Cataluña.

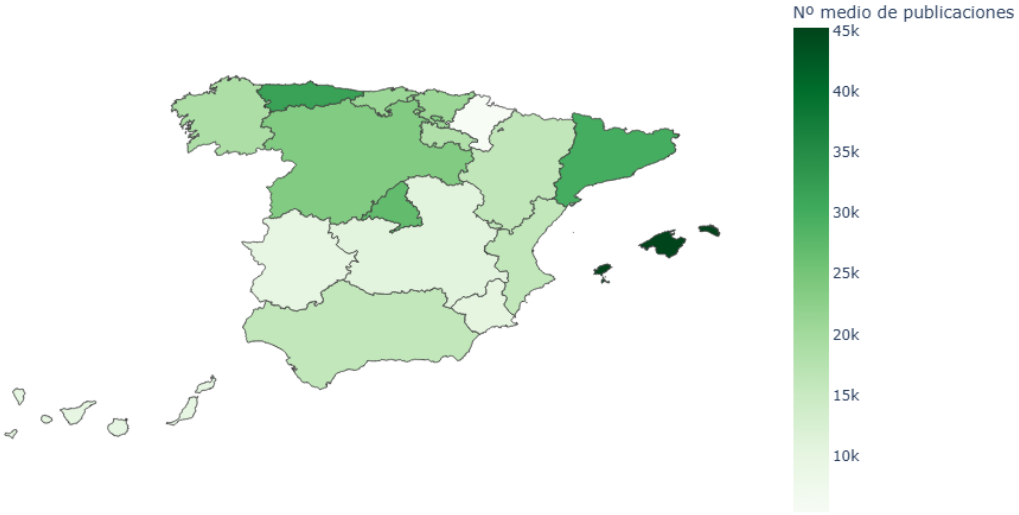

Ilustración 5.10: Frecuencia media de publicación por comunidad autónoma.

Conclusión del bloque: La frecuencia de publicación revela un uso estratégico e intensivo de la red por parte de ciertos partidos y figuras regionales. Aunque partidos como PSOE o PP lideran en volumen absoluto, otros como EH Bildu, JxCAT-JUNTS o ERC muestran una intensidad mucho mayor en la media.

5.2.3. Interacción e impacto

En este bloque se analizó si los políticos y partidos no solo están presentes en X/Twitter, sino si realmente logran generar conversación y movilizar a sus seguidores.

Para ello, se midió tanto la interacción promedio por publicación (es decir, cuántas respuestas, likes y retweets recibe cada tuit) como la interacción relativa (proporción de interacciones por número de seguidores). Esto permitió detectar no solo qué cuentas tienen más repercusión total, sino cuáles logran mayor implicación en relación con su comunidad.

Gabriel Rufián (ERC) fue el político que más interacción generó por publicación, con más de 43.000 interacciones de media, siendo esta cifra cuatro veces más que el promedio del segundo puesto, Míriam Nogueras (JxCAT-JUNTS). A esta le siguieron políticos con elevado número de seguidores como Santiago Abascal (VOX), Pedro Sánchez (PSOE) o Isabel Díaz Ayuso (PP).

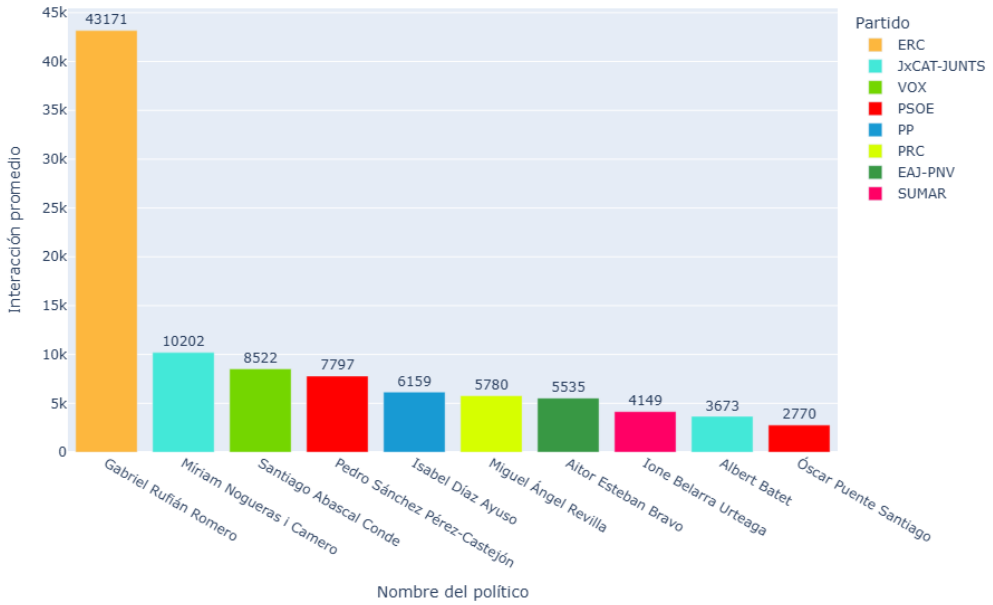

Ilustración 5.11: Top 10 políticos con mayor interacción promedio por publicación.

A nivel de partidos, ERC volvió a destacar muy por encima del resto con más de 21.000 interacciones de media por post. Le siguieron el PRC, JxCAT-JUNTS y VOX. Los partidos estatales más grandes, como el PSOE, se situaron en niveles mucho más bajos, mientras que a nivel de partido el PP volvió a desaparecer del top 10.

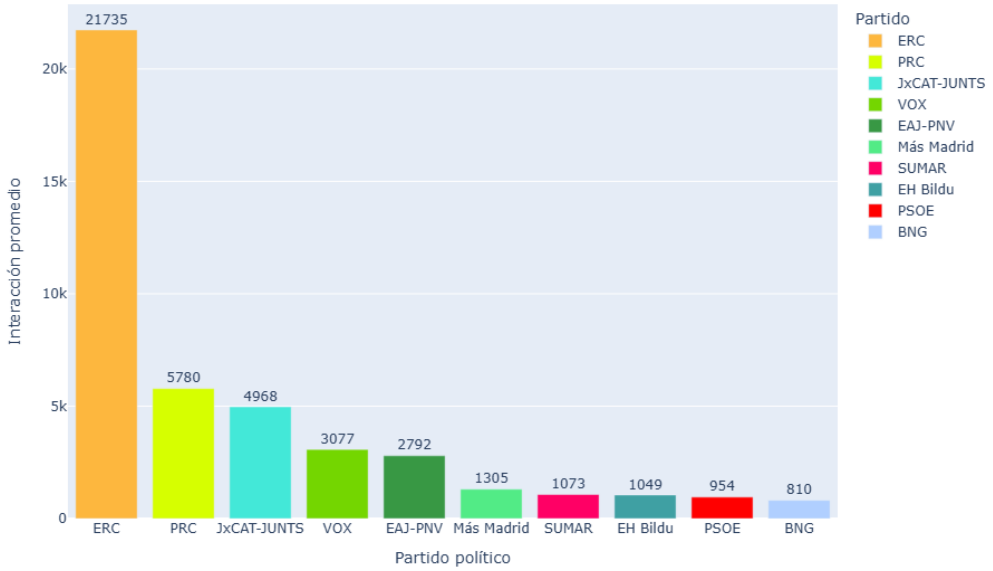

Ilustración 5.12: Top 10 partidos con mayor interacción promedio por publicación

En el siguiente indicador, que muestra el nivel de compromiso por parte de los seguidores,

volvió a destacar Albert Batet (JxCAT-JUNTS) con un ratio de 0,24 interacciones por cada seguidor. Le siguieron Josep Maria Cruset, Inés Granollers (ERC) y varios perfiles del PSOE con ratios notables.

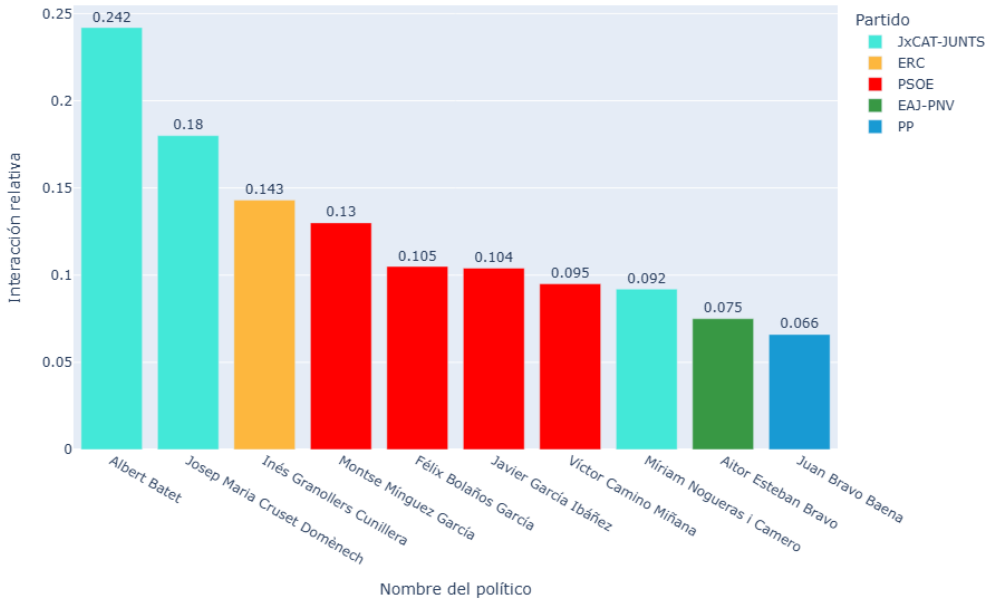

Ilustración 5.13: Top 10 políticos con mayor interacción relativa (por seguidor).

El patrón se repitió a nivel de partido: JxCAT-JUNTS y ERC fueron los que generaron mayor fidelidad e implicación en sus comunidades, seguidos por EAJ-PNV. Por el contrario, VOX, SUMAR, PSOE o PP, pese a tener más seguidores, mostraron ratios de interacción más bajos.

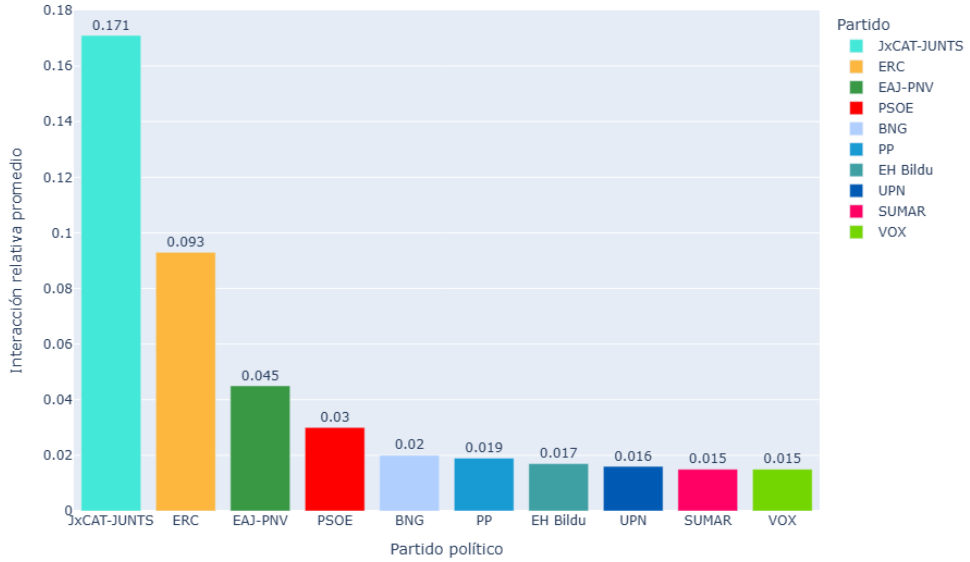

Ilustración 5.14: Top 10 partidos con mayor interacción relativa promedio.

En términos territoriales, Cataluña fue la comunidad con mayor interacción relativa media. Esta diferencia respecto a otras regiones refuerza la idea de un uso más movilizador del discurso político en ese territorio.

Ilustración 5.15: Interacción relativa media por comunidad autónoma.

Conclusión del bloque: La interacción en X/Twitter no depende solo del número de seguidores, sino de la capacidad de los perfiles para movilizar a su audiencia. En este sentido, los partidos nacionalistas catalanes como ERC y JxCAT-JUNTS se posicionaron claramente como los más eficaces en generar conversación y compromiso digital. Esto refuerza la idea de que los partidos y perfiles regionales pueden tener un impacto superior en términos relativos con respecto a elementos de ámbito estatal.

5.2.4. Tonalidad del discurso

Este bloque se centró en analizar el tono de las publicaciones políticas en X/Twitter, es decir, si los mensajes compartidos eran positivos, negativos o neutros. Para ello, se evaluó la proporción de tono por partido, por político individual, por comunidad autónoma y también por temas tratados.

En primer lugar, al observar el tono positivo, destacaron perfiles institucionales como Gonzalo Capellán (PP) con el 100 % de sus publicaciones en tono positivo. También sobresalieron figuras como Yolanda Díaz (SUMAR), única líder de un partido de ámbito estatal que entra en el top 10 de tono positivo. Lo más resaltable es el hecho de que 8 de los 10 políticos de este top son presidentes de Comunidad Autónoma.

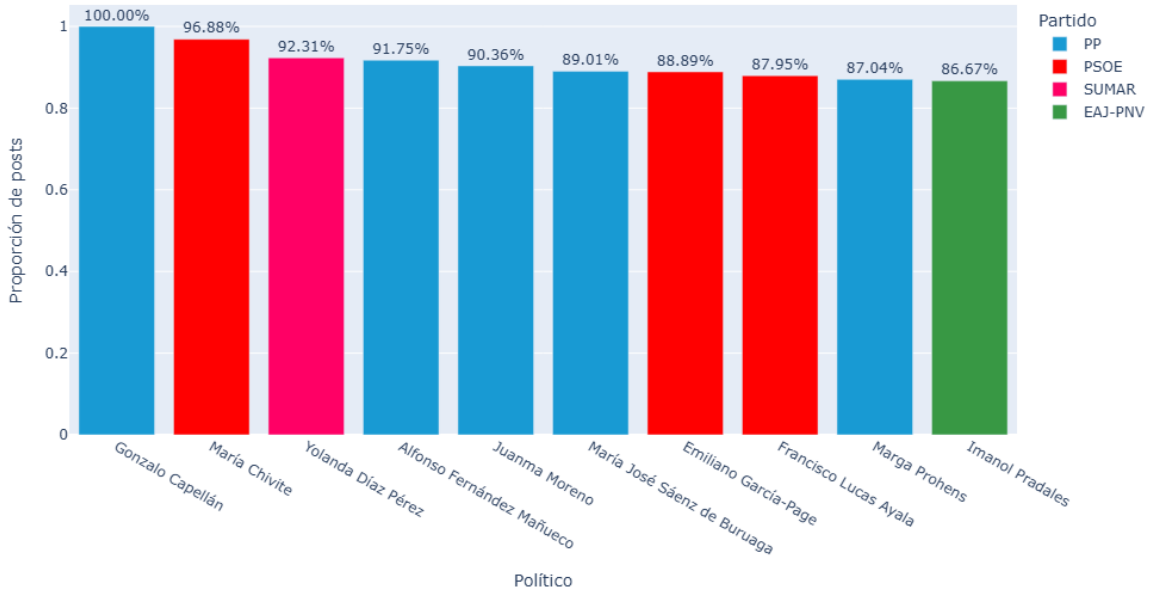

Ilustración 5.16: Top 10 políticos con mayor proporción de publicaciones positivas.

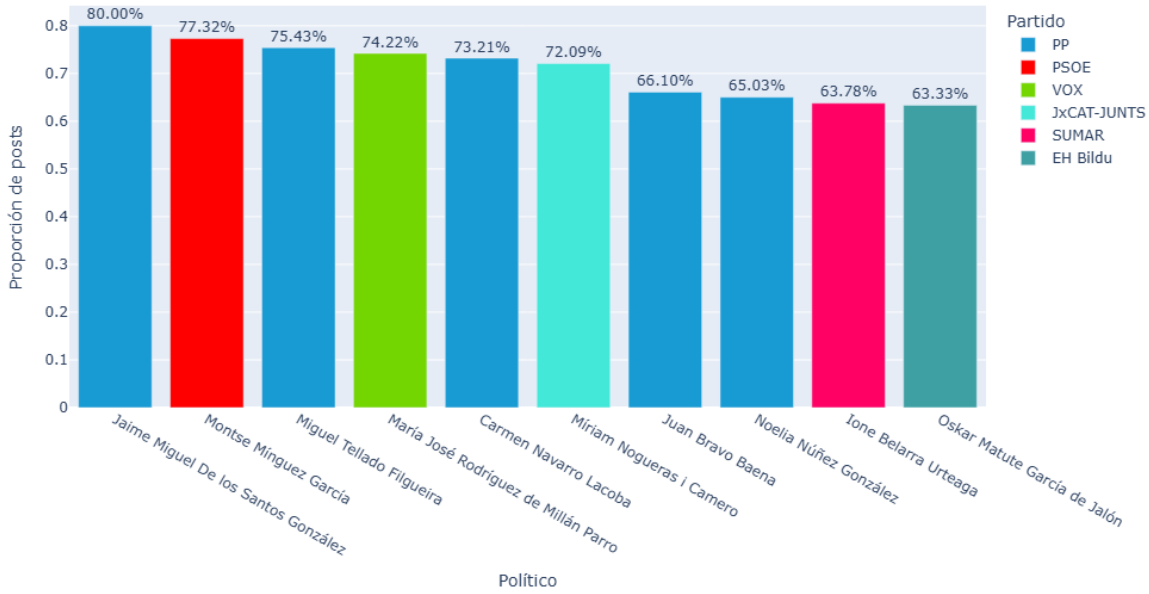

Ilustración 5.17: Top 10 políticos con mayor proporción de publicaciones negativas.

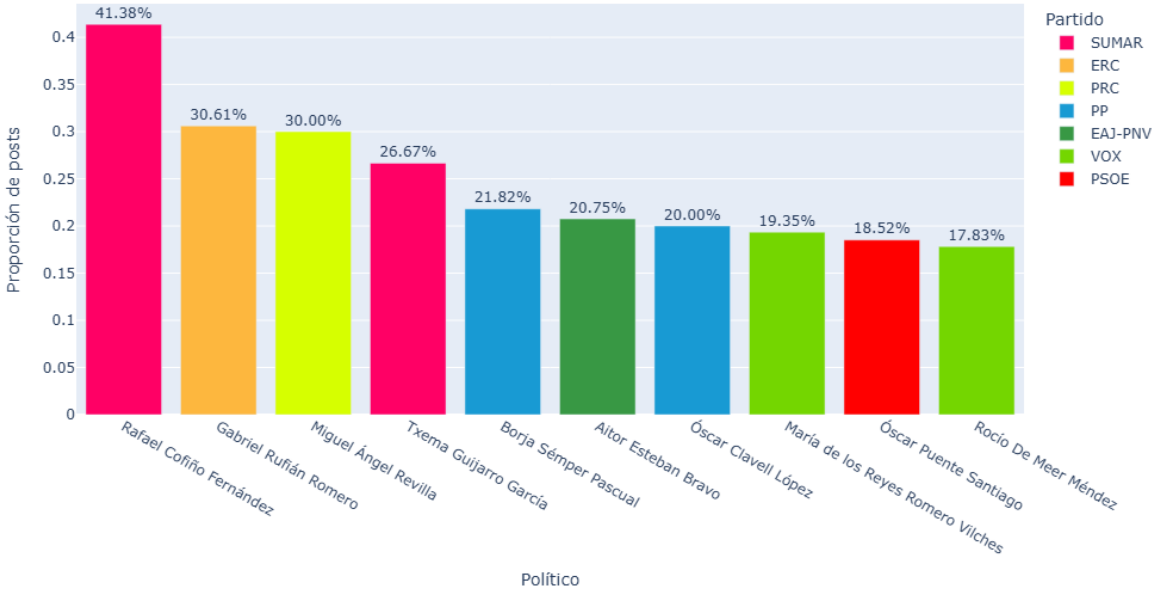

Ilustración 5.18: Top 10 políticos con mayor proporción de publicaciones neutras.

En el otro extremo, los políticos con un discurso más negativo fueron Montse Mínguez (PSOE), Miguel Tellado (PP) o Jaime de los Santos (PP), todos con más del 75 % de sus publicaciones en tono crítico. También destacaron perfiles como Míriam Nogueras (JxCAT-JUNTS) u Oskar Matute (EH Bildu), con una fuerte carga confrontativa en su discurso. A diferencia del discurso positivo, en este caso la totalidad del top 10 está compuesto exclusivamente por miembros del Congreso de los Diputados.

Finalmente, en cuanto a los mensajes neutros, figuras como Gabriel Rufián (ERC), Miguel Ángel Revilla (PRC) o Aitor Esteban (EAJ-PNV) mostraron mayor presencia de publicaciones con un tono más equilibrado o técnico.

Cuando se agruparon los datos por partidos, Coalición Canaria (CCa) lideró claramente en tono positivo (83,75 %), seguida del PSOE, EAJ-PNV o Más Madrid. En cambio, los partidos con un tono más negativo fueron VOX, ERC y JxCAT-JUNTS, todos por encima del 50 % de publicaciones críticas.

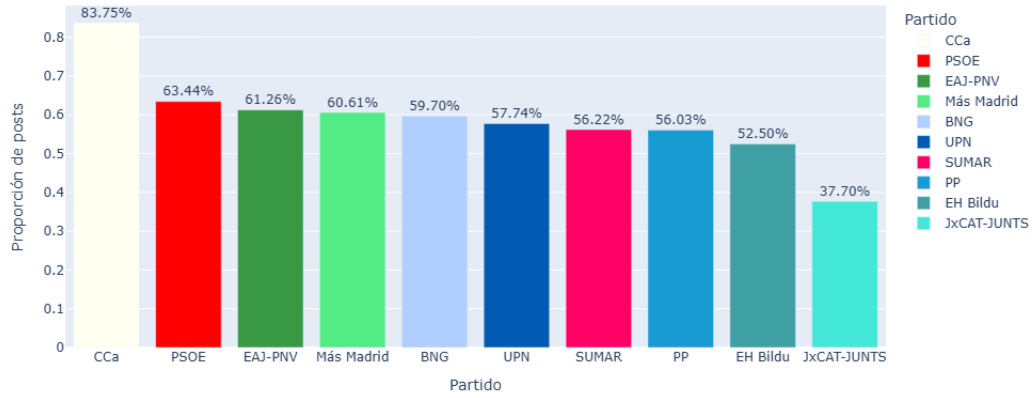

Ilustración 5.19: Proporción de publicaciones por tono positivo según partido político.

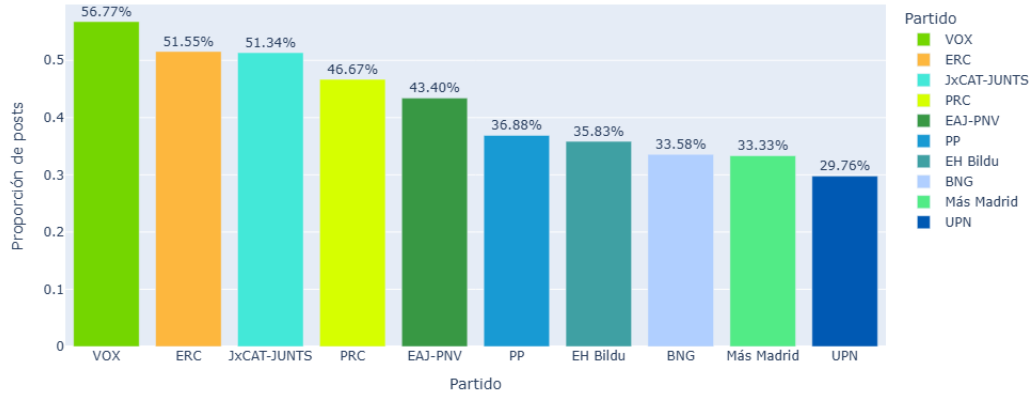

Ilustración 5.20: Proporción de publicaciones por tono negativo según partido político.

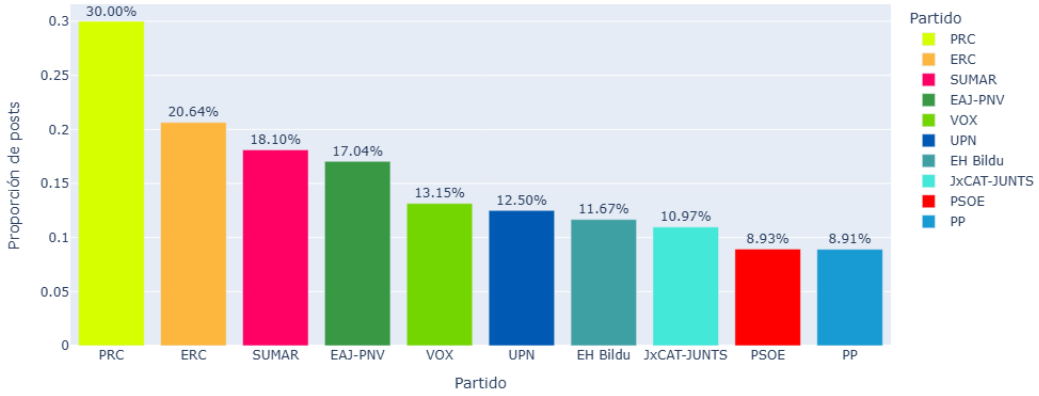

Ilustración 5.21: Proporción de publicaciones por tono neutro según partido político.

Respecto a la distribución territorial del tono, las comunidades con más proporción de mensajes positivos fueron Baleares, Murcia, Castilla y León, Canarias o Extremadura. Por el contrario, Cataluña, Madrid y La Rioja destacaron por su elevado nivel de negatividad en los mensajes. En cuanto a los discursos más neutros, las comunidades con mayor proporción fueron Cataluña, Cantabria y el País Vasco.

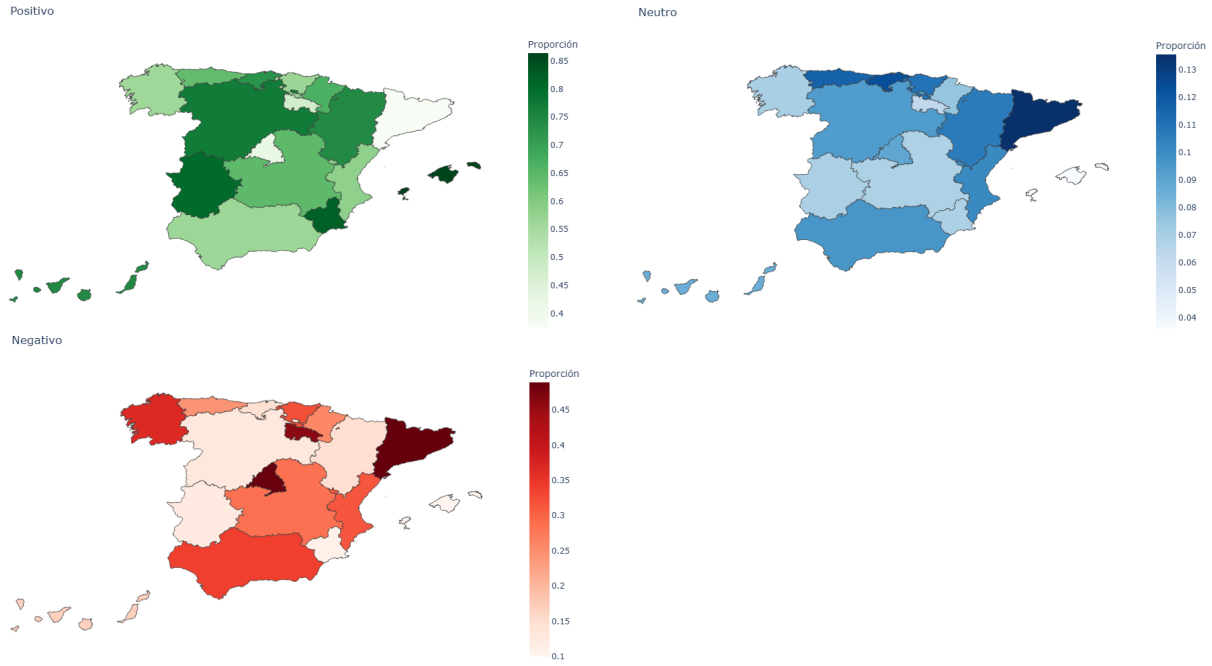

Ilustración 5.22: Distribución del tono por comunidad autónoma (positivo, neutro y negativo).

También se analizó el tono según los temas tratados. Los asuntos relacionados con igualdad, derechos sociales o economía se abordaron mayoritariamente con un tono positivo (85 %). En cambio, las publicaciones centradas en la gestión pública o las instituciones fueron las que mostraron un tono más negativo, con un 57,9 % de mensajes críticos.

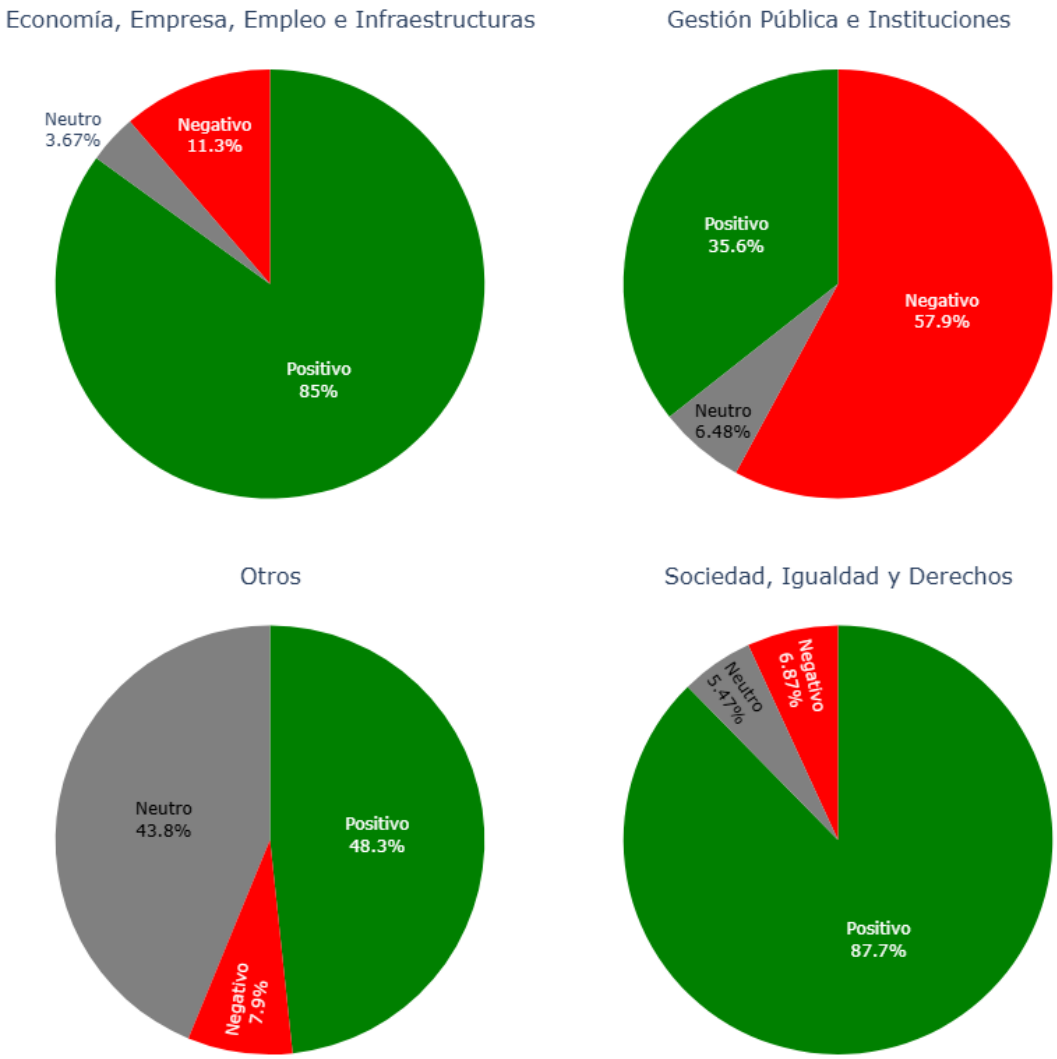

Ilustración 5.23: Distribución del tono por temáticas abordadas.

Conclusión del bloque: El tono del discurso político en X/Twitter varió en función del perfil, el partido, la temática tratada y el territorio. Mientras que el tono positivo fue usado en su mayoría por presidentes autonómicos, otros actores con roles más reivindicativos –como VOX, ERC o JxCAT-JUNTS– destacaron por un discurso más negativo. A nivel geográfico, Cataluña y Madrid se situaron como focos de mayor tensión en el discurso por sus altos porcentajes en tonalidad negativa.

5.2.5. Contenido: palabras clave y entidades

En este bloque se analizó de qué hablaban los políticos en X/Twitter y a quiénes mencionaban. Para ello, se estudiaron las palabras más utilizadas (tokens) y las entidades más

mencionadas (personas, lugares, partidos o instituciones), comparando su uso según el tipo de mensaje, el tono y el tema tratado.

En primer lugar, el tipo de mensaje influyó claramente en el lenguaje. En los **posts** aparecieron palabras más institucionales y con más carácter constructivo como *"trabajo"*, *"seguir"*, *"compromiso"* o *"futuro"*, con mensajes más formales o propositivos. En los **comentarios** se usó un lenguaje más directo y crítico, con términos como *"hacer"*, *"ahora"* o *"gente"*. Por su parte, las **respuestas** fueron más personales y moderadas, con palabras como *"usted"*, *"creo"* o *"respeto"*.

En cuanto a las entidades, algunas fueron comunes en todos los formatos, como *Gobierno*, *España*, *PP* o *Cataluña*. Pero en los comentarios y respuestas también aparecieron más nombres propios y temas polémicos, como *Ayuso*, *Feijóo*, *VOX*, *Zapatero* o *Rajoy*, lo que reflejó una conversación por lo general más centrada en figuras concretas.

El análisis del **tono** también mostró diferencias claras. En los mensajes positivos se usaron palabras relacionadas con el futuro, el esfuerzo o el reconocimiento, como *"mejor"*, *"futuro"* o *"compromiso"*. En los negativos, predominó la crítica con términos como *"pensiones"*, *"política"* o *"ley"*. En los mensajes neutros se habló más de condolencias o apoyo, con palabras como *"familia"*, *"situación"* o *"pésame"*.

También se analizaron las entidades más mencionadas según el tono. En los mensajes positivos se habló de regiones como *Navarra*, *Galicia*, *Andalucía*, *Aragón* o *Murcia*. En los negativos se repitieron nombres de líderes políticos y partidos como *Pedro Sánchez*, *Feijóo*, *VOX* o *Mazón*. Y en los mensajes neutros aparecieron menciones a instituciones, localidades concretas o figuras menos mediáticas.

Por último, se estudió el contenido según los **temas tratados**. Las publicaciones sobre *gestión pública* fueron las más críticas con términos como *"frente"* o *"solo"*, mientras que los temas de *economía* y *derechos sociales* se comunicaron con un tono mucho más positivo con tokens como *"compromiso"*, *"vamos"* u *"orgullo"*. Además, hubo un grupo de mensajes de carácter simbólico o emocional (como condolencias o felicitaciones) donde destacaron palabras como *"familia"*, *"pésame"* o *"cariño"*.

Conclusión del bloque: El contenido del discurso político en X/Twitter varió mucho según el tipo de mensaje, el tono y el tema. Los mensajes positivos hablaron más de gestión y futuro, mientras que los negativos se centraron en la crítica a partidos o líderes concretos. Las entidades mencionadas reflejaron tanto a los protagonistas del debate político como a territorios clave. En conjunto, el lenguaje usado mostró estrategias muy distintas según el perfil y el objetivo de cada publicación.

5.3. Análisis cruzado de indicadores

Este apartado resume los resultados más importantes cuando se ponen en relación todos los bloques analizados: popularidad, actividad, tono del discurso, interacción y contenido. El

objetivo era comprender mejor qué factores marcan la diferencia en el impacto digital de los políticos, y cómo influyen el contexto territorial o el tipo de mensajes que comparten.

5.3.1. Actividad vs. Interacción

Una de las primeras preguntas que se planteó en este trabajo fue si publicar mucho garantiza generar muchas reacciones. Los datos confirman que no siempre es así. Por ejemplo, **Gabriel Rufián (ERC)** o **Míriam Nogueras (JxCAT-JUNTS)** consiguieron equilibrar volumen de publicaciones con un alto nivel de interacción. Sin embargo, perfiles como **Oskar Matute (EH Bildu)**, que superan las 100.000 publicaciones, o **Cristina Narbona (PSOE)**, con más de 60.000, no lograron despertar la misma respuesta en proporción. Esto apunta a que no basta con publicar mucho, sino que el estilo, el tono y el tipo de mensaje juegan un papel clave.

Conclusión: La actividad intensa no garantiza impacto. Suelen generar más reacciones quienes comunican de forma directa, provocadora o con mensajes polarizantes, aunque publiquen menos.

5.3.2. Tono del discurso vs. Respuesta generada

También se quiso comprobar si un tono positivo impulsa la conversación. Sin embargo, los resultados muestran que no necesariamente ocurre así. Partidos con discursos en positivo, como **Coalición Canaria (84 % de mensajes positivos)** o el **PSOE (63 %)**, no figuran entre los que más interacciones consiguen. En cambio, partidos con mensajes críticos y negativos, como **ERC**, **JxCAT-JUNTS** o **VOX**, con porcentajes de publicaciones negativas en torno al 50–56 %, logran movilizar más al público.

Conclusión: Los mensajes críticos, confrontativos o polarizados generan más conversación e implicación en redes sociales. Aunque el tono positivo refuerza una imagen de serenidad institucional, tiende a activar menos debate.

5.3.3. Temas y nivel de debate

El contenido también marca diferencias claras. Los temas relacionados con la *gestión pública* y las *instituciones* concentran la mayor parte de los comentarios negativos y las críticas, con menciones frecuentes a figuras como Pedro Sánchez, Feijóo o al propio Gobierno. En cambio, asuntos como *igualdad, derechos sociales* o *economía* se comunican con un tono más optimista y provocan menos respuestas, quizá porque generan menos polémica.

Conclusión: Las cuestiones vinculadas a la gestión política y al funcionamiento de las instituciones son el mayor foco de debate y crítica. Los temas sociales o económicos, en cambio, son más aceptados y generan menos reacción inmediata.

5.3.4. Territorio y estrategia comunicativa

Cuando se mira el mapa, se aprecian diferencias claras según la comunidad. En **Cataluña**, el discurso político digital es mucho más crítico, movilizador y con alta interacción, especialmente entre los partidos independentistas como ERC o JxCAT-JUNTS. En comunidades como **Baleares**, **Murcia**, **Extremadura** o **Canarias**, predomina un tono positivo, centrado en mensajes de gestión, cohesión y orgullo territorial. Esto sugiere que el contexto político local condiciona fuertemente la forma de comunicar.

Conclusión: Cada territorio adapta el discurso a su realidad política. En zonas más polarizadas, el mensaje tiende a ser más duro y combativo; en otras regiones, la comunicación es más institucional y positiva.

5.3.5. Resumen general

En base a todas las explicaciones anteriores, se llega a la conclusión de que no hay una manera única y efectiva de destacar en X/Twitter. Algunos perfiles consiguieron impacto gracias a una gran constancia (como **Matute** o **Narbona**), mientras que otros lograron mucho con pocos mensajes pero muy contundentes (como **Rufián** o **Abascal**). La estrategia digital se apoyó casi siempre en tres pilares: confrontación con rivales políticos, exposición continua de la ideología de cada uno y presumir de los logros conseguidos. En todos los casos, se observó que el discurso digital está planificado al detalle según la identidad política de cada uno, el territorio en el que actúa y el público al que se dirige.

Capítulo 6

Conclusiones y trabajo futuro

6.1. Conclusiones

Este trabajo tuvo como objetivo principal analizar, mediante técnicas de ciencia de datos, cómo se comunican los políticos españoles en la red social X/Twitter. Para ello, se recopilaron y estudiaron más de 9.000 publicaciones de 78 miembros del Congreso de los Diputados de distintos partidos y representantes de comunidades autónomas.

A lo largo del análisis se comprobó que el estilo comunicativo de la clase política española en redes sociales no fue homogéneo. Algunos perfiles se decantaron por una estrategia de publicaciones constantes, mientras que otros publicaron menos, pero con mensajes más directos o polémicos. También se identificaron diferencias claras entre partidos y territorios: los líderes de partidos nacionalistas y regionales, especialmente en Cataluña y el País Vasco, fueron más activos y críticos, mientras que otros políticos, especialmente presidentes autonómicos de partidos de ámbito estatal, usaron un tono más cordial y positivo.

El análisis desarrollado permitió cumplir satisfactoriamente los objetivos definidos al inicio del proyecto. Se calcularon indicadores de popularidad e interacción para evaluar el impacto real de las cuentas políticas; se identificaron patrones regionales mediante visualizaciones geográficas; se aplicó un análisis de sentimientos para clasificar el tono de los mensajes y de las interacciones recibidas; y, por último, se extrajeron palabras clave y temas dominantes, relacionándolos con su carga emocional y su contexto discursivo.

Cabe señalar que, aunque los datos recopilados ofrecían la posibilidad de realizar muchos más análisis, comparaciones o visualizaciones, se optó por mantener el enfoque centrado en los objetivos iniciales. Esta decisión respondió a la necesidad de respetar tanto los límites de tiempo como de extensión del trabajo.

En relación con el objetivo adicional planteado sobre el desarrollo de una herramienta interactiva, finalmente esta pudo ser implementada utilizando Streamlit, integrando así las visualizaciones dinámicas generadas con Plotly. Esto permitió ofrecer un entorno accesible y sencillo a la hora de explorar los resultados, mejorando así el plan original de mostrar únicamente gráficos estáticos en documentos. La herramienta final facilita la navegación por los datos y el análisis visual de los patrones detectados, cumpliendo con el objetivo de acercar los resultados a usuarios con distintos niveles de conocimiento técnico.

En conclusión, el estudio confirmó que X/Twitter es un canal donde la estrategia, el contexto y el tono del mensaje condicionan fuertemente su alcance e impacto político. No existe una única fórmula para comunicar con éxito: cada perfil adapta su estilo en función de su rol, su partido, su entorno territorial y su público. Esta diversidad demuestra que la comunicación política digital es una práctica profundamente estratégica y contextualizada.

6.2. Trabajo futuro

A partir de este proyecto han surgido muchas ideas interesantes que podrían desarrollarse más adelante. Lo más evidente sería mejorar el modelo de clasificación de tono y, sobre todo, tema. Tener un buen clasificador de temas permitiría identificar categorías más concretas y así evitar tener que limitarlo a etiquetas tan generales.

También se podría ampliar el grupo de políticos analizados, incluyendo alcaldes de ciudades relevantes, eurodiputados u otros representantes activos en redes sociales. Esto ayudaría a obtener una visión más completa y diversa de la comunicación política en aspectos tan concretos como pueden ser la gestión municipal o la postura de los distintos partidos con respecto a la Unión Europea. Además, ampliar la muestra permitiría analizar con más detalle el grado de cercanía entre los políticos y el electorado, es decir, comprobar si la interacción aumenta cuando los representantes están más próximos a la ciudadanía.

Otra línea de trabajo interesante sería aumentar el número de publicaciones y comentarios estudiados, e ir actualizando los datos periódicamente. De esta forma, se podría analizar cómo cambian los discursos y las respuestas ciudadanas a lo largo del tiempo. También permitiría relacionar campañas electorales o situaciones de impacto nacional (elecciones, desastres, eventos deportivos) con el aumento o disminución en el número de publicaciones de un partido.

En cuanto a la aplicación de visualización, quedaría por seguir mejorándola e incorporar más tipos de gráficos o funcionalidades, como la introducción de redes de relaciones entre tokens/entidades.

Por último, convendría plantear el uso de la API de pago de X/Twitter para la recolección de datos de forma sostenible. Frente a las técnicas de *scraping* realizadas en este trabajo, las cuales pueden verse limitadas por cambios en la estructura web o restricciones de acceso y que corren el riesgo de ser bloqueadas, la API oficial ofrece un entorno más estable para obtener grandes volúmenes de cuentas y recopilar datos en tiempo real. De este modo, se garantizaría la calidad, continuidad y legalidad del proceso de obtención de datos.

Uno de los puntos más positivos de este trabajo es que se han generado muchas cuestiones que podrían ser objeto de investigación en el futuro, como por ejemplo:

- ¿Se comunican de forma diferente los hombres y las mujeres? ¿Reciben un tipo de respuesta distinta? ¿Depende de la edad o el partido?
- ¿Influye el nivel de estudios en la calidad del discurso?

- ¿Tienen los partidos regionalistas o nacionalistas más apoyo digital? ¿Depende más del partido o de la persona?
- ¿La edad o la experiencia política afecta al tono de los mensajes?
- ¿Qué temas generan respuestas más positivas o negativas por parte de la población? ¿El tono de la respuesta es independiente del mensaje original?

Aunque este proyecto ha cumplido los objetivos iniciales, queda claro que existen aún muchas líneas abiertas en las que profundizar sobre cómo los políticos españoles usan las redes sociales y cómo responde la ciudadanía a sus mensajes.

6.3. Logros personales y aprendizaje

A nivel personal, este trabajo me ha permitido adquirir muchas competencias nuevas. Por ejemplo, he aprendido a desarrollar procesos de *scraping* en entornos dinámicos, ya que hasta ahora solo había trabajado con páginas estáticas. También he podido diseñar y desplegar una aplicación web desde cero con Streamlit, algo totalmente nuevo para mí. Además, me he enfrentado a la organización de un proyecto completo en solitario, lo que ha sido muy enriquecedor.

Durante la obtención de los datos surgieron varios retos que pude resolver con un poco de creatividad, como los bloqueos al scrapear publicaciones de la red X/Twitter, la gestión de los posts fijados o la ralentización en la descarga de mensajes. Esto último pude solventarlo gracias a la paralelización y al uso de cookies de sesión. Asimismo, la creación de la propia aplicación fue un desafío importante debido a mi experiencia limitada en desarrollo de interfaces web.

En cuanto a habilidades personales, destaco la capacidad de análisis y de resolución de problemas que he fortalecido durante el proyecto, así como una mayor autonomía para investigar, priorizar tareas y documentar mis avances. Todo este proceso me ha motivado a seguir aprendiendo y explorar en el futuro, proyectos de análisis de datos aplicados tanto a la política como a otros temas de interés.

Bibliografía

[1] Abejon, P., Sastre, A., and Linares, V. (2012). Facebook y Twitter en campañas electorales en España. *Anuario Electrónico de Estudios en Comunicación Social "Disertaciones"*, 5(1):129–159.

[2] Abodayeh, A., Hejazi, R., Najjar, W., Shihadeh, L., and Latif, R. (2023). Web Scraping for Data Analytics: A BeautifulSoup Implementation. In *2023 Sixth International Conference of Women in Data Science at Prince Sultan University (WiDS PSU)*, pages 65–69. IEEE.

[3] Alonso-Muñoz, L. and Casero-Ripollés, A. (2023). The appeal to emotions in the discourse of populist political actors from Spain, Italy, France and the United Kingdom on Twitter. *Frontiers in Communication*, 8:1159847.

[4] Antypas, D., Ushio, A., Camacho-Collados, J., Neves, L., Silva, V., and Barbieri, F. (2022). Twitter topic classification. *arXiv preprint arXiv:2209.09824*.

[5] Aziz, Z. A., Abdulqader, D. N., Sallow, A. B., and Omer, H. K. (2021). Python Parallel Processing and Multiprocessing: A Review. *Academic Journal of Nawroz University (AJNU)*, 10(3):344–353.

[6] Bale, A. S., Ghorpade, N., S, R., Kamalesh, S., R, R., and S, R. B. (2022). Web Scraping Approaches and their Performance on Modern Websites. In *2022 3rd International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 1412–1518. IEEE.

[7] Beriain Bañares, A., Crisóstomo Gálvez, R., and Chiva Molina, I.-P. (2022). Comunicación política en España: representación e impacto en redes sociales de los partidos en campaña. *Revista Mexicana de Ciencias Políticas y Sociales*, 67(246):337–364.

[8] Bonilla, Y. and Rosa, J. (2015). #Ferguson: Digital protest, hashtag ethnography, and the racial politics of social media in the United States. *American Ethnologist*, 42(1):4–17.

[9] BotPenguin Team (2024). Fine-Tuning – Glossary. Consultado en junio de 2025.

[10] Boulianne, S. (2015). Social media use and participation: a meta-analysis of current research. *Information, Communication & Society*, 18(5):524–538.

[11] Boulianne, S. and Larsson, A. O. (2021). Engagement with candidate posts on Twitter, Instagram and Facebook during the 2019 election. *New Media & Society*, 25(1):117–139.

[12] Carral, U. and Tuñón-Navarro, J. (2020). Estrategia de comunicación organizacional en redes sociales: análisis electoral de la extrema derecha francesa en Twitter. *Profesional de la información*, 29(6):1–15.

[13] Centro de Estudios Murciano de Opinión Pública (CEMOP) (2022). II Encuesta Nacional de Polarización Política. [online] Universidad de Murcia. Disponible en: https://www.um.es/cemop/informes/2022-encuesta-nacional-polarizacion.pdf.

[14] Chowdhury, G. G. (2003). Natural Language Processing. *Annual Review of Information Science and Technology*, 37:51–89.

[15] DataReportal (2025). Essential X Stats & Digital 2025 Spain. [online] Disponible en: https://datareportal.com/reports/digital-2025-spain.

[16] del Olmo, F. J. R. and Díaz, J. B. (2017). La evolución del debate televisivo como herramienta de comunicación política. Análisis del caso español: de la televisión a Twitter. *Informação & Sociedade: Estudos*, 27(2):235–252.

[17] Del Valle Martín, E. and de la Fuente Valentín, L. (2023). Sentiment Analysis Methods for Politics and Hate Speech Contents in Spanish Language: A Systematic Review. *IEEE Latin America Transactions*, 21(3):408–418.

[18] Díaz Cuervo, G. and Barbera González, R. (2024). Personalización política en redes sociales: uso de Twitter e Instagram en la campaña electoral de 2021 en la Comunidad de Madrid. *Palabra Clave*, 27(2):1–25.

[19] Engesser, S., Ernst, N., Esser, F., and Büchel, F. (2016). Populism and social media: how politicians spread a fragmented ideology. *Information, Communication & Society*, 20(8):1109–1126.

[20] Fraunhofer Institute for Surface Engineering and Thin Films IST (n.d.). Implementation strategy for a data-mining project with CRISP-DM in surface technology. https://www.ist.fraunhofer.de/en/expertise/simulation-digital-services/data-acquisition-model-based-process-optimization/crisp-dm-surface-technology.html. Consultado el 15 de mayo de 2025.

[21] Giatsoglou, M., Chatzakou, D., Salamanos, N., Diamantaras, K. I., Vakali, A., and Vassiliadis, A. (2017). Sentiment analysis leveraging emotions and word embeddings: A Twitter study. *Information Processing & Management*, 54(5):813–831.

[22] Giuggioli, L. and Pellegrini, T. (2022). Lexical approaches to sentiment analysis in highly inflected languages. *Journal of Computational Linguistics and Applications*, 14(2):55–74.

[23] Gojare, S., Joshi, R., and Gaigaware, D. (2015). Analysis and design of selenium webdriver automation testing framework. *International Journal of Computer Applications*, 120(22):341–346.

[24] González-Pizarro, L., Fanjul-Peyro, C., and De-La-Fuente-Cabrero, C. (2021). Interacciones en Twitter entre audiencias e influencers. Metodología de análisis de sentimiento, polaridad y comportamiento comunicacional. *Revista Latina de Comunicación Social*, 79:1–22.

[25] Habibi, M. N. and Sunjana (2019). Analysis of Indonesia Politics Polarization before 2019 President Election Using Sentiment Analysis and Social Network Analysis. *International Journal of Modern Education and Computer Science (IJMECS)*, 11(11):22–30.

[26] Hinguruduwa, L., Marx, E., Soru, T., and Riechert, T. (2021). Assessing Language Identification over DBpedia. In *2021 IEEE 15th International Conference on Semantic Computing (ICSC)*, pages 296–297. IEEE.

[27] Iman, Z., Sanner, S., Bouadjenek, M. R., and Xie, L. (2017). A longitudinal study of topic classification on twitter. In *Proceedings of the International AAAI Conference on Web and Social Media*, pages 552–555.

[28] Jungherr, A. (2016). Twitter use in election campaigns: A systematic literature review. *Journal of Information Technology & Politics*, 13(1):72–91.

[29] Khder, M. A. (2021). Web scraping or web crawling: State of art, techniques, approaches and application. *International Journal of Advances in Soft Computing & Its Applications*, 13(3):145–168.

[30] Koroteev, M. V. (2021). BERT: a review of applications in natural language processing and understanding. *arXiv preprint arXiv:2103.11943*.

[31] Laurikkala, J. (2001). Improving Identification of Difficult Small Classes by Balancing Class Distribution. In *Artificial Intelligence in Medicine: 8th Conference on Artificial Intelligence in Medicine in Europe, AIME 2001*, pages 63–66, Cascais, Portugal. Springer Berlin Heidelberg.

[32] Lee, K.-O., Palsetia, D., Narayanan, R., Patwary, M. M., Agrawal, A., and Choudhary, A. (2011). Twitter trending topic classification. In *2011 IEEE 11th International Conference on Data Mining Workshops*, pages 251–258. IEEE.

[33] Lin, T.-Y., Goyal, P., Girshick, R., He, K., and Dollár, P. (2017). Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988.

[34] Lotfi, C., Srinivasan, S., Ertz, M., and Latrous, I. (2021). Web Scraping Techniques and Applications: A Literature Review. In *SCRS Conference Proceedings on Intelligent Systems*, pages 381–394, Saguenay, Canada. SCRS. LaboNFC, University of Quebec at Chicoutimi.

[35] Masnita, Y., Kasuma Ali, J., Zahra, A., Wilson, N., and Murwonugroho, W. (2024). Artificial Intelligence in Marketing: Literature Review and Future Research Agenda. *Journal of System and Management Sciences*, 14(1):120–140.

[36] Matalon, Y., Magdaci, O., Almozlino, A., and Yamin, D. (2021). Using sentiment analysis to predict opinion inversion in Tweets of political communication. *Scientific Reports*, 11(1):7250.

[37] McKinney, W. (2011). pandas: a foundational Python library for data analysis and statistics. *Python for High Performance and Scientific Computing*, 14(9):1–9.

[38] Milosav, D., Dickson, Z., Hobolt, S. B., Klüver, H., Kuhn, T., and Rodon, T. (2025). The youth gender gap in support for the far right. *Journal of European Public Policy*, in press:1–25.

[39] Mukhoti, J., Kulharia, V., Sanyal, A., Golodetz, S., Torr, P. H. S., and Dokania, P. K. (2020). Calibrating deep neural networks using focal loss. *Advances in Neural Information Processing Systems*, 33:15288–15299.

[40] Reinhardt, W., Ebner, M., Beham, G., and Costa, C. (2009). How people are using twitter during conferences. In *Creativity and Innovation Competencies on the Web. Proceedings of the 5th EduMedia*, pages 145–156, Salzburg, Austria.

[41] Rish, I. (2001). An Empirical Study of the Naive Bayes Classifier. In *IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence*, pages 41–46, Seattle, USA. IJCAI.

[42] Rodríguez-Ibáñez, M., Gimeno-Blanes, F. J., Cuenca-Jiménez, P. M., Soguero-Ruiz, C., and Rojo-Álvarez, J. L. (2021). Sentiment analysis of political tweets from the 2019 Spanish elections. *IEEE Access*, 9:101847–101862.

[43] Smith, B. (2024). A Complete Guide to BERT with Code. Consultado en junio de 2025.

[44] Statista (2020). Opinión sobre la influencia de las redes sociales en temas políticos en la UE 2018. [online] Disponible en: https://es.statista.com/estadisticas/606211/.

[45] Statista (2025). X (Twitter): distribución porcentual de los usuarios por género en España. [online] Disponible en: https://es.statista.com/estadisticas/515756/.

[46] Stolee, G. and Caton, S. (2018). Twitter, Trump, and the Base: A Shift to a New Form of Presidential Talk? *Signs and Society*, 6(1):147–165.

[47] Svetlov, K. and Platonov, K. (2019). Sentiment Analysis of Posts and Comments in the Accounts of Russian Politicians on the Social Network. In *Proceedings of the 25th Conference of FRUCT Association*, pages 299–306, Yaroslavl, Russia. FRUCT Association.

[48] Tan, W., Ruan, Y., Liang, L., Zhang, Y., and He, X. (2021). Neutrality detection for sentiment analysis: A survey. *arXiv preprint arXiv:2109.07597*.

[49] Tripathy, R. M., Sharma, S., Joshi, S., Mehta, S., and Bagchi, A. (2014). Theme based clustering of tweets. In *Proceedings of the 1st IKDD Conference on Data Sciences*, pages 1–5.

[50] Uzun, E., Yerlikaya, T., and Kirat, O. (2018). Comparison of Python Libraries Used for Web Data Extraction. In *International Scientific Conference on Engineering, Technologies and Systems (TECHSYS 2018)*, pages 87–92. Technical University – Sofia, Plovdiv branch.

[51] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30:1–11.

[52] We Are Social (2025). Digital 2025: La guía esencial del estado global de lo digital. https://wearesocial.com/es/blog/2025/02/digital-2025-la-guia-esencial-del-estado-global-de-lo-digital/.

[53] Weller, K., Bruns, A., Burgess, J., Mahrt, M., and Puschmann, C., editors (2013). *Twitter and Society*. Peter Lang, New York.

[54] Zhang, Z. and Sabuncu, M. R. (2018). Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. In *Advances in Neural Information Processing Systems*.

[55] Zhao, L., Chen, F., Lu, C., and Ramakrishnan, N. (2015). Dynamic theme tracking in Twitter. In *2015 IEEE International Conference on Big Data (Big Data)*, pages 561–570. IEEE.

[56] Zhou, C., Li, Q., Li, C., Yu, J., Liu, Y., Wang, G., and Sun, L. (2024). A comprehensive survey on pretrained foundation models: A history from bert to chatgpt. *International Journal of Machine Learning and Cybernetics*, pages 1–65.