UNIVERSITAT POLITÈCNICA DE VALÈNCIA

DSIC DEPARTAMENT DE SISTEMES INFORMÀTICS I COMPUTACIÓ

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

Dpto. de Sistemas Informáticos y Computación

Generación automática de resúmenes y extracción de ideas principales en noticias web

Trabajo Fin de Máster

Máster Universitario en Inteligencia Artificial, Reconocimiento de Formas e Imagen Digital

AUTOR/A: Asensio Benedicto, Ana Isabel

Tutor/a: Domingo Ballester, Miguel

Cotutor/a externo: García Rubio, Pedro

CURSO ACADÉMICO: 2023/2024

UNIVERSITAT POLITÈCNICA DE VALÈNCIA

DSIC

Departament de Sistemes Informàtics i Computació
Universitat Politècnica de València

Generación automática de resúmenes y extracción de las ideas principales en noticias web

TRABAJO FIN DE MÁSTER

Máster Universitario en Inteligencia Artificial, Reconocimiento de Formas e Imagen Digital

Autora: Ana Isabel Asensio Benedicto

Tutor UPV: Miguel Domingo Ballester

Curso 2023-2024

Resumen

Actualmente, el acceso a internet está al alcance de la mano de la mayor parte de la población, lo que facilita la publicación y consumo de información. Todos los días se publica una gran cantidad de noticias nuevas a un ritmo difícil de seguir. Es por ello que, para poder estar al día y seleccionar aquellas noticias que realmente son de interés de forma rápida, surge la utilidad de este trabajo. Su objetivo es la generación automática de los resúmenes de dichas noticias y la extracción de las ideas principales que contienen. Para ello, se ha utilizado noticias reales como base de datos pertenecientes al campo de interés de la empresa colaboradora ForwardKeys, así como otras bases de datos externas para el entrenamiento de modelos. Estos datos se han modelado a partir de modelos preentrenados de las familias de Pegasus y T5, contando con un modelo de generación de resúmenes general y otro específico entrenado únicamente con texto de noticias. La evaluación de estos modelos se ha realizado a través de la métrica ROUGE y sus variantes.

Palabras clave: generación de resúmenes; extracción de ideas principales; inteligencia artificial; Transformers; procesamiento de lenguaje natural

Abstract

Nowadays, access to Internet is within the reach of most population, which facilitates the publication and consumption of information. Every day, a large amount of new news is published at a pace that is difficult to keep up with. This is why, to keep up to date and select those news that are of real interest quickly, the usefulness of this work arises. Its aim is the automatic generation of summaries of the aforementioned news and the extraction of the key ideas they contain. With this aim, real news have been used as a database belonging to the field of interest of the collaborating company ForwardKeys, as well as other external databases for model training. These data have been modeled from pre-trained models of the Pegasus and T5 families, with a general summary generation model and a specific model trained only with news text. The evaluation of these models has been performed through the ROUGE metric and its variants.

Key words: summarization; key ideas extraction; artificial intelligence; Transformers; natural language processing

Resum

Actualment, l'accés a internet està a l'abast de la mà de la major part de la població, la qual cosa facilita la publicació i consum d'informació. Tots els dies es publiquen una gran quantitat de notícies noves a un ritme difícil de seguir. És per això que, per a poder estar al dia i seleccionar aquelles notícies que realment són d'interés de manera ràpida, sorgix la utilitat d'este treball. El seu objectiu és la generació automàtica dels resums d'estes notícies i l'extracció de les idees principals que contenen. Per a això, s'han utilitzat notícies reals com a base de dades pertanyents al camp d'interés de l'empresa col·laboradora ForwardKeys, així com altres bases de dades externes per

a l'entrenament de models. Estes dades han sigut modelades a partir de models preentrenats de les famílies de Pegasus i T5, comptant amb un model de generació de resums general i un altre específic entrenat únicament amb text de notícies. L'avaluació d'estos models s'ha fet a través de la mètrica ROUGE i les seues variants.

Paraules clau: generació de resums; extracció d'idees principals; intel·ligència artificial; Transformers; processament de llenguatge natural

Índice general

Índice de figuras

Índice de tablas

CAPÍTULO 1
Introducción

Actualmente, el acceso a Internet está al alcance de la mano de gran parte de la población mundial y, por tanto, a la vasta información disponible en la red. Esta situación consta de una parte muy beneficiosa que consiste en la posibilidad de estar documentados de cualquier tema que se desee con solo unos pocos clics. Sin embargo, esta realidad también puede presentar un factor negativo, y es que esta sobreexposición constante a noticias y textos puede resultar abrumadora, ya que al día pueden publicarse miles de noticias y es complejo estar informado de todas ellas. Es por ello que, si se desea estar lo más actualizado posible, la capacidad para sintetizar grandes volúmenes de datos en resúmenes breves y concisos toma una fuerza relevante y necesaria. Para poder llevar a cabo esta capacidad, es vital la generación automática de resúmenes y la extracción de ideas principales. Estas técnicas son esenciales en el procesamiento del lenguaje natural (NLP; del inglés *natural language processing*), el aprendizaje automático y la inteligencia artificial, permitiendo transformar textos extensos en versiones más manejables, tratando de preservar a la vez el significado y la esencia del contenido original.

La generación automática de resúmenes hace referencia al proceso a partir del cual los algoritmos y modelos crean síntesis de textos de forma automática. Actualmente, existen dos enfoques principales para la extracción de estos resúmenes: la forma extractiva, donde se seleccionan y combinan frases directamente del texto original; y la técnica abstractiva, donde se generan frases nuevas que capturan las ideas principales del texto.

Por otro lado, la extracción de ideas principales se enfoca en identificar y resaltar los conceptos más importantes así como las ideas clave dentro de un texto, facilitando una comprensión rápida y eficiente de la información esencial. Esta extracción da como resultado una frase completa que podría asemejarse a un titular, ya que trata de englobar la idea principal del texto sin entrar en detalles. Esta técnica es particularmente útil en campos como la investigación académica, el periodismo, y el análisis de grandes documentos corporativos.

Este trabajo se enmarca dentro de una colaboración con la empresa ForwardKeys[1], una compañía cuya actividad se centra en el diseño, el desarrollo y la comercialización de sistemas informáticos para la prestación de servicios relacionados con la industria del viaje. Más concretamente, el trabajo se relaciona con la parte de inteligencia de

[1] https://forwardkeys.com/.

negocio, ya que existe una necesidad de estar actualizado con las noticias que puedan surgir en el día a día relacionadas con este ámbito. Habitualmente, este trabajo se realiza de forma manual, lo que implica un consumo de tiempo y un esfuerzo humano que podría verse considerablemente reducido mediante el uso de herramientas de inteligencia artificial.

Gracias al avance de la ciencia y, en particular, de la inteligencia artificial, hoy en día existen diversas herramientas y modelos que ya permiten extraer estos resúmenes y mensajes clave de los textos sean estos de diversas índoles. En este trabajo en concreto, se apostará por encontrar el modelo que mejor consiga sintetizar noticias en línea sobre turismo, aviación y temáticas relacionadas con vuelos y viajes. Para ello, se van a utilizar vairas bases de datos distintas. En primer lugar, se harán servir aquellas noticias recogidas en Feedly por parte de ForwardKeys según la selección que han hecho sus empleados de aquellas noticias y temáticas que resultan útiles para la empresa. En segundo lugar, se utilizará una base de datos con noticias de la BBC[2] etiquetadas con su correspondiente resumen y por último otro conjunto de datos de noticias etiquetadas tanto con el resumen como con las ideas clave extraídas de estas. Estas dos últimas bases de datos se han extraído del repositorio de Kaggle[3].

1.1 Motivación

Como se ha mencionado anteriormente, la gran cantidad de información nueva que se genera cada día supone un problema significativo cuando se trata de mantenerse al tanto de todos los nuevos artículos publicados o de seleccionar qué información es verdaderamente relevante para el usuario. Esta situación es especialmente crítica en un entorno en el que la toma de decisiones depende en gran medida de la actualidad de la información. A diario se publican miles de noticias, informes y estudios que podrían contener datos de valor para distintos sectores, pero el volumen y la velocidad a la que esta información aparece hacen prácticamente imposible procesarlo todo de manera manual.

El acto de leer noticias con la esperanza de que sean relevantes, solo para descubrir al final que no lo son, consume un tiempo considerable. Este problema no solo afecta a individuos que desean estar informados, sino también a empresas que basan sus decisiones estratégicas en información precisa y actualizada. En estos casos, dedicar tiempo a filtrar manualmente qué contenido es valioso puede resultar en pérdidas de productividad, especialmente en sectores donde las decisiones rápidas y bien informadas son clave para mantenerse competitivos.

En este contexto, surge la necesidad de desarrollar herramientas que automaticen la generación de resúmenes, capaces de filtrar la información irrelevante y ofrecer a los usuarios solo lo más importante de manera rápida y eficiente. ForwardKeys, empresa colaboradora en este Trabajo de Final de Máster, enfrenta precisamente este desafío. En esta empresa es fundamental estar al tanto de todas las noticias que puedan surgir diariamente, ya que estas pueden contener información muy valiosa sobre tendencias de viaje, eventos globales, restricciones y regulaciones que afectan al comportamien-

[2] https://www.bbc.com/news.
[3] https://www.kaggle.com/.

to de los viajeros, o cambios importantes en puntos turísticos de interés. Este tipo de información es crucial para entender fluctuaciones en los patrones de viaje y para anticiparse a eventos que podrían tener un impacto en los datos que la empresa maneja.

Además, al tratarse de una empresa que analiza grandes volúmenes de datos sobre movimientos de pasajeros y tendencias de viajes a nivel global, la capacidad de obtener resúmenes rápidos y precisos puede marcar la diferencia entre una reacción inmediata y una oportunidad perdida. La información recopilada a través de estas noticias y reportes puede ayudar a entender mejor el contexto en el que se desarrollan ciertos eventos - ya sean de naturaleza económica, política o social - y cómo estos afectan directamente a la industria del turismo.

Esta necesidad de estar constantemente informados es el motor principal que ha motivado este trabajo, ya que no solo se pretende crear una solución que facilite el procesamiento de grandes volúmenes de información, sino que también se busca que esta herramienta actúe como un punto de partida para la implementación más amplia de inteligencia artificial en ForwardKeys. La generación automática de resúmenes no solo puede aumentar la eficiencia operativa dentro de la empresa, sino que además abre la puerta a nuevas oportunidades para mejorar la toma de decisiones basadas en datos y la identificación de tendencias en tiempo real.

De hecho, la creación de este tipo de soluciones basadas en inteligencia artificial y aprendizaje automático permite que la empresa dé un paso hacia la automatización de procesos que antes dependían en gran medida del esfuerzo humano. Con la capacidad de reducir el ruido informativo y centrarse en los datos más relevantes, ForwardKeys podría optimizar su análisis de información además de sus estrategias de mercado, sus previsiones sobre el comportamiento de los viajeros y su respuesta a cambios inesperados en el panorama mundial.

Este proyecto, por tanto, tiene como objetivo principal resolver una necesidad inmediata en la empresa, que se amplía a sentar las bases para futuros desarrollos en inteligencia artificial que permitan a ForwardKeys seguir innovando y adaptándose a un entorno empresarial cada vez más complejo y competitivo. Con la implementación de una herramienta automatizada de generación de resúmenes, se espera mejorar tanto la eficiencia en el manejo de la información como la capacidad de la empresa para responder con rapidez y precisión a los cambios que surgen diariamente en su sector.

1.2 Objetivos

En ForwardKeys se ha planteado como objetivo principal el desarrollo de una herramienta para la generación automática de resúmenes y la extracción de las ideas clave a partir de noticias web. Esta herramienta tiene como finalidad que sirva de ayuda a los empleadods para ser más rápidos y eficientes a la hora de leer y seleccionar aquellas noticias que pueden aportar valor real a los trabajos de la empresa. Este objetivo general, al ser tan amplio y ambicioso, se ha dividido a su vez en objetivos más específicos para mayor claridad y abordaje:

- Recopilación de una base de datos de artículos con un índice de similitud cercano al de las noticias pertenecientes a ForwardKeys

- Uso de modelos preentrenados para la generación automática de resúmenes de noticias web
- *Fine-tuning* de modelos preentrenados para la generación automática de resúmenes
- Uso de modelos preentrenados para la extracción de ideas principales de noticias web
- *Fine-tuning* de modelos preentrenados para la extracción de ideas principales

Para la obtención de los resultados finales y, por tanto, consecución de objetivos, se empleará la métrica ROUGE [19] para medir cómo de bueno es el resumen y lo bien que se consigue capturar la esencia del texto en la idea extraída. Esta métrica vendrá acompañada con distintas variantes que se analizarán detalladamente para poder seleccionar el mejor modelo.

Cabe recalcar que el alcance de este trabajo no busca poner en funcionamiento la herramienta durante el desarrollo del proyecto, sino que el objetivo principal, como bien se ha expuesto al principio de este apartado, consiste en desarrollar la herramienta de modelaje que permita la generación de estos resúmenes e ideas principales. En función de los resultados obtenidos tras este estudio, se tratará de mejorar las métricas en la medida de lo posible y posteriormente poner en funcionamiento el modelado para que pueda utilizarse en el día a día de la empresa.

1.3 Impacto esperado

Este trabajo tiene un impacto directo en la empresa involucrada. Principalmente, se busca ayudar y automatizar el trabajo de algunos de los empleados que dedican gran parte de su tiempo laboral a la lectura de artículos en busca de nuevos enfoques y líneas de trabajo. A pesar de que hoy en día existen muchas herramientas que permiten ya la generación de estos resúmenes, el valor en este trabajo se aporta al entrenar los modelos focalizados en el ámbito realmente importante para la empresa. Los generadores automáticos de resúmenes y de ideas principales son más generalistas, por lo que se espera que este estudio consiga unos resúmenes más precisos y acordes con los artículos de interés propios.

En cuanto a la parte humana, esta generación automática de resúmenes se realiza buscando una mejora de rendimiento, permitiendo a estas personas aprovechar su tiempo de forma más creativa llevando a cabo tareas propias de humanos más relacionadas con pensamiento crítico y desarrollo de ideas novedosas que permitan crecer a la empresa. Por otro lado, también se espera seguir abriendo las puertas a la inteligencia artificial dentro de la empresa, ya bien sea con temas relacionados con la lingüística computacional o con campos numéricos (detección de tendencias, de datos anómalos, anticipación de eventos, desarrollo de nuevos productos, etc.).

A día de hoy, el desarrollo e implementación de este trabajo tendrá un impacto meramente intraempresarial, pero podría ser el detonante para la innovación e investigación de esta rama para mejorar los procesos de ForwardKeys.

1.4 Estructura de la memoria

A continuación se exponen los distintos apartados que conforman la estructura de esta memoria para facilidar la comprensión total de la tarea que se ha llevado a cabo:

- ***Capítulo 1. Introducción***. En este primer capítulo se expone una idea generalista de la temática que se ha abordado durante la realización de este trabajo así como la motivación que ha llevado a ello. También se presentan presentan los objetivos y el impacto que se prentende conseguir tras la finalización del proyecto.
- ***Capítulo 2. Estado del arte***. Durante el estado del arte se hará una revisión de algunos temas estrechamente relacionados con la generación automática de resúmenes así como los trabajos más destacados y actuales que hay sobre esta área de procesamiento de lenguaje natural.
- ***Capítulo 3. Metodología***. A lo largo del tercer capítulo se introducen los modelos a utilizar para la consecución de los objetivos planteados así como las bases de datos que se han utilizado para el entrenamiento de estos modelos. Además, se presentan las distintas herramientas que se han utilizado a lo largo de todo el proceso.
- ***Capítulo 4. Preparación y comprensión de los datos utilizados***. En este apartado se explicarán los distintos procesos que se han llevado a cabo para preparar los datos y que puedan ser consumidos por los modelos elegidos en ambas tareas.
- ***Capítulo 5. Resultados y discusión***. En el capítulo quinto se presentarán los resultados obtenidos por los modelos tras realizar distintas pruebas y experimentos. Se compararán las métricas obtenidas en cada uno de ellos tanto a nivel de calidad de dato como computacional y se discutirá sobre ellos.
- ***Capítulo 6. Conclusiones y trabajos futuros***. Durante este capítulo se hará una recapitulación de todos los elementos estudiados y obtenidos durante la realización de este proyecto con el objetivo de extraer unas conclusiones claras al respecto. También se hablará sobre líneas de trabajo abiertas que se quedan al finalizar este trabajo. Debido a que es un trabajo centrado en dar unos primeros pasos en Inteligencia Artificial, se planteará hacia dónde se quiere seguir dirigiendo esta primera toma de contacto dentro de la empresa colaboradora.

CAPÍTULO 2
Estado del arte

2.1 Antecedentes

2.1.1. Métodos y técnicas en la generación automática de resúmenes

Desde sus inicios, la generación de resúmenes se ha abordado utilizando principalmente métodos extractivos, que seleccionan las partes más relevantes del texto original, y abstractivos, que generan nuevas frases que condensan el contenido original. La elección del enfoque depende en gran medida de los requisitos específicos de la aplicación, con los métodos extractivos siendo más directos y los abstrativos ofreciendo mayor flexibilidad y concisión.

Enfoque extractivo

El enfoque extractivo alberga los métodos más tradicionales dentro de la generación automática de resúmenes. Dichos métodos consisten en seleccionar fragmentos del texto a sintetizar, ya sean oraciones o párrafos más extensos, que se consideran representativos del texto total y ordernarlos creando una coherencia capaz de resumir el contenido del texto en un párrafo breve. Algunas de las técnicas más comunes de las técnicas extractivas son las siguientes:

- TF-IDF (*Term Frequency-Inverse Document Frequency*): esta técnica trata de convertir un documento de texto en un formato estructurado. Esta representación se hace a través de un valor numérico que refleja la importancia de una palabra dentro de un documento, pues su valor aumenta proporcionalmente al número de veces que aparece el término en el texto. TF-IDF es el resultado de combinar dos medidas clave:
  - *Term Frequency* (TF): esta primera medida es la encargada de calcular la frecuencia con la que un término se repite en un documento en relación con el total de palabras que lo conforman. De esta forma, aquellas palabras que tengan una mayor frecuencia de aparición tendrán un mayor valor de TF. Este valor se calcula de la siguiente forma:

$$\mathrm{TF}(t,d) = \frac{f_{t,d}}{\sum_{t' \in d} f_{t',d}} \tag{2.1}$$

donde:

- $f_{t,d}$ es la frecuencia del término t en el documento d.
- $\sum_{t' \in d} f_{t',d}$ es la suma de todas las frecuencias de los términos en el documento d.

- *Inverse Document Frequency* (IDF): esta segunda medida se ocupa de determinar cómo de común es una palabra dentro de un corpus de documentos. La fórmula del **IDF** se define como:

$$\text{IDF}(t) = \log \frac{N}{n_t} \tag{2.2}$$

donde:

- N es el número total de documentos en el corpus.
- n_t es el número de documentos que contienen el término t.

Finalmente, la fórmula que combina ambas medidas para obtener el valor TF-IDF final es la siguiente:

$$\text{TF-IDF}(t,d) = \text{TF}(t,d) \times \text{IDF}(t) \tag{2.3}$$

donde:

- $\text{TF}(t,d)$ es la frecuencia del término t en el documento d.
- $\text{IDF}(t)$ es la inversa de la frecuencia de documentos que contienen el término t, definida como $\text{IDF}(t) = \log \frac{N}{n_t}$.

Esta técnica es ampliamente utilizada dentro de las tareas de recuperación de la información así como de la minería de texto [34] [35]. Asimismo, también puede ser una buena técnica dentro de la generación de resúmenes. Es este caso, su funcionamiento consiste en asignar una puntuación a las palabras dentro de un documento según su relevancia y ordenarlas para construir resúmenes basándose en sus frecuencias relativas [3].

- *TextRank* [24]: este método es un derivado del algoritmo *PageRank* [5] utilizado por el motor de búsqueda Google. El algoritmo de *PageRank* trata de medir la importancia de las página web basándose en la probabilidad de que un usuario visite una página, basándose en la presencia de enlaces entre dos páginas. El algoritmo *TextRank* tiene una lógica similar, ya que se extrae a partir de este. La diferencia está en que este algoritmo sustituye las páginas web por oraciones. Su funcionamiento consiste en aplicar un modelo de grafo diseñado para identificar aquellas oraciones de más importancia dentro del texto a resumir [22] [46]. Dentro de los métodos para la extracción de palabras clave, *TextRank* se considera uno de los más esenciales debido a varios motivos. En primer lugar, este algoritmo utiliza las relaciones del vocabulario para ordenar las palabras clave basándose en la extracción de estas palabras del texto y poder así reflejar mejor la información semántica entre las unidades del texto. Además, puede encontrar aquellos elementos más informativos separándolos en distintas unidades y construyendo un grafo de texto entre ellas. Por último, su aplicación a problemas reales [4] [12] [39] y el hecho de no necesitar ningún tipo de entrenamiento previo, lo hace un algoritmo sencillo y de alta usabilidad [48].

- Modelos basados en grafos: este tipo de modelos tienen como objetivo representar el texto como si de un grafo se tratase [25]. La definición de un grafo podría resumirse como una estructura matemática que consta de dos elementos, los nodos (también denominados vértices) y las aristas (enlaces) que conectan los nodos entre sí. Dentro del contexto de la generación de resúmenes, los nodos simulan las unidades del texto a tratar y las aristas hacen referencia a las similitudes y relaciones semánticas. Estos métodos han demostrado ser efectivos en la identificación de las ideas principales de un texto donde la precisión es clave, como por ejemplo en resumen de documentos técnicos o legales, así como en la detección de noticias falsas [26]. Además, este tipo de modelos sigue expandiéndose y trata de buscar nuevos enfoques. Los más avanzados se dirigen hacia el uso de grafos semánticos donde las aristas también representan relaciones semánticas como sinonimia o relaciones antológicas [17]. Por otro lado, los nodos pasan en este caso a representar conceptos clave o entidades mencionadas en el texto. Gracias a este nuevo enfoque es posible crear resúmenes centrados alrededor de los temas principales que se tratan en el texto.

Enfoque abstractivo

Los métodos abstractivos tratan de generar resúmenes obtenidos tras reescribir o parafrasear el contenido original del texto. Este enfoque presenta una mayor complejidad que los métodos recién explicados ya que requiere una comprensión profunda del contexto y la semántica del texto. Afortunadamente, hoy en día existen técnicas que permiten realizar este tipo de tareas a pesar de su dificultad. Entre las más destacadas se encuentran las siguientes:

- Redes neuronales recurrentes [8]: este tipo de redes neuronales, más conocidas como RNN por sus siglas en inglés (*Recurrent Neural Networks*), fue uno de los primeros modelos neuronales utilizados para la generación automática de resúmenes. Este tipo de redes así como sus variantes, como LSTM (*Long Short-Term Memory*) [37] y GRU (*Gated Recurrent Units*) [2] han sido especialmente útiles a la hora de manejar secuencias de texto gracias a su capacidad de capturar dependencias a largo plazo. Un ejemplo de este tipo de modelos sería el conocido Seq2Seq (*Sequence-to-Sequence*) [40], donde se utiliza una RNN para la codificación del texto y otra para la decodificación en forma de resumen. A pesar de su efectividad, estos modelos presentan limitaciones como el conocido desvanecimiento del gradiente. Este factor dificulta el aprendizaje de dependecias a largo plazo. Además, las redes recurrentes tienen un procesamiento lento, lo que supone un problema en la escalabilidad para la generación de resúmenes en textos más extensos.

- Transformers y modelos de atención [42]: la introducción de esta nueva arquitectura ha supuesto un punto de inflexión dentro de las tareas de generación automática de resúmenes. Una gran ventaja de estos modelos es el procesamiento paralelizado de todas las palabras en un documento, lo que implica mayor eficiencia y capacidad de manejar las relaciones a largo plazo en los documentos. Este factor se debe en gran parte a los mecanismos de atención que se utilizan en

esta arquitectura, ya que permite al modelo enfocarse en distintas partes del texto de acuerdo a su relevancia para la tarea. Debido a su complejidad y extensión, esta arquitectura se explicará a continuación en un apartado propio.

- Modelos preentrenados: estos modelos han supuesto un gran avance dentro del procesamiento del lenguaje natural ya que permite transferir el conocimiento entre tareas y dominios. Específicamente dentro de la generación de resúmenes se ha conseguido mejorar de forma significativa la precisión y coherecia de los resúmenes generados. El enfoque de estos modelos implica entrenar un modelo con una gran cantidad de datos y posteriormente realizar *fine-tuning* para tareas específicas, lo que permite esa mejora en la calidad del resultado obtenido. Dentro de estos modelos destacan algunos como *BERT* [13], diseñado originalmente para tareas de clasificación pero adaptado para tareas de generación de resúmenes, *GPT-3* [6], siendo un modelo generativo con un alto rendimiento en la síntesis de texto, y *T5* [32], un modelo creado por Google capaz de adaptarse a distintas tareas.

A pesar de la gran potencialidad de este tipo de modelos, el enfoque abstractivo también presenta una serie de retos todavía por resolver. Dentro de estos desafíos se encuentra la garantía de la coherencia y precisión del resumen generado, pues no dejan de ser modelos artificiales con margen a cometer errores. Además, al igual que un ser humano, estos modelos también pueden contener sesgos heredados de los datos con los que se entrenan, siendo un factor delicado en un ámbito como el de las noticias donde la neutralidad y la objetividad deberían ser elementos esenciales. Por último, estos modelos suelen funcionar bien en un dominio específico, mientras que su rendimiento puede verse influido negativamente al afrontar textos de otra índole. A pesar de estas desventajas, estos modelos siguen siendo un área de investigación muy activa sobre los que se sigue estudiando para resolver todos estos inconvenientes.

2.1.2. Procesamiento de lenguaje natural

El Procesamiento del Lenguaje Natural (NLP), o más conocido como NLP por sus siglas en inglés (*Natural Language Processing*), es el campo computacional que trata de analizar y representar el texto basándose en un conjunto de teorías y tecnologías de forma natural a distintos niveles de análisis lingüístico. El objetivo de este área es conseguir un procesamiento del lenguaje lo más similar posible a un humano para una serie de tareas y aplicaciones [18]. Este sector de la inteligencia artificial surge con el propósito de facilitar el trabajo humano y la comunicación con un ordenador mediante el uso del lenguaje natural. Las aplicaciones de NLP son bastante amplias, abarcando desde la traducción automática y la generación de resúmenes hasta la comprensión de sentimientos en textos y la interacción con asistentes virtuales. El desarrollo de NLP ha estado marcado por varias etapas tecnológicas, comenzando con reglas y modelos estadísticos, siguiendo hacia métodos basados en aprendizaje automático hasta llegar a la adaptación de modelos de aprendizaje profundo, que han transformado la capacidad de las máquinas para comprender y tratar el lenguaje humano.

Los primeros intentos de abordar problemas de procesado de lenguaje natural se basaban en reglas manuales y técnicas estadísticas. Estos métodos utilizaban gramá-

ticas, diccionarios y reglas sintácticas para analizar el lenguaje. Sin embargo, su complejidad y falta de flexibilidad limitaban su capacidad para manejar la variabilidad y ambigüedad del lenguaje natural. Una de las primeras técnicas estadísticas ampliamente utilizadas fue el modelo *N-grama* [38], que predice la probabilidad de una palabra dada su contexto de las *n-1* palabras anteriores. Aunque esta técnica no ha dejado de resultar útil, este enfoque sufre de problemas como la explosión combinatoria y la falta de consideración de la estructura semántica.

Otros de los modelos que más destacaron dentro de esta etapa tecnológica son los Modelos de Markov [30], principalmente utilizados para tareas como el etiquetado de las partes del discurso (POS tagging [23]) y la desambiguación de palabras. Estos modelos funcionan tratando el texto como una secuencia de estados, donde la probabilidad de transición entre estos se basa en datos históricos.

Posteriormente, estos modelos fueron evolucionando hasta llegar a una nueva etapa marcada por el aprendizaje automático. Debido al aumento de la capacidad de cómputo y la disponibilidad de grandes volúmenes de datos, la llegada de estas nuevas técnicas irrumpieron con fuerza en el campo de NLP. Este tipo de modelos consiguen aprender directamente de los datos, un hecho que provocó una mejora significativa en la capacidad de tratar el lenguaje natural. Algunas de las técnicas dominantes durante esta etapa son *Bag of Words* (BoW) [29], TF-IDF [1] o *Word Embedding* [21]. La primera de ellas es una de las pioneras, por lo que su sencillez es lo que más destaca, pues consiste en representar un documento como un conjunto de palabras sin tener en cuenta el order entre ellas. A pesar de que esta simplicidad podría ser beneficiosa, también tiene un punto simplista que se hace visible mediante la pérdida de relación semántica entre las palabras. Una técnica que mejora BoW es la recién presentada en el anterior capítulo TF-IDF, que consigue mejorar su rendimiento al ponderar los términos en función de la relevancia que estos tengan en el documento en relación con un corpus específico. De esta forma se consigue mejorar la capacidad del modelo para que pueda conseguir identificar términos relevantes. Por último, uno de los métodos más avanzados del aprendizaje automático para NLP son los *Word Embeddings*. Lo que hacen este tipo de métodos es representar el documento en un espacio vectorial, lo que captura la semántica de las palabras en vectores densos de baja dimensión permitiendo una mejor comprensión las relaciones de este tipo.

Por último, se encuentran los modelos basados en aprendizaje profundo, dominados principalmente por la presencia de redes neuronales. Esta técnica ha causado una revolución en distintos ámbitos, entre ellos NLP, permitiendo la construcción de modelos que mejoran el rendimiento de los enfoques utilizados previamente en diversas tareas. Algunas de las aplicaciones clave de esta disciplina son la traducción automática, que ha evolucionado desde el uso de reglas y frases predefinidas hacia métodos basados en modelos estadísticos y, finalmente, a redes neuronales y transformadores; el análisis de sentimientos, capaz de extraer opiniones, emociones y sentimientos utilizando *embeddings* y Transformers en modelos de clasificación; y el desarrollo de *chatbots* y asistentes virtuales, como Siri y Alexa, que emplean técnicas avanzadas de aprendizaje profundo para entender y generar lenguaje humano en respuesta a consultas de los usuarios.

A pesar de todos los avances que ha habido en el campo del procesamiento del lenguaje natural, sigue habiendo algunos puntos a mejorar. Uno de ellos es el relacionado con el sentido común, la ambigüedad y la generación de un texto coherente

capaz de seguir una contextualización en tareas complejas. Las mejoras que ha habido en este aspecto son significativas, pero aún queda recorrido en estos aspectos para conseguir comprender y generar de forma más similar el lenguaje humano desde una perspectiva computacional, es decir, seguir dotando a las máquinas de habilidades principalmente humanas. Además, otro factor que sigue presente como una barrera a traspasar es el sesgo, ya que este tipo de modelos tienden a reflejar y amplificar los sesgos presentes en los datos se entrenamiento. En numerososo casoso se ha visto cómo este detalle puede conducir a resultados subjetivos o discriminatorios en aplicaciones críticas. En cuanto a una perspectiva de interpretabilidad, cabe destacar que un modelo más complejo implica menor interpretabilidad, plantendo desafíos de transparencia y explicabilidad de los resultados obtenidos por los modelos de aprendizaje profundo.

2.1.3. Transformers

Los Transformers [42] son una arquitectura de red neuronal introducida en 2017 que ha conseguido superar a los modelos anteriores basados en redes neuronales tanto recurrentes (RNN) [8] como convolucionales (CNN) [16], principalmente en el ámbito de NLP. Esto se debe en gran parte a su capacidad para manejar de manera eficiente secuencias de texto de longitud variable y detectar dependencias entre estas a largo plazo. Desde su introducción en el mundo científico, los Transformers se han convertido en la base de los modelos de lenguaje más avanzados, como BERT [13], GPT [6] y T5 [33], impulsando avances significativos en tareas como traducción automática, generación de texto, y clasificación de texto. Esta arquitectura se basa en un mecanismo de atención que permite a los modelos enfocarse en diferentes partes de la secuencia de entrada al procesar cada palabra, lo que mejora la captura de relaciones contextuales entre palabras, independientemente de la distancia entre ellas.

A continuación se detalla de forma más precisa los distintos elementos que conforman el total de esta arquitectura, la cual puede verse ilustrada en la Figura 2.1:

- Mecanismo de atención. El mecanismo de atención es la parte central de toda la arquitectura Transformer. El funcionamiento de esta pieza se basa en la asignación de pesos a cada palabra en la secuencia de entrada según la importancia que tenga esta para la predicción del término actual. Esta acción es posible gracias a tres conceptos clave:
  - *Query (Q)*: la *query* o consulta hace referencia a la palabra del texto que se está procesando en un instante i. En el contexto de la atención, cada palabra genera una *query* que se utiliza para buscar relaciones con otros elementos de la secuencia.
  - *Key (K)*: la clave o *key* es una representación asociada con cada elemento en la secuencia que actúa como un identificador. Las consultas se comparan con las claves para establecer la importancia de un elemento con respecto a otro, es decir, la clave es lo que utiliza el modelo para decidir si un valor es relevante para la consulta que está buscando.
  - *Value (V)*: el valor o *value* es la parte que contiene la información a utilizar en caso de que la consulta encuentre una clave relevante. Tras la comparación de una consulta con la clave y el cálculo de su importancia mediante un

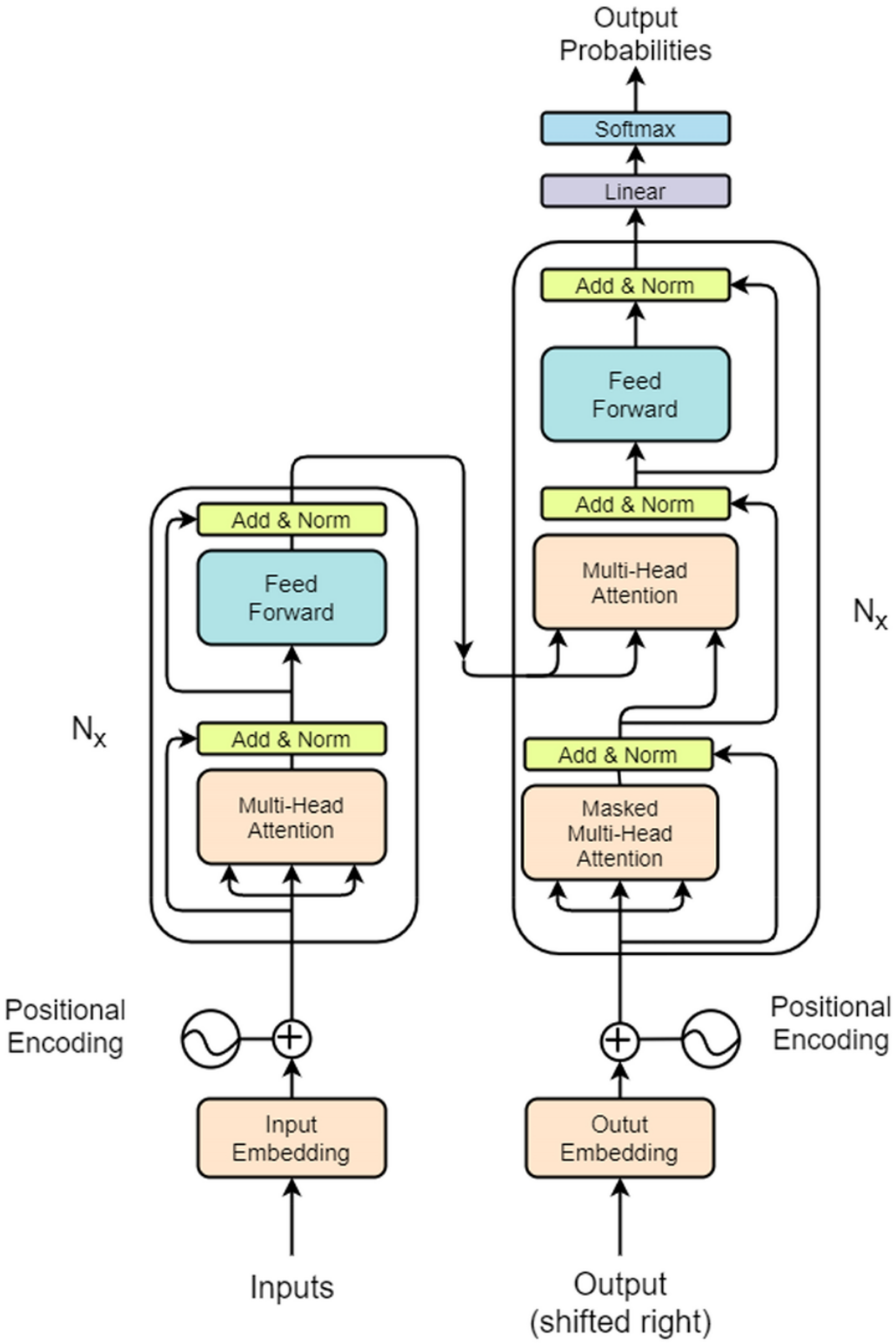

Figura 2.1: Arquitectura del modelo Transformer [42].

producto escarlar normalizado, el valor correspondiente a la llave es lo que realmente se extrae y se combina para formar la salida final del mecanismo de atención.

La fórmula que se utiliza para calcular la atención es la siguiente:

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^\top}{\sqrt{d_k}}\right)\mathbf{V} \tag{2.4}$$

donde:

- **Q** (query): hace referencia a la matriz de consultas, donde cada fila corresponde a una consulta generada a partir de los elementos de la secuencia de entrada.
- **K** (key): representa la matriz de claves, donde cada fila corresponde a una clave asociada a cada uno de los elementos de la secuencia de entrada.
- **V** (value): se interpreta como la matriz de valores, donde cada fila contiene la información que se utilizará si la clave correspondiente es relevante para la consulta.
- $\mathbf{K}^\top$: es la traspuesta de la matriz de claves **K**, lo que permite calcular el producto escalar entre la consulta y todas las claves.
- d_k: representa la dimensionalidad de los vectores de claves (y consultas). Se utiliza para escalar el producto escalar, reduciendo así el riesgo de que los valores de la softmax sean demasiado pequeños o grandes, lo que podría hacer que el modelo aprenda más lentamente.

- *Multi-head attention*. Una de las aportaciones más novedosas que aporta la arquitectura Transformer es la introducción de la atención multi-cabeza. Este mecanismo permite que el modelo sea capaz de enfocarse en paralelo en diferentes partes de la secuencia de entrada desde distintas perspectivas. Cada cabeza se encarga de procesar la secuencia de una manera distinta, lo que permite determinar relaciones más variadas y complejas dentro del texto.

- *Encoder-Decoder*. La arquitectura Transformer puede dividirse en dos bloques principales, que son el codificador (*encoder*) y el decodificador (*decoder*). La parte del *encoder* es la encargada de tomar la secuencia de entrada y transformarla en una representación interna. Este bloque se compone de varias capas de autoatención y capas *feed-forward* que permiten al modelo capturar dependencias a largo plazo y representar la secuencia de forma contextualizada. Por otra parte, el decodificador utiliza la representación que ha generado este bloque para producir la secuencia de salida. Este *decoder* también está formado por capas de autoatención y atención cruzada, permitiendo al modelo generar secuencias de salida condicionadas por la entrada.

- *Positional encoding*. Puesto que los Transformers no se caracterizan por poseer una estructura secuencial como sí lo tienen las redes neuronales recurrentes, esta arquitectura utiliza un mecanismo denominado *positional encoding*. Esta técnica se crea con el objetivo de introducir información relativa al orden de las palabras dentro de la secuencia mediante el uso de un vector de posición a los embeddings de las palabras antes de que estas pasen al modelo.

Dado el gran impacto que tuvo la publicación de este artículo ([42]) en el mundo de la inteligencia artificial, muchas empresas comenzaron a desarrollar modelos basados en la arquitectura del Transformer. En primer lugar, está BERT (*Bidirectional Encoder Representations from Transformers*) [14] y todas sus variantes que, como su nombre bien indica, es un modelo de Transformer bidireccional que ha logrado muy buenos resultados dentro de una serie de tareas de distinta naturaleza dentre del área de NLP. El hecho de que sea bidireccional implica que se considera el contexto de una palabra tanto desde la izquierda como desde la derecha. Este factor es el responsable de una mejora considerable en la comprensión del texto, ya que el contexto que se abarca es mucho más amplio y menos restrictivo. Este modelo ha sido entrenado con una gran cantidad de datos pertenecientes a dos tareas de lenguaje enmascarado y predicción de la siguiente frase. Tras este preentrenamiento inicial, BERT se ajusta a tareas específicas con un conjunto de datos de menor tamaño. Esto es lo que porteriormente permite su adaptación a varios tipos de aplicaciones. Por otro lado, otro de los grandes modelos que han surgido a partir de los Transformers es GPT (*Generative Pretrained Transformer*) [31], que ha sido desarrollado por la empresa OpenAI[1] y ampliamente conocido por su uso en la aplicación *Chat-GPT*. A diferencia de BERT, este modelo es unidireccional y está enfocado principalmente a la generación automática de texto. La generación de este texto se realiza palabra por palabra y utiliza únicamente el contexto anterior de la secuencia. Una de las características de este modelo que marca un avance significativo en la generación de texto es la innecesidad de realizar *fine-tuning* al modelo para que realice las tareas de NLP que se desean. Otro de los grandes modelos es el T5 (*Text-To-Text Transfer Transformer*) [32], desarrollado por Google, que trata todas las tareas de procesamiento del lenguaje como un problema de trasnformación de texto a texto, es decir, cualquier tarea bien sea de traducción, resumen o respuestas, se convierte en la transformación de un texto de entrada en un texto de salida. Esta simplificación deriva en un entrenamiento mucho más sencilla y permite que el modelo pueda adaptarse fácilmente a distintas tareas que se propongan.

Todos estos avances han dado un giro a la forma en la que se abordan las distintas tareas de NLP pues, como se ha visto, un mismo modelo puede utilizarse en distintas tareas. Fuera de la parte de utilidad, a nivel de calidad también han conseguido mejorar el nivel de precisión, fluidez y coherencia de los métodos más tradicionales y de otros más novedosos como los basados en RNN. Pero al igual que en las técnicas más sofisticadas, estas mejoras implican un coste asociado y en este caso supone un alto coste computacional y la necesidad de cantidades enormes de datos para poder entrenar estos modelos. Al igual que en los modelos expuestos anteriormente, los Transformers y derivados siguen presentando problemas de sesgo heredado de los datos y de interpretabilidad de los resultados o procesos que estos realicen internamente. Ya que es una de las áreas más recientes y más novedosas, también es de las que recibe más atención e investigación a día de hoy, por lo que investigaciones futuras están focalizadas en resolver estos inconvenientes para poder ofrecer unos modelos multimodales, robustos y eficientes para poder ser utilizados a nivel mundial en tiempo real.

[1] https://openai.com/.

2.2 Estado del arte

La generación automática de resúmenes y la extracción de ideas principales son áreas críticas dentro del procesamiento del lenguaje natural que han experimentado avances significativos en las últimas décadas. Con la creciente disponibilidad de grandes volúmenes de datos textuales, estas tecnologías se han vuelto esenciales para facilitar la comprensión y el análisis rápido de la información. En este apartado, se revisan algunos de los estudios más relevantes y recientes en ambos campos, destacando las tendencias, enfoques metodológicos y aplicaciones prácticas.

2.2.1. Generación automática de resúmenes

La generación automática de resúmenes tiene como objetivo producir versiones condensadas de documentos largos, preservando las ideas más relevantes y la coherencia del texto original. Los enfoques en este campo se pueden dividir en extractivos, que seleccionan partes del texto original, y abstractivos, que generan nuevas frases basadas en el contenido. Un ejemplo representativo de los enfoques extractivos es el modelo SummaRuNNer, propuesto por Nallapati et al. (2016) [27]. Este modelo utiliza redes neuronales recurrentes que capturan la relación entre las oraciones y la importancia de las mismas dentro del documento. Esta red se encarga de evaluar la relevancia de cada oración en función de factores como la posición de la oración, su contenido informativo y su relación con otras oraciones. Este modelo logró resultados notables en conjuntos de datos como CNN/Daily Mail, mostrando mejoras en la métrica ROUGE respecto a métodos tradicionales basados en grafos y estadísticas.

El uso de modelos de lenguaje preentrenados ha llevado a avances significativos en la generación extractiva de resúmenes. Liu y Lapata (2019) [20] introdujeron BERTSUM, una extensión del modelo BERT que adapta la arquitectura de los Transformers para tareas de resumen. BERTSUM implementa una estructura jerárquica que permite capturar dependencias largas y relaciones entre oraciones dentro de un documento, lo que resulta en una clara mejora en la selección de oraciones clave. Este enfoque ha establecido nuevos estándares en varias métricas de evaluación, destacándose en su capacidad para manejar textos más largos y complejos de manera efectiva.

En un contexto global, la capacidad de generar resúmenes en múltiples idiomas es crucial. Hasan et al. (2021) [11] exploraron modelos basados en Transformers capaces de realizar tareas de resumen en diferentes lenguas sin requerir entrenamiento específico para cada idioma. Su enfoque utiliza un modelo multilingüe preentrenado que se ajusta para tareas de resumen mediante una *fine-tuning* multilingüe, logrando buenos resultados en varios idiomas a la vez.

Dentro de un enfoque más abstractivo, también ha habido importantes estudios centrados en modelos basados principalmente en Transformers. See et al. (2017) [36] introdujeron un modelo de resumen abstractivo que se basa en la arquitectura Seq2Seq (*Sequence-to-Sequence*) con un mecanismo de atención. Este modelo también incorpora un componente de copia, que permite al generador copiar palabras directamente del texto original cuando sea necesario con el objetivo de mejorar la fiabilidad del resumen. Este trabajo ha sido influyente, sirviendo como base para muchos estudios posteriores que buscan mejorar tanto la fluidez como la precisión de los resúmenes ge-

nerados. La introducción de Transformers ha marcado un hito en la generación de resúmenes abstractivos. Basados en esta arquitectura, se han desarrollado modelos como T5 (Text-To-Text Transfer Transformer), propuesto por Raffel et al. (2020) [33]. T5 reformula todas las tareas de NLP, incluida la generación de resúmenes, como un problema de traducción de texto a texto, logrando resultados de vanguardia en múltiples tareas de resumen. El resumen multilingüe también ha sido explorado en el contexto de métodos abstractivos. Ladhak et al. (2020) [15] propusieron un modelo que extiende las capacidades del Transformer para manejar múltiples idiomas, generando resúmenes abstractivos en diferentes lenguas. Este enfoque aprovecha la capacidad del modelo para aprender representaciones compartidas entre idiomas, lo que permite generar resúmenes coherentes y precisos sin necesidad de traducciones intermedias. Otro avance interesante es la capacidad de controlar la longitud y el estilo del resumen generado. Fan et al. (2018) [9] introdujeron métodos para ajustar el resumen a una longitud deseada sin perder coherencia, lo cual es crucial para aplicaciones prácticas donde las restricciones de espacio son importantes. Esto se logra mediante la manipulación de los mecanismos de atención y generación en el modelo Seq2Seq.

2.2.2. Extracción de las ideas principales

La extracción de ideas principales se centra en identificar los conceptos clave de un texto, lo que es fundamental para tareas como la indexación de información, la generación de etiquetas automáticas, y la mejora de la búsqueda de documentos.

Dentro de este campo, la evolución de las redes neuronales ha permitido mejoras significativas en la precisión de estos sistema. Cheng y Lapata (2016) [7] propusieron un enfoque innovador utilizando RNNs para evaluar la importancia de cada oración en un documento. Su modelo analiza tanto el contenido informativo de cada oración como su relación con otras oraciones en el texto. Esto permite una mejor captura de la estructura discursiva del documento, lo que resulta en una extracción más precisa de las ideas principales. Para mejorar la captura de la estructura del documento, Yang et al. (2019) [45] desarrollaron un modelo de atención jerárquica que considera tanto el contexto de la oración en sí como el contexto global, es decir, el documento completo. Este modelo jerárquico mejora la precisión en la extracción de ideas principales al combinar diferentes niveles de información contextual. Al igual que en la generación de resúmenes, la extracción de ideas principales también explora el campo de los Transformers. Xie et al. (2019) [44] introdujeron un enfoque que utiliza BERT para extraer características profundas de los textos, lo que mejora la identificación de ideas clave. BERT, al ser un modelo bidireccional, permite una comprensión más rica del contexto, lo que es particularmente útil para la extracción precisa de conceptos en documentos complejos.

Además de los avances basados en redes neuronales, los enfoques que combinan técnicas clásicas con nuevas metodologías han mostrado resultados prometedores. Los métodos basados en grafos, como el algoritmo TextRank introducido por Mihalcea y Tarau (2004) [24], siguen siendo populares debido a su simplicidad y eficacia. TextRank construye un grafo de las oraciones o palabras del documento, donde los nodos más conectados representan las ideas principales. Aunque estos métodos son menos sofisticados que los modelos neuronales, todavía se utilizan ampliamente, especialmente en sistemas donde la simplicidad y la velocidad son prioridades. Recientemen-

te, se han propuesto varias extensiones a TextRank para mejorar su precisión. Ganesan (2018) [10], por ejemplo, introdujo una variante que incorpora un análisis de dependencia sintáctica para mejorar la relevancia de las oraciones seleccionadas. Estas mejoras permiten a los algoritmos clásicos competir con técnicas más modernas en ciertas aplicaciones.

La extracción de ideas principales en textos largos y complejos sigue siendo un desafío debido a la dificultad de mantener la coherencia y la relevancia en documentos extensos. Zhong et al. (2020) [49] propusieron un enfoque que primero segmenta el texto en partes más manejables antes de aplicar técnicas de extracción. Este método permite a los modelos enfocarse en secciones más pequeñas del texto, lo que mejora la precisión en la identificación de las ideas principales sin perder la estructura general del documento. La adaptabilidad de los modelos es otra área de investigación activa. Modelos que pueden ajustarse dinámicamente a la longitud del texto y a la complejidad del contenido, como los propuestos por Narayan et al. (2021) [28], representan un avance significativo en la extracción de ideas principales en textos complejos. Estos modelos utilizan mecanismos de atención adaptativa que ajustan el nivel de detalle del análisis según la longitud y la estructura del documento.

CAPÍTULO 3
Metodología

En este capítulo se explicarán los distintos conjuntos de datos que se han utilizado para la realización de este trabajo así como su implementación y el software requerido para su ejecución.

3.1 Descripción del conjunto de datos

Los datos que se han utilizado en este trabajo son de naturaleza puramente textual. Dado que se trata de una generación de resúmenes y extracción de ideas principales sobre noticias web, los datos que se han utilizado para el entrenamiento también son relativos a artículos de la misma índole que aquellos que posteriormente se utilizarán durante la puesta en producción de la herramienta. Para ello, se ha tratado con tres conjuntos de datos distintos.

En primer lugar, la primera base de datos pertenece a noticias filtradas por la empresa. En el día a día de ForwardKeys se utiliza Feedly[1] como agregador de noticias para estar al día de las novedades que ocurren en el sector que le incumbe. Por ello, en esta herramienta se pueden encontrar distintos filtros que agrupan las noticias según la naturaleza de estas noticias. Dado que se desea utilizar los modelos generados en este trabajo para el resumen de estos artículos, se ha considerado como primera base de datos la recopilación de todos aquellos textos que están almacenados en este tablón de filtros. Estas noticias, como bien se ha comentado, se clasifican según unos filtros como la región del mundo a la que haga referencia, el año de publicación del texto y el tipo de clientes con el que podría estar relacionada la noticia. Todas estas se han obtenido mediante las aplicaciones integradas en Feedly que permiten descargarlas en Dropbox[2] para tener un histórico de aquellas noticias que han sido relevantes en algún momento. Este conjunto cuenta en total con 933 noticias. Esta base de datos es de especial importancia, pues es el conjunto en el que se basa principalmente el trabajo. Las noticias que aparecen en este dataset son aquellas que la empresa en algún momento ha considerado relevantes y que han servido de ayuda para tomar decisiones de carácter estratégico y estar alineados con las tendencias y actualidades más recientes del sector de las aerolíneas.

[1] https://feedly.com/.
[2] https://www.dropbox.com/.

Este primer conjunto de datos no está etiquetado con su correspondiente resumen e idea principal, por lo que ha sido necesario buscar otras bases de datos que contengan esta información para el posterior entrenamiento de los modelos. Por ello se han extraído del repositorio público Kaggle dos nuevas bases de datos. La primera de ellas es una recopilación de noticias publicadas en internet por la BBC, contando en total con 2225 textos de noticias con sus correspondientes resúmenes. Estas noticias también están clasificadas según su temática, pues pueden albergar contenido sobre negocio, entretenimiento, política, deportes y tecnología. Este conjunto de datos contiene un fichero de texto con la noticia en sí y otro con el resumen de este. Este conjunto se utilizará únicamente para el objetivo de la generación de resúmenes dado que no dispone de la información necesaria para la realización de la otra tarea. Por úlitmo, la segunda base de datos, extraída del mismo repositorio, es una colección de noticias extraídas de distintas fuentes. Esta base de datos contiene aproximadamente 98500 noticias, de las cuales se conoce el autor, la fecha de publicación, la idea principal, el enlace donde poder leer la noticia, el resumen y el texto completo. Esta base de datos se utilizará para las dos tareas de este trabajo junto con las bases de datos recién explicadas.

3.2 Técnicas de modelado de datos

Actualmente, existe una amplia variedad de modelos a utilizar para el entrenamiento de modelos que sean capaces de generar resúmenes y de extraer las ideas clave. Dado que hay muchos avances en este ámbito, finalmente se ha optado por escoger modelos preentrenados pertenecientes al enfoque abstractivo, ya que es el que recoge el mayor número de novedades y cuenta con técnicas más actuales y sofisticadas. Aún habiendo reducido el número de posibilidades, dentro de este sector también se pueden encontrar diversos modelos como Pegasus [47], Gemma [41], GTP [31], etc. Puesto que la idea de este trabajo pretende encontrar el mejor modelo posible, se ha optado por escoger cuatro modelos preentrenados, que se emplearán tanto para la tarea de resúmenes como para la parte de extracción de ideas. El motivo de utilizar los mismos modelos para ambas tareas reside en tratar de reutilizar el mismo modelo para llevar a cabo las dos tareas, intentando optimizar lo máximo posible el rendimiento de la empresa. Estos modelos se han escogido a partes, es decir, uno de estos modelos será más generalista (habrá sido entrenado con todo tipo de textos desde artículos breves hasta grandes documentos) mientras que el segundo únicamente habrá sido preentrenado con textos extraídos de noticias. Los modelos finales escogidos son las dos versiones de Pegasus y otras de T5.

- Pegasus: los modelos preentrenados de la familia Pegasus están basados en la arquitectura de Transformers, que ha demostrado ser muy eficaz en tareas de NLP. Pegasus, en particular, está diseñado específicamente para tareas de resumen de textos, lo que lo hace muy adecuado para este propósito. En primer lugar, se ha utilizado el modelo *pegasus-x-base*, una versión general de Pegasus entrenado en una amplia gama de tareas de resumen y comprensión de texto. Este modelo está diseñado para generar resúmenes concisos y coherentes de una gran variedad de tipos de documentos. Por otro lado, como segundo modelo más enfocado a la generación de resúmenes de noticias se ha escogido *pegasus-xsum*. Esta variante ha sido entrenado con el conjunto de datos XSum, que contiene noticias

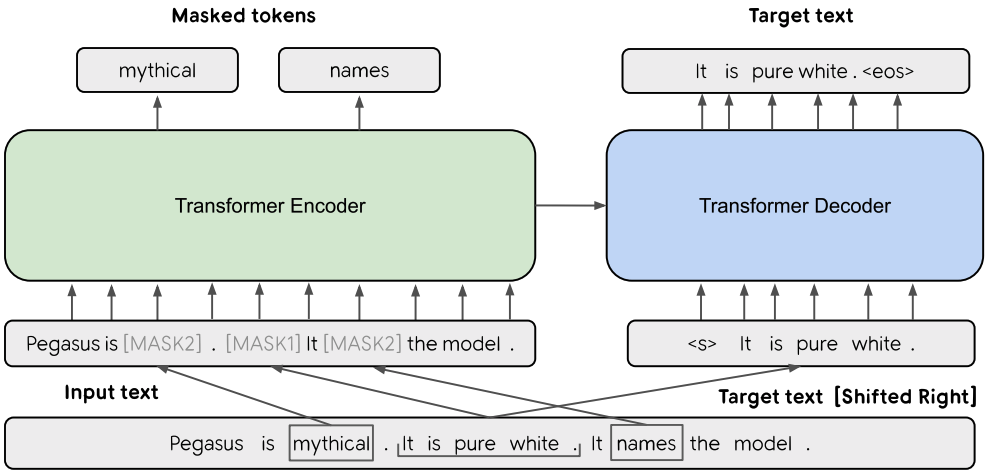

Figura 3.1: Arquitectura básica del modelo Pegasus [47].

y artículos. En esta base de datos los resúmenes tienden a ser más informativos y concisos, siendo conocido por ser un conjunto de datos muy desafiante debido a su enfoque en la generación de resúmenes de una sola frase. *pegasus-xsum* se especializa en resúmenes abstractivos de alta calidad, convirtiéndolo en una buena opción para casos en los que se requiere condensar grandes volúmenes de información en unas pocas líneas, tal y como se espera en este trabajo. Pegasus ha sido entrenado en una gran cantidad de datos que incluyen múltiples dominios, lo que le permite generar resúmenes de alta calidad en diversos contextos. El entrenamiento a gran escala permite que el modelo entienda mejor la estructura y el contenido relevante de los textos, ayudando a producir resúmenes más precisos y relevantes. *pegasus-xsum* es particularmente efectivo en el dominio de las noticias, ya que el modelo está ajustado para generar resúmenes concisos y claros de artículos periodísticos. Además, ambos modelos han demostrado un rendimiento competitivo en benchmarks de resúmenes, como el conjunto de datos CNN/DailyMail y XSum. Esto significa que en evaluaciones automáticas de calidad de resúmenes, superan a otros modelos en términos de métricas de precisión como ROUGE, lo que los hace confiables para tareas de resumen en el mundo real. Finalmente, estos modelos son optimizados para ser eficientes, permitiendo generar resúmenes de manera rápida sin requerir cantidades excesivas de recursos computacionales. Esto es crucial para aplicaciones que requieren generación de resúmenes a gran escala o en tiempo real.

- T5: el funcionamiento de estos modelos trata de manejar todas las tareas de NLP como tareas de traducción de texto a texto. Esto significa que el modelo es extremadamente versátil y puede lidiar con tareas de resumen, traducción, clasificación, entre otras, dentro de un mismo marco. Para generar resúmenes, simplemente se reformula la tarea como la conversión de un documento largo en una versión más corta, lo que se ajusta adecuadamente al enfoque de este modelo. T5 está basado en la arquitectura Transformer, que ha revolucionado el campo del NLP debido a su capacidad para manejar dependencias a largo plazo en secuencias de texto, superando las limitaciones de los modelos recurrentes como LSTM

y GRU. La arquitectura de este modelo es un Transformer estándar con dos componentes principales como lo son el codificador y el decodificador. El primer componente procesa la secuencia de entrada generando una representación contextualizada para cada token mientras que el decodificador genera la secuencia de salida según las representaciones anteriores junto con los tokens generados hasta el momento. Una parte esencial de T5 es la atención multi-cabeza, ya que es este ingrediente el que permite que el modelo se enfoque a la vez en distintas partes de la secuencia de entrada. En cada capa del Transformer se utilizan varias cabezas de atención, donde cada una aprende a focalizarse en diferentes relaciones dentro del texto. Este mecanismo es el que permite capturar las dependencias y mejorar la comprensión. Otra capa utilizada por los modelos T5 es la capa de normalización y conexiones residuales en cada capa de la red. Durante el preentrenamiento, T5 emplea una técnica de enmascarado similar a BERT, donde se ocultan ciertos tokens de la entrada y el modelo aprende a predecir los tokens faltantes. Sin embargo, T5 enmascara secuencias enteras de tokens, en lugar de solo palabras individuales. Esto fomenta que el modelo aprenda a generar frases coherentes, lo que es crucial para tareas como la generación de texto y el resumen. T5 fue entrenado y evaluado en varias configuraciones de tamaño, desde *T5-Small* hasta *T5-XXL*. Las diferentes versiones del modelo varían en el número de parámetros, capas, cabezas de atención y tamaño de la representación interna, permitiendo a los usuarios equilibrar el rendimiento del modelo con los requisitos de recursos computacionales. Su preentrenamiento se realizó en el Colossal Clean Crawled Corpus (C4) [32], un conjunto de datos masivo derivado de páginas web. Este corpus fue limpiado rigurosamente para eliminar contenido de baja calidad. Durante el preentrenamiento, T5 aprendió a completar textos enmascarados, desarrollando una comprensión profunda del lenguaje a partir de una amplia variedad de textos. Además, el modelo puede ser ajustado finamente en conjuntos de datos específicos de resúmenes (como CNN/DailyMail o XSum), lo que le habilita para especializarse en generar resúmenes de alta calidad para tipos específicos de contenido. Esto mejora su precisión y relevancia en la extracción de ideas principales. En cuanto a rendimiento, ha demostrado ser altamente competitivo en benchmarks de resumen como CNN/DailyMail, superando a muchos otros modelos. Sus capacidades avanzadas de modelado del lenguaje permiten producir resúmenes concisos y coherentes que capturan las ideas clave del texto original.

3.3 Evaluación

Para la evaluación de los modelos, se ha empleado la métrica ROUGE (Recall-Oriented Understudy for Gisting Evaluation) [19], ampliamente utilizada dentre del campo del NLP para la evaluación de la generación automática de texto. Esta métrica se basa en la comparación entre un resumen generado y otro de referencia que generalmente ha sido creado por una persona humana. ROUGE trata de medir la superposición de unidades de texto como n-gramas, palabras o cadenas de caracteres entre el resumen generado y las etiquetas reales. Este valor ha sido fundamental en la evolución y evaluación de la calidad de los resúmenes , principalmente en aquellas tareas donde se busca una evlauación cuantitativa y automatizada. En cuanto a

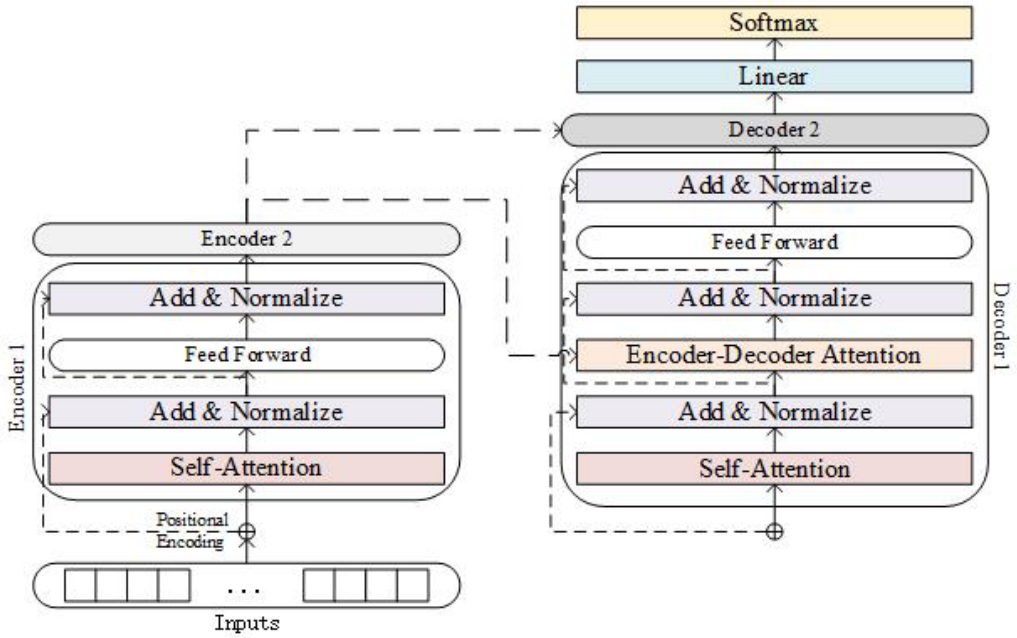

Figura 3.2: Arquitectura general del modelo preentrenado T5 [43].

su funcionamiento, ROUGE calcula diferentes variantes de precisión, recall y F1 score, dependiendo del tipo de coincidencia que se evalúe entre el resumen generado y su correspondiente referencia.

- ROUGE-1 mide la coincidencia de unigramas entre el resumen generado y su etiqueta. Esta métrica es la más básica de todas, pues tiene por objetivo reflejar qué tan bien el resumen captura las palabras más importantes del texto original. Más concretamente, lo que hace es contar las palabras coincidentes entre el resulmen generado y las referencias y a parir de este ratio calcula la precisión, el *recall* y el F1. La interpretación de esta variante puede ser algo confusa, ya que un valor alto de ROUGE-1 sugiere que la generación ha coseguido capturar muchas de las palabras pero no necesariamente la estructura o el orden de las ideas.

- ROUGE-2 evalúa la coincidencia de bigramas entre los pares de resúmenes. Evalúa la preservación de pares de palabras vecinas, por lo que puede ser un mejor indicador de coherencia y relación entre términos. Su funcionamiento es igual que en la otra variante de la métrica, pero en este caso compara bigramas en vez de palabras únicas. Por este mismo motivo, la interpretación de ROUGE-2 resulta más coherente, ya que un valor alto implica tanto la buena inclusión de palabras clave como el mantenimiento de las relaciones contextuales.

- ROUGE-L va más allá y se enfoca en la coincidencia de la subsecuencia común más larga entre los resúmenes generados y los referentes. Este enfoque permite mayor flexibilidad que la métrica anterior, ya que no requiere que las palabras coincidan de forma consecutiva, sino que considera como mayor subsecuencia aquella en la que las palabras aparecen en el mismo orden. En este caso, la métrica se calcula identificando la subsecuencia común más larga entre ambos textos y se calculan la precisión, el recall y el F1 basándose únicamente en la longitud de esta subsecuencia. En valor alto de ROUGE-L refleja que el resumen consigue

mantener la extructura general del texto original, capturando las ideas principales en un orden similar el del texto fuente. Esta variante tiene a su vez otra variante denominada ROUGE-LSum, que está adaptada específicamente para tareas de resumen de documentos más extensos. En lugar de calcular la subsecuencia coincidente más larga de forma simple, lo que hace es considerar la estructura del documento y cómo las frases clave se distribuyen a lo largo del resumen. Esto permite capturar la estructura y el flujo del documento original.

Además de la métrica ROUGE y sus variantes, también se han calculado los intervalos de confianza calculados para cada una de estas métricas. Estos valores representan las métricas de evaluación que se espera que contengan el valor real de la métrica en un cierto porcentaje de las veces. Generalmente, este porcentaje está estipulado al 95 %. Estos intervalos se expresan a través de tres valores:

- *Low*: este valor corresponde al límite inferior del intervalo de confianza para la métrica específica. Indica el valor más bajo que podría alcanzar la métrica según el intervalo de confianza calculado. Es decir, representa una estimación conservadora del rendimiento del modelo.
- *Mid*: este es el valor promedio de la métrica, por eso mismo suele ser el que se toma como representativo del rendimiento real del modelo, siendo consecuentemente el que más se usa para la comparación de modelos.
- *High*: este valor corresponde al límite superior. Indica el valor más alto que podría alcanzar la métrica según el intervalo calculado. Representa una estimación optimista del rendimiento del modelo.

A pesar de que pueda parecer redundante contemplar todas las variantes junto con sus intervalos de confiaza respectivos, todos estos valores son cruciales para entender la fiabilidad de los resultados. Por la parte de las variantes ROUGE es importante valorar los distintos solapamientos que hay entre los resúmenes para poder hacer una comparación más amplia y concienciada a la hora de seleccionar el modelo que mejor se ajuste a la tarea abordada. Por otro lado, los valores del intervalo de confianza son esenciales para poder entender la fiabilidad de los resultados obtenidos. Si el intervalo entre el valor más bajo y el más alto es estrecho significa que hay una alta certeza en la estimación de la métrica. En caso contrario, se estaría ante un caso donde hay mayor incertidumbre en el rendimiento real del modelo.

3.4 Software y librerías

Los modelos entrenados en este trabajo han sido ejecutados en la plataforma Amazon SageMaker[3] utilizando una instancia de tipo *g4dn xlarge* con 4 vCPU y 16 GiB, siendo este el entorno de ejecución asociado al departamento de Ciencia de Datos de ForwardKeys. Además, los datos han sido almacenados en un *bucket* de S3 en AWS[4]. Para la implementación del código ha sido necesario el uso de librerías como *pandas*,

[3]https://aws.amazon.com/es/sagemaker/
[4]https://aws.amazon.com/es/

datasets o *transformers*. A continuación se proporciona el listado de librerías esenciales para el entrenamiento de los modelos utilizados:

- *s3fs*: esta biblioteca de Python permite interactuar con el sistema de almacenamiento de objetos de Amazon S3 de manera similar a como se trabaja con un sistema de archivos local. Gestiona las interacciones con S3 de forma sencilla sin la necesidad de credenciales gracias a los permisos otorgados a cada usuario, facilitando tareas como la lectura, escritura y eliminación de archivos. Proporciona una interfaz basada en el estándar de Python, lo que permite realizar operaciones con rutas de S3 usando funciones familiares como open, read, y write. Esto simplifica el manejo de datos en la nube desde entornos de Python y ha sido fundamental para la carga de las bases de datos.

- *pandas*: esta biblioteca combinada con *s3fs* en combinación permite leer y escribir fácilmente datos en Amazon S3. Al integrarse con *s3fs*, *pandas* puede acceder a los archivos almacenados en S3 directamente mediante rutas en formato *s3://*, lo que ha facilitado el trabajo al utilizar datos en la nube sin necesidad de descargarlos localmente. Esto ha permitido trabajar con datos almacenados en S3 en tiempo real.

- *Transformers*: esta biblioteca de HuggingFace[5] ha sido una herramienta necesaria para trabajar con los modelos de procesamiento de lenguaje natural basados en la arquitectura de Transformers que se han utilizado a lo largo del desarrollo del trabajo. Facilita la carga, entrenamiento y uso de modelos preentrenados para diversas tareas, ya sea generación de resúmenes y extracción de ideas principales como otras relativas a la clasificación o traducción automática. Es compatible con frameworks como PyTorch y TensorFlow, y ofrece una API sencilla que permite aplicar modelos de última generación de manera eficiente y rápida.

- *Datasets*: para poder preprocesar y utilizar los conjuntos de datos en este trabajo ha sido necesaria la importación de esta biblioteca, ya que soporta datos en múltiples formatos y ofrece una interfaz simple para cargar, manipular y compartir datasets, optimizando su uso tanto en memoria como en disco. La combinación de esta biblioteca con la anterior ha permitido el desarrollo de los modelos mencionados anteriormente.

- *PyTorch*: PyTorch es una biblioteca popular de aprendizaje automático desarrollada por Facebook AI Research. Ofrece un entorno flexible y eficiente y destaca por su diseño basado en grafos dinámicos. Esto facilita la depuración y la investigación, y es ampliamente utilizado para el desarrollo de aplicaciones de inteligencia artificial como la propuesta en este proyecto.

- *nltk*: esta biblioteca de Python está diseñada para trabajar con datos de lenguaje natural. Proporciona herramientas para el procesamiento de texto, incluyendo tokenización, etiquetado de partes del discurso, análisis sintáctico, lematización y más. Se ha utilizado en este caso para el preprocesado de los datos y su análisis.

[5] https://huggingface.co/

- *rouge-score*: ROUGE es la métrica utilizada para evaluar la calidad de los resúmenes generados por los modelos preentreados comparándolos con los resúmenes de validación. La biblioteca *rouge-score* en Python implementa esta métrica, permitiendo calcular las variantes de esta métrica. Es ampliamente utilizada en tareas de generación de texto, como resúmenes y traducción automática.

3.5 Implementación y diseño de experimentos

Como se ha comentado en el apartado anterior, toda la implementación de código utilizado para la ejecución de este trabajo se ha realizado en Python.

En primer lugar, se ha realizado la selección de los modelos para realizar el entrenamiento. En un primer planteamiento inicial, se evaluó la idea de utilizar dos modelos distintos según la tarea a aplicar. Esto quiere decir que se pretendía utilizar un par de modelos (general y entrenado con textos) para la tarea de generación de resúmenes y otro par para la tarea de extracción de las ideas principales. El objetivo de esta separación residía principalmente en escoger modelos específicos de cada tarea con la intención de conseguir el mejor objetivo posible. Sin embargo, a la hora de seleccionar los modelos para la segunda parte del trabajo, se llegó a la conclusión de que las ideas clave no dejan de ser un resumen más resumido y conciso, por lo que un modelo de generación de resúmenes sería igual de válido para la tarea. Por ello, finalmente se ha optado por utilizar dos pares de modelos a ambas tareas. El beneficio de este úlimo enfoque es múltiple. Por una parte, permite comparar el rendimiento de los modelos según la tarea que se está llevando a cabo. Mientras, por otro lado también puede valorarse la opción de utilizar un único modelo para ambas tareas, mejorando la eficiencia de la empresa al tener que utilizar un único modelo. Finalmente los modelos escogidos son Pegasus, con *pegasus-x-base* y *pegasus-xsum* y T5, con *t5-small* y *t5-small-xsum-en*.

Una vez decididos los modelos finales a utilizar, se pasa a la definición de los experimentos a realizar durante el trabajo. Dado que hay 2 tareas distintas y 4 modelos en total, hay 8 experimentos básicos donde se ha probado el modelo preentrenado en crudo para ver cómo funciona con los datos de testeo disponibles para evaluación. Después, también se han ajustado estos modelos mediante *fine-tuning* utilizando los datos de entrenamiento y validación que se han extraído de las bases de datos externas explicadas anteriormente. Por último, también se ha planteado otra serie de experimentos donde únicamente se utiliza para el entrenamiento de los modelos aquellas noticias más similares a las proporcionadas por ForwardKeys.

Visualmente, podría decirse que el procedimiento a seguir es el que se enmarca a través de las combinaciones planteadas en la Figura 3.3:

La implementación final del trabajo consiste en probar modelos según si es un modelo preentrenado con texto general o restringido al ámbito de las noticias, si los textos utilizados son de diversas temáticas o únicamente relacionadas con las de la empresa y si el modelo se entrena con los datos o no. Tras la combinación de todas estas opciones para los ocho modelos distintos, se han obtenido los resultados que se expondrán y analizarán en capítulos posteriores.

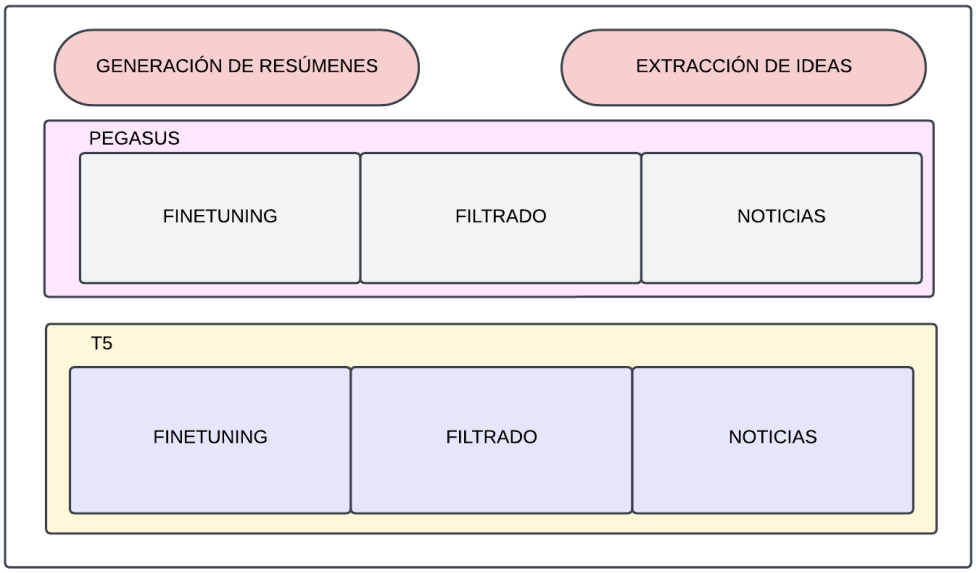

Figura 3.3: Esquema de los experimentos realizados según su uso de fine-tuning, la base de datos con filtrado o no y la versión del modelo preentrenado.

3.6 Marco legal

En el desarrollo del presente trabajo, cuyo objetivo es la generación de resúmenes automáticos y la extracción de ideas clave a partir de textos de noticias web utilizando modelos de lenguaje preentrenados como Pegasus y T5, se han considerado cuidadosamente los aspectos legales, éticos y contractuales implicados. La selección de estos modelos se basa en su capacidad avanzada para procesar lenguaje natural, así como en el hecho de que ambos han sido preentrenados en grandes corpus de datos compuestos exclusivamente por textos de dominio público. Esta elección garantiza que el uso de estos modelos no infringe derechos de autor, ya que los datos utilizados para su preentrenamiento no están sujetos a restricciones de propiedad intelectual. Este enfoque es fundamental para cumplir con las normativas internacionales sobre derechos de autor, que protegen las obras literarias, incluyendo textos de noticias, de su uso no autorizado.

Además, los datos empleados en las fases de validación y testeo de este proyecto han sido proporcionados por la empresa para la cual se desarrolla este trabajo. Estos datos han sido manejados en estricto cumplimiento de los acuerdos de confidencialidad y las obligaciones contractuales que rigen el acceso y uso de la información corporativa, protegiendo así la propiedad intelectual que entra en juego a la hora de realizar el filtrado para la selección de las noticias implicadas.

El uso de modelos preentrenados como Pegasus y T5 plantea además consideraciones éticas, particularmente en relación con el posible sesgo inherente en los modelos. Ambos modelos han sido preentrenados en datos públicos, lo que implica que podrían reflejar sesgos presentes en dichos datos.

El marco legal que rige este proyecto no se limita únicamente al cumplimiento de las leyes de derechos de autor y protección de datos, sino que también abarca la res-

ponsabilidad ética en la creación y uso de tecnología de procesamiento del lenguaje natural. Los modelos T5 y Pegasus, aunque poderosos, requieren un manejo responsable para asegurar que su implementación en tareas sensibles, como el resumen de noticias, no comprometa la integridad de la información ni la privacidad de los individuos.

Además, cualquier resultado de este trabajo que sea divulgado fuera del ámbito de la empresa o en publicaciones científicas deberá ser revisado cuidadosamente para asegurar que no se infrinjan derechos de terceros. Se ha puesto especial énfasis en la correcta atribución de las fuentes utilizadas, siguiendo las normativas de citación académica y profesional, lo que garantiza la transparencia y el respeto a los derechos de los creadores originales de contenido.

En conclusión, este trabajo se ha desarrollado dentro de un marco legal y ético riguroso, trantando de asegurar en todo momento el cumplimiento de las normativas de propiedad intelectual, protección de datos y confidencialidad. La elección de modelos preentrenados como Pegasus y T5, junto con un enfoque proactivo en la mitigación de sesgos y el manejo responsable de los datos, refuerzan el compromiso del proyecto con las mejores prácticas en el desarrollo de tecnologías de procesamiento del lenguaje natural. Este enfoque no solo protege los derechos e intereses de todas las partes involucradas, sino que también garantiza la calidad, precisión y ética en los resultados obtenidos.

CAPÍTULO 4
Preparación de los datos

En este cuarto capítulo se hablará sobre el procesado que se ha aplicado a los datos en los distintos entrenamientos así como el filtrado aplicado para ver qué datos utilizar durante el entrenamiento, validación y testeo.

4.1 Preprocesado

Como se venía comentando en la sección anterior, para la realización de este trabajo se han utilizado tres bases de datos. La primera de ellas se ha recopilado a partir de un tablón de filtros de Feedly. La vida de este conjunto de datos comienza en el momento en el que la noticia en cuestión es seleccionada como texto de interés, pues el contenido de los artículos presenta gran relevancia para la temática de ForwardKeys. Una vez se reconoce la noticia como útil, esta se almacena en un tablón, el cual a su vez se divide en distintas carpetas según la categoría a la que pertenezca. Estas categorías se crean según el objetivo para el que sea interesante la noticia, bien sean clientes, mercados específicos, años en concreto, etc. Todo este filtrado es propiedad intelectual de la compañía, ya que son sus trabajadores quienes se encargan de esta tarea. Una vez los artículos están en su directorio correspondiente, se procede con un tratamiento más informático. Dado que estos artículos están en una página en la nube, es necesario descargarlos de alguna manera. En este caso en concreto, se ha apostado por las aplicaciones integradas dentro de la página web, ya que permitían guardar una copia de seguridad de estos artículos en formato PDF. Tras la disposición de los textos en local, ha sido necesario almacenar todas aquellas noticias en una base de datos conjunta indistintamente de la categoría a la que pertenezcan. Después de haber pasado todos los artículos a un formato adecuado para su tratamiento con Python, se ha procedido a la parte de etiquetado. Dado que para el entrenamiento es necesario tanto el dato como su etiqueta, en este caso el texto y el resumen/la etiqueta, y únicamente se disponía de la primera parte, ha sido necesario el etiquetado manual de una parte de todos los textos descargados por parte de la empresa. Para agilizar este proceso y hacerlo más automático, se ha hecho uso de alguna herramienta en línea de generación de resúmenes, aunque posteriormente se han revisado todas las etiquetas para garantizar una coherencia tanto con la noticia como textual. El etiquetado de las noticias únicamente se ha llevado a cabo para los conjuntos de validación y test. Por ello, antes de nada se han dividido de forma aleatoria las noticias en estos dos conjuntos y otro para en-

trenamiento. Esta separación ha tratado de mantener noticias de cada subcategoría en las distintas particiones, teniendo finalmente 53 noticias en el conjunto de validación, 53 en el conjunto de test y las 827 restantes para el entrenamiento y filtrado, que se detallará en las siguientes líneas.

Una vez lista la primera base de datos, se ha procedido con los conjuntos externos. Como ya se ha mencionado, estos conjuntos de datos son de acceso público y han sido muchas veces el conjunto de datos a utilizar en competiciones de Kaggle, por lo que no ha sido necesario un preprocesado muy exhaustivo. Las tareas que se han hecho principalmente han sido la selección de aquellas características imprescindibles para la tarea que incumbe en este caso y la unión de ambas bases de datos para las distintas tareas. Por tanto, finalmente se ha contado con una gran base de datos para la parte de los resúmenes y otra para la extracción de ideas. Estas noticias tienen en común con la base de datos inicial que son de una misma naturaleza de artículo informativo, teniendo una extensión y una estructura similar que comparten todos los textos de este tipo. Sin embargo, el contenido de estos artículos puede ser muy variado, por lo que se ha decidido hacer una segunda base de datos a partir de esta que contenga únicamente aquellas noticias que tengan un índice de similitud similar al de las noticias proporcionadas por ForwardKeys que se han particionado anteriormente en el conjunto de entrenamiento, el cual no contiene etiquetas. La idea principal de este filtrado de noticias viene motivada por la creencia de que es posible que entrenar un modelo con noticias del mismo estilo de redacción y pertenecientes al mismo ámbito o similares pueda ofrecer un mejor rendimiento y por tanto mejores resúmenes de mayor calidad. Este índice de similitud se calcula a partir de la distancia coseno, que se rige por la siguiente fórmula:

$$\text{Similitud coseno} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} \tag{4.1}$$

Esta distancia es una medida que evalúa cómo de similares son dos vectores en un espacio multidimensional, por lo que está muy relacionada con el análisis de texto y su representación vectorial. El valor de esta métrica se calcula como el coseno del ángulo entre estos vectores. Esta métrica puede ir desde -1 hasta 1. Dos vectores con similitud coseno 1 quiere decir que estos son idénticos o que apuntan en la misma dirección, es decir, tienen máxima similitud. Si la similitud es -1, entonces ambos vectores son opuestos, o dicho en otras palabras, tienen máxima disimilitud. Por último, aquellos que tengan un valor cercano a cero implica que no tienen nada de similitud.

Para la representación vectorial, se han planteado distintas opciones de cómo proceder. Esta parte del cálculo de similitud entre documentos tiene múltiples opciones más clásicas explicadas anteriormente como TF-IDF o BoW o técnicas más modernas como las que se aplican hoy en día en aprendizaje profundo.

Puesto que en todo el trabajo se ha apostado por inteligencia artificial y técnicas más avanzadas así como uso de modelos preentrenados y de lenguaje, finalmente se ha decidido utilizar un modelo preentrenado de BART para tokenizar y representar vectorialmente los documentos. Además de los motivos recién mencionados, también se cree que, al ser una técnica más reciente y sofisticada, los resultados obtenidos a partir de este modelo serán más fieles a la realidad que las otras técnicas que, a pesar de ser clásicas y muy utilizadas incluso en la actualidad, pueden carecer de sentido contextual y estar más deprecadas.

Por último, tras haber tokenizado todos los textos y calculado la similitud coseno, se ha escogido únicamente aquellas noticias que tuvieran un índice de similitud entre la base de datos externa y la propia mayor a 0.8 para formar una segunda base de datos con noticias filtradas. El resultado de este preproceso son cuatro bases de datos: dos conjuntos generales con todas las noticias recopiladas -una para generación de resúmenes y otra para extracción de ideas principales- y otras dos con aquellas noticias más similares a las recopiladas desde ForwardKeys.

Finalmente, las bases de datos utilizadas según su preprocesado son:

- Forwardkeys: primera base de datos, particionada en entrenamiento (827 noticias), validación (53 noticias) y test (53 noticias).
- BBC y News summary sin filtrar: primera base de datos externa para la tarea de generación de resúmenes automático. Inicialmente cuenta con más de 73000 noticias, pero por motivos de capacidad de cómputo finalmente se ha reducido a una muestra de 14000 artículos.
- BBC y News summary filtrado: segunda base de datos externa para la generación de resúmenes, obtenida a partir del conjunto de datos anterior mediante el filtrado con BART. En este caso se cuentan con 11786 noticias.
- News summary plus sin filtrar: primera base de datos externa para la extracción de ideas principales. Inicialmente cuenta con más de 50000 noticias, pero por motivos de capacidad de cómputo finalmente se ha reducido a una muestra de 14000 artículos, mismo tamaño que la base de datos de la tarea anterior.
- News summary plus filtrado: segunda base de datos externa para la extracción de ideas principales, obtenida a partir del conjunto de datos anterior mediante el filtrado con BART. Este último conjunto se compone de 10493 textos.

CAPÍTULO 5
Resultados y discusión

En este capítulo se presentan los resultados obtenidos tras la implementación de los experimentos previamente diseñados. Se analizarán los resultados cuantitativos obtenidos mediante las métricas ROUGE y se discutirán las diferencias entre los modelos T5 y Pegasus, tanto en su versión básica como con finetuning y el filtrado de noticias. El objetivo es identificar cuál de estos enfoques consigue un mejor rendimiento en la tarea de generación automática de resúmenes y la extracción de ideas principales, para concluir con una discusión sobre el modelo más apropiado en función del contexto y la tarea específica.

5.1 Generación automática de resúmenes

En esta primera tarea, el objetivo se centra en generar resúmenes automáticos a partir de noticias. Para ello, se han empeado distintos modelos preentrenados: por un lado versiones generales de T5 y Pegasus y, por otra parte, variantes que han sido ajustadas mediante finetuning con los datos, tanto del conjunto de datos completo como el relativo al de noticias filtradas únicamente. Se ha evaludao el rendimiento de estos modelos usando las métricas ROUGE-1, ROUGE-2, ROUGE-L y ROUGE-Lsum, las cuales se centran en la precisión de las palabras, frases y la coherencia estructural entre los textos generados y los de referencia.

En primer lugar, los modelos T5 preentrenados en crudo presentan unos resultados relativamente bajos en las métricas evaluadas, especialmente en las primeras ejecuciones sin ajuste fino (ver Figura 5.1). El modelo básico de *t5-small*, por ejemplo, alcanza apenas un 0.004 en F-measure en la métrica ROUGE-1 y unos valores todavía más bajos en ROUGE-2, siendo estos aproximadamente de 0.002. Estos resultados indican una falta de coincidencia entre las palabras clave generadas por el modelo y las del resumen de referencia, lo que se puede traducir como un resumen poco apropiado para cumplir el objetivo de la tarea.

Sin embargo, cuando se realiza finetuning sobre estos modelos, los resultados mejoran considerablemente, como puede observarse en el incremento de F-measure en *t5-small-xsum* tras el ajuste fino, alcanzando un valor cercano a 0,008 en ROUGE-1. Esto sugiere que, aunque T5 puede funcionar de forma limitada en tareas generales de generación de texto, el entrenamiento especializado y la adaptación a un dominio

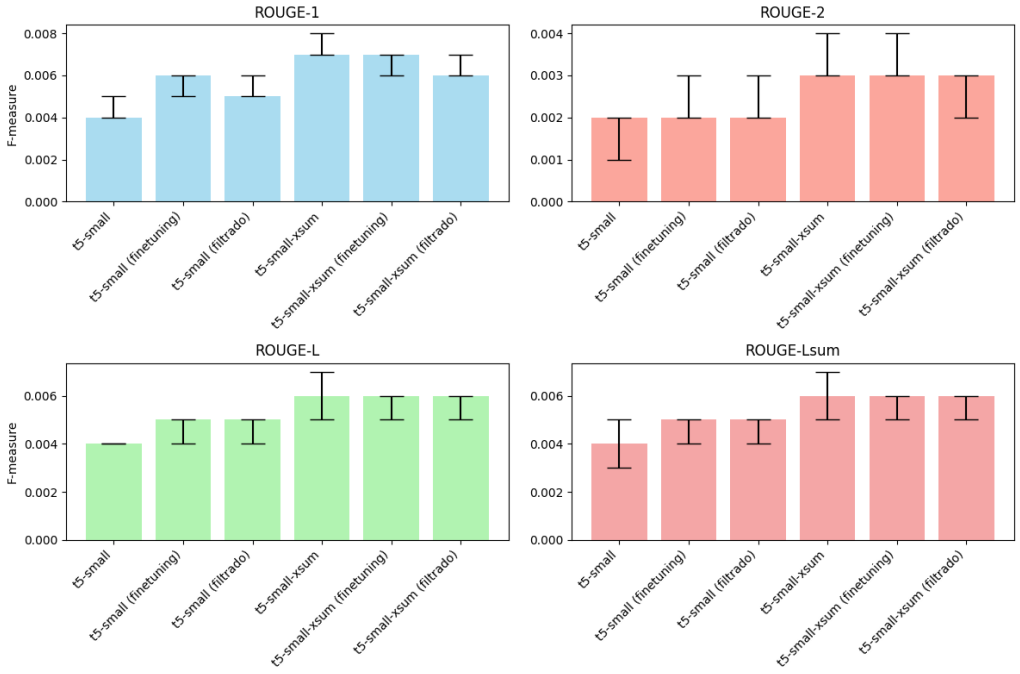

Figura 5.1: Comparación entre los distintos modelos de T5 para resúmenes según F-measure.

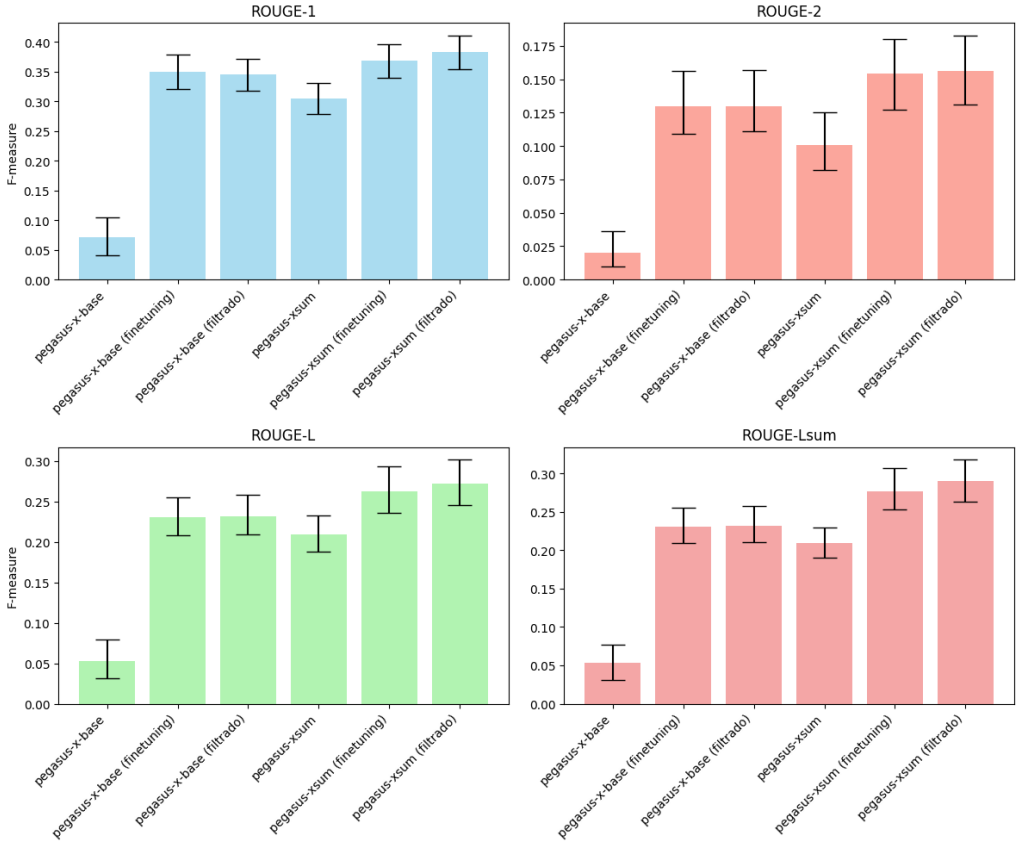

Figura 5.2: Comparación entre los distintos modelos de Pegasus para resúmenes según F-measure.

concreto, en este caso el de las noticias, mejoran la capacidad del modelo para generar resúmenes más precisos y relevantes.

En contraste, los modelos de la familia Pegasus presentan un rendimiento superior desde el inicio (ver Figura 5.2). El modelo *pegasus-x-base* alcanza valores más altos que *t5-small*, incluso sin realizar *finetuning*, superando los valores de 0,25 en ROUGE-1 y alcanzando alrededor de 0,10 en ROUGE-2, lo que indica una mayor capacidad para generar palabras y bigramas relevantes en comparación con T5. Este rendimiento puede deberse a que los modelos de la familia Pegasus fueron diseñados específicamente para tareas de resumen de texto, mientras que T5 es un modelo más generalista que no ha sido entrenado para abordar ninguna tarea en concreto.

Al realizar el finetuning, Pegasus mejora todavía más, alcanzando casi 0,40 en ROUGE-1 y superando 0,15 en ROUGE-2, lo que demuestra que los modelos especializados pueden alcanzar una precisión y cobertura significativamente superiores tras ser ajustados con datos específicos del dominio de aplicación.

Además del ajuste que se realiza a los modelos preentrenados, se han evaluado los modelos aplicando un filtrado a los conjuntos de noticias basado en la similitud del contenido con respecto a noticias del interés de la empresa. Los modelos entrenados con los conjuntos filtrados muestran una ligera mejora en las métricas, aunque estas diferencias no son tan pronunciadas como las observadas tras el finetuning. Por ejemplo, tanto en los modelos *t5-small-xsum* como en *pegasus-xsum*, el filtrado ofrece una mejora en ROUGE-Lsum y ROUGE-2, pero estas diferencias son marginales, lo que sugiere que el filtrado de noticias no es el factor determinante en el rendimiento del modelo, sino más bien el proceso de ajuste fino. Esto puede deberse al hecho de que las noticias, independientemente del contenido, guardan una similidud entre ellas, pues enseñan al modelo a resumir noticias web.

En ambos casos se ve claramente cómo los resultados indican que los modelos preentrenados en crudo tienen dificultades para realizar la tarea de generación de resúmenes. El *finetuning* ofrece tanto en los modelos T5 como Pegasus un aumento sustancial en la *precision* y el *recall* de los modelos, lo que se refleja en las métricas ROUGE. El conjunto filtrado mejora ligeramente las métricas, pero no de manera significativa. Algo que sí presenta también una mejora ciertamente significativa es el uso de modelos especializados y preentrenados con textos de noticias, pues estos demuestran ser más efectivos en la tarea de generación automática de resúmenes de artículos de características estilométricas similares.

Además de los gráficos presentados previamente, los resultados cuantitativos obtenidos a partir de las métricas ROUGE para los modelos T5 y Pegasus se presentan de manera más detallada en las Tablas 5.1 y 5.2. Estas tablas muestran de forma clara cómo varía el rendimiento de los modelos en función de si se ha aplicado finetuning y del tipo de conjunto de datos empleado, ya sea este general o filtrado. En primer lugar, para el modelo T5, los resultados revelan una diferencia notable entre los modelos sin finetuning y aquellos que han sido ajustados. Por ejemplo, en la Tabla 5.1, los valores de precisión y F-measure para ROUGE-1 en el modelo sin finetuning (0,778 de precisión) son más bajos en comparación con el mismo modelo ajustado con finetuning (0,786 de precisión), especialmente cuando se utiliza el conjunto filtrado de noticias. Esto indica que el ajuste fino permite al modelo mejorar en tareas de generación de resúmenes al captar mejor la información relevante del texto.

Por otro lado, en el caso del modelo Pegasus, los resultados presentados en la Tabla 5.2 refuerzan esta tendencia. Los modelos ajustados mediante *finetuning* y aquellos

Tabla 5.1: Resultados de la evaluación de modelos T5 para generación de resúmenes.

Modelo	Finetuning	Datos	Metric	Precision	Recall	F-measure
t5-small	No	Completo	ROUGE-1	0,778	0,002	0,004
			ROUGE-2	0,274	0,001	0,002
			ROUGE-L	0,699	0,002	0,004
			ROUGE-Lsum	0,699	0,002	0,004
	Sí	Completo	ROUGE-1	0,778	0,003	0,006
			ROUGE-2	0,319	0,001	0,002
			ROUGE-L	0,705	0,003	0,005
			ROUGE-Lsum	0,705	0,003	0,005
t5-small-xsum	No	Completo	ROUGE-1	0,833	0,004	0,007
			ROUGE-2	0,397	0,002	0,003
			ROUGE-L	0,738	0,003	0,006
			ROUGE-Lsum	0,738	0,003	0,006
	Sí	Completo	ROUGE-1	0,781	0,003	0,007
			ROUGE-2	0,382	0,002	0,003
			ROUGE-L	0,683	0,003	0,006
			ROUGE-Lsum	0,682	0,003	0,006
	Sí	Filtrado	ROUGE-1	0,776	0,003	0,006
			ROUGE-2	0,373	0,001	0,003
			ROUGE-L	0,678	0,003	0,006
			ROUGE-Lsum	0,678	0,003	0,006

entrenados con noticias filtradas muestran un mejor rendimiento, tanto en precisión como en F-measure. Por ejemplo, el modelo *pegasus-xsum* sin *finetuning* consigue una F-measure de 0,305 en ROUGE-1, mientras que al aplicar ajuste fino esta métrica sube a 0,368, y cuando además se entrena con el conjunto de datos filtrados, la F-measure alcanza un valor de 0,383. Estos resultados demuestran que no solo el *finetuning* es crucial para mejorar el rendimiento en la generación de resúmenes, sino que el uso de datos filtrados también añade una capa adicional de mejora, aunque esta última sea menos significativa en algunos casos.

En cuanto a las métricas ROUGE-2 y ROUGE-L, se observa una tendencia similar en ambas familias de modelos. Los modelos ajustados con *finetuning* muestran un mejor desempeño en todas las métricas, lo que resalta la importancia de este proceso para la generación de secuencias más complejas y coherentes. En particular, en la métrica ROUGE-L, que evalúa la coherencia estructural del texto, el modelo *pegasus-xsum* con ajuste fino y entrenado con datos filtrados consigue una F-measure de 0,290, significativamente mejor que el modelo sin ajuste. Este resultado es clave para asegurar que los resúmenes generados no solo sean precisos, sino que mantengan la estructura general del texto, lo que es esencial en tareas que requieren mantener una narrativa coherente.

Mirando estos resultados desde un punto de vista más cualitativo, se ve también cómo en los mismos resúmenes generados por los modelos las diferencias son claras. Un ejemplo de resumen generado por el modelo *pegasus-xsum* con *finetuning* es el siguiente:

Emirates Airline has announced a partnership with the Bahamas government to promote tourism in the country and boost visitor traffic into the islands from across the airline's network, in a bid to boost visitor traffic into the islands from across the network.

Tabla 5.2: Resultados de la evaluación de modelos Pegasus para generación de resúmenes.

Modelo	Finetuning	Datos	Metric	Precision	Recall	F-measure
pegasus-x-base	No	Completo	ROUGE-1	0,160	0,062	0,071
			ROUGE-2	0,025	0,019	0,020
			ROUGE-L	0,137	0,045	0,053
			ROUGE-Lsum	0,141	0,046	0,053
	Sí	Completo	ROUGE-1	0,313	0,432	0,349
			ROUGE-2	0,116	0,163	0,130
			ROUGE-L	0,205	0,289	0,231
			ROUGE-Lsum	0,204	0,289	0,230
	Sí	Filtrado	ROUGE-1	0,311	0,423	0,345
			ROUGE-2	0,117	0,161	0,130
			ROUGE-L	0,209	0,286	0,232
			ROUGE-Lsum	0,208	0,286	0,232
pegasus-xsum	No	Completo	ROUGE-1	0,405	0,253	0,305
			ROUGE-2	0,135	0,083	0,101
			ROUGE-L	0,277	0,173	0,209
			ROUGE-Lsum	0,278	0,174	0,209
	Sí	Completo	ROUGE-1	0,459	0,333	0,368
			ROUGE-2	0,202	0,136	0,154
			ROUGE-L	0,336	0,237	0,263
			ROUGE-Lsum	0,352	0,252	0,277
	Sí	Filtrado	ROUGE-1	0,464	0,353	0,383
			ROUGE-2	0,196	0,142	0,156
			ROUGE-L	0,334	0,251	0,272
			ROUGE-Lsum	0,355	0,267	0,290

Por otro lado, si se compara el modelo equivalente de la familia T5, es decir, *t5-small-xsum*, el resumen generado es el siguiente:

The latest flight cancellations by country.

Si a nivel numérico las diferencias ya eran más que evidentes, a nivel cualitativo es innegable el mejor rendimiento de Pegasus sobre T5. El primer modelo consigue generar un resumen coherente y fluido, más allá de lo bien que consiga resumir el contenido de la noticia original. Sin embargo, el segundo resumen que se presenta carece de sentido y evidencia un mal funcionamiento del modelo, donde este es incapaz de generar una frase y se queda

5.2 Extracción de las ideas principales

En esta segunda tarea, el objetivo es identificar y extraer el contenido más relevante de un texto, en lugar de simplemente generar un resumen del mismo. Esto requiere

que el modelo identifique las palabras clave y frases más significativas que mejor representen el contenido principal.

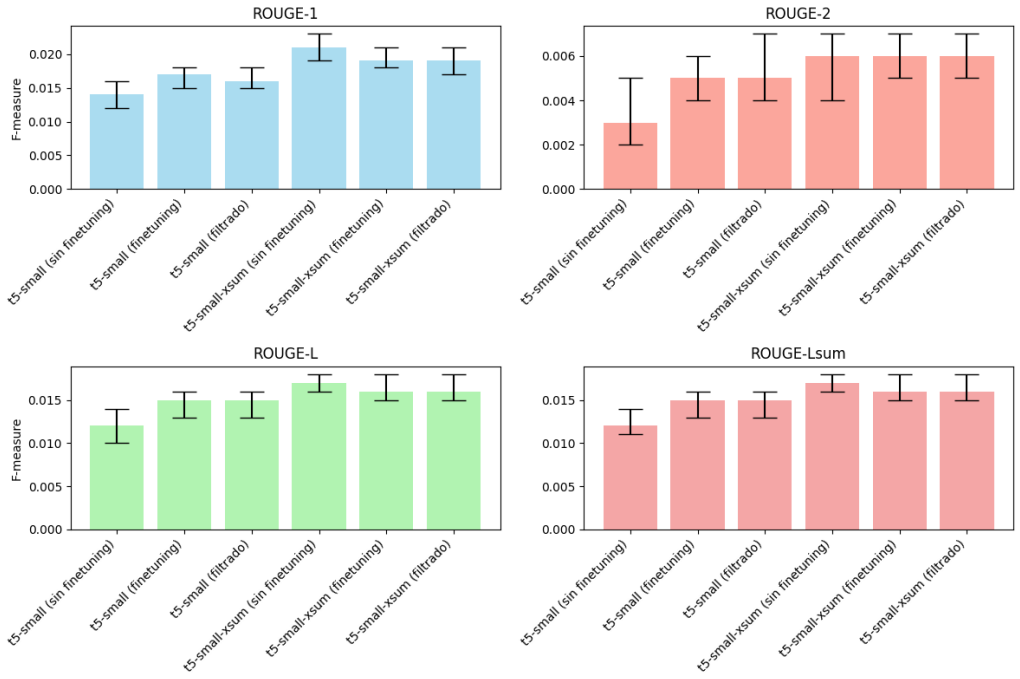

Figura 5.3: Comparación entre los distintos modelos de T5 para extracción de ideas principales según F-measure.

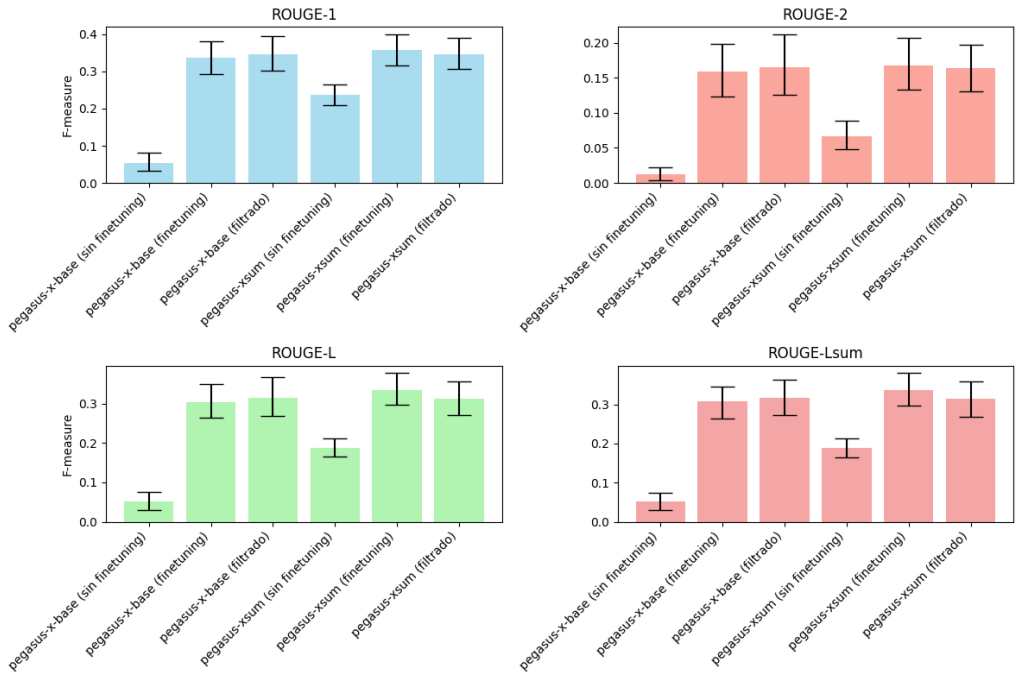

Figura 5.4: Comparación entre los distintos modelos de Pegasus para extracción de ideas principales según F-measure.

El rendimiento de los modelos T5 en la tarea de extracción de ideas sigue un patrón similar al observado en la tarea de generación de resúmenes. Los modelos preentrenados en crudo, sin ningún ajuste fino, presentan dificultades para extraer ideas

relevantes de forma precisa. En particular, el modelo *t5-small* presenta unos valores bajos de ROUGE-2, con F-measure por debajo de 0,005, lo que indica que el modelo no captura adecuadamente las relaciones contextuales o semánticas entre las palabras clave dentro de la noticia.

Tras aplicar *finetuning*, los resultados mejoran, especialmente en ROUGE-Lsum y ROUGE-L, donde el modelo *t5-small-xsum* ajustado alcanza valores alrededor de 0,015 en F-measure. Sin embargo, los resultados no son tan altos como en la tarea de generación de resúmenes, lo que sugiere que la extracción de ideas principales es una tarea más compleja para los modelos T5. Esto puede deberse a que esta tarea requiere una comprensión más profunda del contenido y contexto del texto, mientras que la generación de resúmenes puede depender más de patrones preentrenados.

Los modelos Pegasus también destacan en esta tarea, y aunque el rendimiento en la tarea de extracción de ideas es inferior al de la generación de resúmenes, los valores obtenidos son considerablemente más altos que los de T5. Por ejemplo, *pegasus-xsum* sin ajuste fino alcanza valores cercanos a 0,10 en ROUGE-2, y tras el *finetuning*, los valores de ROUGE-Lsum y ROUGE-L superan el 0,30 en F-measure.

Este comportamiento refuerza la hipótesis de que Pegasus está mejor diseñado para captar relaciones contextuales complejas en el texto, lo que le permite identificar mejor las ideas clave que describen la esencia de la noticia.

Al igual que en la tarea de generación de resúmenes, el uso de conjuntos filtrados no parece ofrecer una mejora significativa en las métricas de extracción de ideas principales. Las diferencias entre los modelos entrenados con el conjunto general y el conjunto filtrado son mínimas, lo que sugiere que el filtrado del conjunto de datos no es el factor clave en el rendimiento del modelo.

Al analizar las métricas ROUGE para la extracción de ideas principales en los modelos T5 y Pegasus, se observa una clara tendencia en los resultados que refuerza la importancia de ajustar finamente estos modelos, tal como se ha evidenciado en la tarea de generación de resúmenes. Los modelos ajustados con *finetuning* y aquellos entrenados con conjuntos de datos filtrados tienden a obtener mejores puntuaciones tanto en *precision* como en F-measure, lo cual es esencial en tareas como la extracción de ideas donde se necesita capturar con precisión la esencia del texto original.

En la Tabla 5.3, que presenta los resultados para el modelo *pegasus-x-base* en crudo, se observa que las métricas de F-measure y *recall* son considerablemente bajas. En particular, ROUGE-1 en su valor medio solo llega a una F-measure de 0,054, mientras que ROUGE-2 y ROUGE-L presentan cifras aún menores, con una F-measure de 0,012 y 0,051 respectivamente. Esto refuerza la idea de que los modelos sin ajuste fino no pueden capturar adecuadamente la información clave de los textos, lo que resulta en un pobre rendimiento en tareas que requieren una alta precisión y coherencia.

Al aplicar *finetuning* en el modelo *pegasus-x-base* (ver Tabla 5.3), el rendimiento mejora significativamente. El valor medio de F-measure para ROUGE-1 asciende a 0,337, un aumento notable en comparación con el modelo sin ajuste fino. Las métricas ROUGE-2 y ROUGE-L también muestran mejoras significativas, alcanzando una F-measure de 0,159 y 0,305, respectivamente. Estos resultados destacan la importancia de ajustar finamente el modelo para lograr una extracción de ideas más precisa y

Tabla 5.3: Resultados de la evaluación de modelos Pegasus en extracción de ideas principales.

Modelo	Finetuning	Datos	Metric	Precision	F-measure
pegasus-x-base	No	Completo	ROUGE-1	0,084	0,054
			ROUGE-2	0,008	0,012
			ROUGE-L	0,081	0,051
	Sí	Completo	ROUGE-1	0,462	0,337
			ROUGE-2	0,229	0,159
			ROUGE-L	0,420	0,305
	Sí	Filtrado	ROUGE-1	0,459	0,347
			ROUGE-2	0,232	0,165
			ROUGE-L	0,420	0,315
pegasus-xsum	No	Completo	ROUGE-1	0,181	0,237
			ROUGE-2	0,052	0,067
			ROUGE-L	0,144	0,188
	Sí	Completo	ROUGE-1	0,481	0,358
			ROUGE-2	0,233	0,168
			ROUGE-L	0,450	0,335
	Sí	Filtrado	ROUGE-1	0,479	0,347
			ROUGE-2	0,237	0,164
			ROUGE-L	0,434	0,313

coherente. Además, los intervalos de confianza (ver Figura 5.4) son más reducidos, lo que indica mayor consistencia en los resultados.

Cuando se filtran las noticias, se observan ligeras mejoras adicionales, particularmente en ROUGE-L, donde la F-measure aumenta a 0,315 en el valor medio. Sin embargo, las diferencias no son tan marcadas como en el paso de no aplicar *finetuning* a aplicarlo. Esto sugiere que, si bien los datos filtrados pueden aportar una mejora, el impacto más significativo proviene del ajuste en el conjunto de datos completo.

Por otro lado, el modelo *pegasus-xsum* sin *finetuning* muestra una mejor capacidad de extracción de ideas que *pegasus-x-base* en su forma cruda, con una F-measure media de 0,237 para ROUGE-1, aunque aún por debajo de los modelos ajustados. Al aplicar *finetuning* los resultados mejoran significativamente, con una F-measure de 0,358 para ROUGE-1 y un aumento similar en ROUGE-2 y ROUGE-L. Estos resultados subrayan el valor de ajustar finamente el modelo en el conjunto completo de datos para mejorar la precisión en la tarea de extracción.

En el caso de T5, los resultados obtenidos (ver Tabla 5.4) son consistentes con lo observado en la tarea de generación de resúmenes. Aunque la *precision* es alta (0,573 en ROUGE-1 para *t5-small* sin *finetuning*), el *recall* y la F-measure son extremadamente bajos, llegando apenas a 0,014 en la F-measure para ROUGE-1. Esto muestra que, aunque T5 puede identificar palabras clave relevantes, tiene dificultades para capturar la estructura completa y las ideas principales del texto en esta configuración. Tras aplicar finetuning hay una ligera mejora en *recall* y F-measure, pero los resultados siguen siendo bajos en comparación con los obtenidos con Pegasus.

Tabla 5.4: Resultados de la evaluación de modelos T5 en extracción de ideas principales.

Modelo	Finetuning	Datos	Metric	Precision	F-measure
t5-small	No	Completo	ROUGE-1	0,573	0,014
			ROUGE-2	0,150	0,003
			ROUGE-L	0,510	0,012
	Sí	Completo	ROUGE-1	0,613	0,017
			ROUGE-2	0,198	0,005
			ROUGE-L	0,537	0,015
	Sí	Filtrado	ROUGE-1	0,620	0,016
			ROUGE-2	0,222	0,005
			ROUGE-L	0,556	0,015
t5-small-xsum	No	Completo	ROUGE-1	0,626	0,021
			ROUGE-2	0,179	0,006
			ROUGE-L	0,510	0,017
	Sí	Completo	ROUGE-1	0,634	0,019
			ROUGE-2	0,212	0,006
			ROUGE-L	0,544	0,016
	Sí	Filtrado	ROUGE-1	0,624	0,019
			ROUGE-2	0,207	0,006
			ROUGE-L	0,536	0,016

En cuanto al uso de noticias filtradas, los resultados siguen la misma tendencia observada en los modelos Pegasus. El filtrado de noticias mejora ligeramente las métricas, pero no lo suficiente como para suponer una diferencia significativa respecto a los resultados obtenidos con el conjunto completo de datos.

En cuanto a la calidad escrita de los resúmenes, en este segundo objetivo también se ven diferencias claras, como por ejemplo en este caso de un modelo de T5:

Lufthansa has canceled 2,000 more flights to Europe, a German carrier has

Y por otro lado, una idea principal obtenida a partir de Pegasus:

India has increased sanitary rules to prevent Covid-19 cases in neighbouring China.

En el primer caso, la tendencia general de las ideas extraídas a partir de los modelos de la familia T5 es generar frases inclompletas, lo que se traduce en una calidad cuestionable y métricas más bajas de lo deseado. Por otro lado, los modelos Pegasus, como bien se aprecia en el segundo ejemplo, son capaces de generar ideas completas y más coherentes, pudiendo desempeñar una función más util en la extracción de ideas principales. Estos análisis cualitativos confirman los análisis cuantitativos que se han realizado en ambas secciones, concluyendo que los modelos Pegasus consiguen mejor rendimiento y pueden generar textos coherentes y útiles para la función deseada por la empresa.

5.3 Discusión

A lo largo de los experimentos, se observan varias tendencias consistentes. En primer lugar, los modelos preentrenados en crudo no ofrecen un rendimiento adecuado para ninguna de las dos tareas, tanto en la generación de resúmenes como en la extracción de ideas principales. Esto puede explicarse por la falta de especialización de los modelos en el dominio de las noticias, que requiere una comprensión profunda de los textos y sus relaciones contextuales.

El *finetuning* de los modelos en un conjunto de datos específico mejora significativamente las métricas ROUGE. Esta mejora es más notable en las métricas de *precision* y F-measure, lo que indica que los modelos ajustados son más capaces de generar contenido relevante y coherente. Sin embargo, el *recall* no mejora de manera tan drástica, lo que sugiere que, aunque los modelos con ajuste fino son más precisos, aún enfrentan desafíos para capturar toda la información relevante.

En cuanto a los modelos T5 y Pegasus, los resultados indican que Pegasus supera sistemáticamente a T5 en ambas tareas, incluso sin realizar *finetuning*. Esto es esperable, dado que Pegasus fue diseñado específicamente para tareas de generación de resúmenes, mientras que T5 es un modelo más generalista que puede aplicarse a una gama más amplia de tareas. No obstante, los modelos T5 ajustados mediante ajuste fino pueden ofrecer un rendimiento aceptable, aunque no alcanzan los niveles de *precision* y *recall* de Pegasus. Quizá sería interesante en un futuro probar primero a realizar *finetuning* de T5 con un conjunto de datos de grandes dimensiones de resúmenes y posteriormente utilizar ese modelo para volver a realizar finetuning sobre estas noticias. De esta forma, podría entrenarse primero el modelo para la tarea de generación de resúmenes y ver si mejoran posteriormente los resultados y calidad de los resúmenes generados.

Finalmente, el uso de conjuntos filtrados ofrece una mejora marginal en las métricas, pero su impacto no es tan significativo como el *finetuning*. Esto sugiere que el ajuste fino en un conjunto de datos específico es la clave para mejorar el rendimiento del modelo, mientras que el filtrado del conjunto de datos puede ofrecer pequeñas mejoras adicionales en términos de precisión.

Los resultados de este trabajo sugieren que el *finetuning* es esencial para mejorar el rendimiento de los modelos preentrenados en la tarea de generación de resúmenes y la extracción de ideas principales. Además, los modelos especializados, como Pegasus, ofrecen un rendimiento superior en comparación con modelos más generalistas como T5. Sin embargo, el uso de conjuntos filtrados ofrece solo mejoras marginales, lo que sugiere que el entrenamiento en un conjunto de datos específico es el factor determinante en el éxito del modelo.

CAPÍTULO 6
Conclusiones y trabajos futuros

En este útlimo capítulo se hará un resumen de todo el trabajo junto con las conclusiones obtenidas después de su realización. Además, se hablará sobre posibles líneas de trabajo a seguir tras haber dibujado las primeras trazas con este proyecto.

6.1 Conclusiones

En este trabajo se ha abordado el problema de la generación automática de resúmenes y la extracción de ideas principales de noticias web utilizando modelos de NLP preentrenados, con especial énfasis en los modelos Pegasus y T5. Los resultados obtenidos a lo largo de los experimentos realizados permiten extraer varias conclusiones relevantes sobre el desempeño y las limitaciones de estos modelos en las tareas propuestas.

En primer lugar, los modelos preentrenados sin finetuning han mostrado un desempeño limitado, evidenciando que, si bien estos modelos cuentan con una gran capacidad para aprender representaciones generales del lenguaje, no son capaces de adaptarse adecuadamente a tareas específicas como la generación de resúmenes o la extracción de ideas principales sin un ajuste adecuado dentro de un dominio tan específico como el que se ha tratado en este proyecto. Esto se observa en los resultados de *precision* y *recall*, donde en varios casos los valores de F-measure han permanecido por debajo del 25 %. Esto sugiere que el entrenamiento adicional con datos específicos es indispensable para mejorar el rendimiento de los modelos en estas tareas.

El finetuning ha demostrado ser una estrategia efectiva para mejorar significativamente el rendimiento de los modelos escogidos. Los resultados finalmente obtenidos indican que el ajuste con conjuntos de datos especializados ha permitido mejorar sustancialmente las métricas de evaluación, especialmente en las tareas de generación de resúmenes. En el caso del modelo pegasus-x-base, el valor de F-measure mejora significativamente tras el ajuste fino, alcanzando un valor de 0.347 en la métrica ROUGE-1, frente a un 0.054 inicial antes del fine-tuning. Este hallazgo resalta la importancia de adaptar los modelos preentrenados a los datos específicos de la tarea, lo que permite a los modelos aprender a identificar patrones y generar resúmenes más precisos y coherentes.

Es importante destacar que el modelo Pegasus, que fue diseñado específicamente para la tarea de generación de resúmenes, ha demostrado un rendimiento superior en comparación con el modelo T5 en la mayoría de las pruebas realizadas. Este resultado está en línea con la hipótesis de que los modelos especializados tienen una ventaja en tareas concretas en comparación con los modelos más generalistas. Aunque T5 ha presentado un buen desempeño en algunas métricas, alcanzando valores de precisión cercanos al 70 % en ciertos experimentos, su rendimiento ha sido más inconsistente en comparación con Pegasus. Esto sugiere que, para tareas como la generación de resúmenes, los modelos diseñados específicamente para esta tarea ofrecen una ventaja competitiva.

Además de la generación de resúmenes, la extracción de ideas principales se ha presentado como una tarea más desafiante. Los valores de F-measure en esta tarea han sido inferiores a los obtenidos en la generación de resúmenes, incluso después de aplicar el fine-tuning. Esto se debe a que la extracción de ideas clave es una tarea más compleja para los modelos preentrenados, posiblemente debido a la naturaleza más concisa y abstracta del resultado que se espera de estos modelos en comparación con los resúmenes generales del contenido. Los resultados indican que los modelos aún tienen dificultades para identificar y condensar correctamente las ideas más importantes de un texto tras el ajuste de estos.

Otro aspecto importante que se ha abordado en este trabajo es el impacto del preprocesamiento y la filtración de datos en el rendimiento de los modelos. En varios experimentos se han utilizado conjuntos de datos filtrados para evaluar si un mejor preprocesamiento de los datos podría mejorar el desempeño de los modelos. Los resultados indican que, si bien el uso de datos filtrados puede mejorar ligeramente las métricas de recall, el incremento no es lo suficientemente significativo como para justificar un cambio radical en la metodología. Esto sugiere que, aunque el preprocesamiento es importante, se requiere de técnicas más sofisticadas para tener un impacto notable en el rendimiento global del modelo.

Tras la revisión, análisis y comparación de las distintas métricas obtenidas en todos los experimentos realizados, se escoge el modelo *pegasus-xsum* con *finetuning* y entrenado con todo el conjunto de noticias tanto para la tarea de generación automática de resúmenes como para la extracción de ideas principales del texto. Esto se debe a que, como se viene resaltando durante el trabajo, los modelos especializados en la generación de resúmenes de textos de noticias han venido demostrando un mejor rendimiento que los generales. También, el finetuning consigue mejorar significativamente en varias de las métricas ROUGE calculadas, por lo que el ajuste tiene un impacto real en los resúmenes obtenidos. Comparando entre los modelos ajustados con la base de datos general y la específica, se ha observado que en ciertas ocasiones el filtrado mejora los resultados, pero de forma muy ligera por lo que no merece la pena utilizar más recursos en el filtrado y preprocesamiento para los números finalmente obtenidos. Un punto positivo de esta conclusión es el ahorro computacional de la empresa, ya que se puede utilizar un mismo modelo para ambas tareas. Desde el punto de vista industrial, esta ventaja es beneficiosa en términos cuantitativos, ya que siempre es deseable la optimización de modelos para tratar de ser lo más eficiente posible a todos los niveles.

A pesar de los avances logrados, este trabajo también presenta ciertas limitaciones que deben ser reconocidas. Una de las principales limitaciones es la capacidad de ge-

neralización de los modelos. Aunque se han logrado mejoras significativas con el fine-tuning, los modelos aún tienen dificultades para generalizar adecuadamente cuando se enfrentan a textos de temáticas muy diversas o que presentan estilos de escritura menos convencionales. Esto se observa en la tarea de extracción de ideas principales, donde los resultados son más inconsistentes que en la generación de resúmenes. La capacidad de los modelos para captar el contexto y la semántica de ideas clave es un área que aún necesita mejoras.

Además, es importante considerar que los modelos basados en NLP actuales aún dependen en gran medida de la calidad y diversidad del corpus de entrenamiento. Modelos como Pegasus y T5 pueden mostrar sesgos hacia los datos con los que han sido entrenados, lo que afecta la neutralidad y precisión de los resúmenes generados. Esto resalta la necesidad de utilizar corpus de entrenamiento lo más amplios y diversos posible para mejorar la capacidad de generalización de estos modelos.

Por último, una de las limitaciones más notadas a lo largo de la ejecución de los modelos es la capacidad de cómputo de las máquinas utilizadas. A pesar de que ForwardKeys es una empresa con un gran potencial, el hecho de no tener procesos basados en inteligencia artificial ha supuesto una barrera a la hora de realizar este trabajo. A día de hoy, los procesos que más capacidad de cómputo necesitan son modelos de aprendizaje automático, que consiguen ejecutarse sin problema con CPU. Por este motivo, a pesar de que finalmente se ha podido disponer de una máquina con GPU, esta no era lo suficientemente potente como para ejecutar modelos más grandes y por tanto más pesados que generalmente suele ofrecer mejores resultados. Esta limitación ha afectado principalmente a la elección de los modelos y al número de noticias permitido en el uso del entrenamiento de los modelos.

6.2 Trabajos futuros

Con base en los resultados obtenidos, se abren varias direcciones para investigaciones futuras. En primer lugar, a pesar de contar con un modelo que ha demostrado rendir mejor que el resto en ambas tareas, se podría seguir investigando y buscar otros modelos para tratar de encontrar el mejor posible. Para eso, como recién se ha mencionado en la sección anterior, se necesitaría una máquina con mayor capacidad de cómputo. Además, la combinación de modelos de NLP con enfoques más sofisticados de preprocesamiento de datos podría ayudar a mejorar la capacidad de los modelos para captar la esencia de un texto de manera más precisa. Técnicas de aprendizaje activo o entrenamiento con retroalimentación podrían permitir que los modelos se ajusten de manera más dinámica y eficiente a los datos específicos de la tarea.

Finalmente, un trabajo futuro por hacer, dado que es la continuación pensada tras este desarrollo, es la integración de este modelo como un sistema de generación automática de resúmenes para portales de noticias como Feedly. Las mejoras en el rendimiento de los modelos presentados en este trabajo servirán de base para desarrollar herramientas más robustas y precisas para la automatización de tareas de resumen y extracción de ideas clave, facilitando la accesibilidad y procesamiento de grandes volúmenes de información textual en tiempo real.

A lo largo del desarrollo de este trabajo, se han identificado diversas áreas que pueden explorarse y mejorarse en investigaciones futuras. Aunque los resultados obtenidos reflejan avances importantes en la tarea de generación de resúmenes y la extracción de ideas principales, también han revelado ciertas limitaciones de los modelos y técnicas empleadas, lo que abre un amplio espacio para futuras mejoras y estudios adicionales.

Una posible línea a investigar podría ser el uso de entrenamiento continuo o multitarea para que los modelos preentrenados puedan aprender de forma acumulativa a través de diferentes tareas, mejorando su capacidad de generalización sin necesidad de realizar múltiples sesiones de fine-tuning específicas para cada tarea.

El desempeño de los modelos en la tarea de extracción de ideas principales ha sido notablemente inferior al de la generación de resúmenes, lo que sugiere que esta área aún presenta desafíos considerables. Para abordar este problema, investigaciones futuras podrían explorar enfoques más avanzados de modelado semántico, como el uso de redes neuronales de atención más complejas o Transformers especializadas en la identificación de elementos clave. Estas técnicas permitirían a los modelos capturar mejor las relaciones semánticas profundas entre las frases, mejorando su capacidad para identificar las ideas más relevantes y centrales en un texto. Además, una línea interesante de investigación sería combinar técnicas de entrenamiento supervisado y no supervisado, utilizando enfoques como el aprendizaje de representación densa para mejorar la capacidad del modelo de entender el contenido de los textos a nivel más profundo y estructural.

Por otro lado, el preprocesamiento de datos ha sido identificado como un factor clave para mejorar el rendimiento de los modelos. En trabajos futuros, sería interesante investigar el impacto de técnicas de preprocesamiento más avanzadas, como la filtración semántica automática, en la que se eliminan o destacan partes irrelevantes de los textos antes de que los modelos los procesen. Esto podría ayudar a los modelos a centrarse en la información más relevante desde el principio, reduciendo el ruido en los datos de entrada.

También sería valioso estudiar el uso de representaciones más ricas de texto, como *embeddings* mejorados, para mejorar la capacidad del modelo de identificar patrones y relaciones entre palabras y frases. Por ejemplo, el uso de *embeddings* contextuales como BERT o RoBERTa junto con los modelos actuales podría mejorar la calidad de los resúmenes generados o la identificación de ideas principales, ya que proporcionan representaciones más matizadas y ricas del significado contextual.

Otra línea de investigación interesante sería la creación de sistemas híbridos que combinen lo mejor de varios modelos de procesamiento de lenguaje natural. En lugar de depender de un solo modelo, como Pegasus o T5, se podrían combinar varios enfoques, utilizando modelos específicos para la identificación de ideas clave y otros modelos para la generación de resúmenes. Estos sistemas híbridos podrían optimizar la extracción de información mediante un enfoque más segmentado y especializado en cada subtarea del procesamiento del lenguaje.

Además, se podría experimentar con la combinación de enfoques de reglas basadas en heurísticas y modelos estadísticos o basados en deep learning. La inclusión de reglas lingüísticas y conocimientos especializados puede ser útil para guiar a los

modelos en ciertas tareas donde los patrones lingüísticos y sintácticos sean cruciales, como la extracción de ideas principales.

Finalmente, un área prometedora de investigación es la mejora en los métodos de evaluación de la calidad de los resúmenes y las ideas principales generadas. Una metodología de Human-in-the-Loop, en la que los usuarios humanos evalúan y retroalimentan al sistema en tiempo real, podría mejorar significativamente la precisión de los modelos. En este enfoque, los humanos desempeñarían un rol activo en el proceso de refinamiento del sistema, proporcionando retroalimentación que los modelos podrían utilizar para mejorar su capacidad de generar resúmenes y extraer ideas clave más coherentes y útiles.

Además de los trabajos en torno a la optimización técnica, uno de los claros trabajos futuros de este proyecto es la implementación de estos modelos en la industria, dado que es el principal objetivo inmediato de este trabajo. La idea del tema de este proyecto venía motivado por la necesidad real de los trabajadores de disponer de alguna herramienta que agilizara el proceso de selección de noticias, por lo que una vez desarrollada la herramienta de generación de resúmenes, el siguiente paso claramente inmediato es su puesta en marcha en producción. Otra opción que también sería interesante sería la de desarrollar también un clasificador que categorizara las noticias según si estas fueran a ser relevantes para la empresa o no. De esta forma, el trabajador únicamente tendría que ver la clasificación que se le ha dado a las noticias, leer el resumen y finalmente decidir si realmente le resulta relevante o no.

Fuera de la temática de este proyecto, este trabajo también abre las puertas al mundo de la inteligencia artificial en Forwardkeys. Además de la parte de procesamiento de lenguaje natural, también se propone como trabajo futuro trabajar en un ámbito con datos de naturaleza más numérica. Esto se hará con la intención de desarrollar modelos capaces de detectar anomalías, datos incorrectos, tendencias y muchos otros patrones cruciales para poder anticipar situaciones y aumentar el potencial de la empresa.

Bibliografía

[1] Haisal Dauda Abubakar, Mahmood Umar y Muhammad Abdullahi Bakale. "Sentiment classification: Review of text vectorization methods: Bag of words, Tf-Idf, Word2vec and Doc2vec". En: *SLU Journal of Science and Technology* 4.1 (2022), págs. 27-33.

[2] Abien Fred M. Agarap. "A Neural Network Architecture Combining Gated Recurrent Unit (GRU) and Support Vector Machine (SVM) for Intrusion Detection in Network Traffic Data". En: *Proceedings of the 10th International Conference on Machine Learning and Computing*. Vol. 5. 2018, 26–30.

[3] Prafulla Bafna, Dhanya Pramod y Anagha Vaidya. "Document clustering: TF-IDF approach". En: *Proceedings of the International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*. 2016, págs. 61-66.

[4] Bartłomiej Balcerzak, Wojciech Jaworski y Adam Wierzbicki. "Application of TextRank Algorithm for Credibility Assessment". En: *Proceedings of the IEEE/-WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)* (2014), págs. 451-454.

[5] Sergey Brin y Lawrence Page. "The anatomy of a large-scale hypertextual Web search engine". En: *Computer Networks and ISDN Systems* 30.1 (1998), págs. 107-117.

[6] Tom B Brown. "Language models are few-shot learners". En: *arXiv preprint arXiv:2005.14165* (2020).

[7] Jianpeng Cheng y Mirella Lapata. "Neural Summarization by Extracting Sentences and Words". En: *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 2016, págs. 484-494.

[8] Jeffrey L Elman. "Finding structure in time". En: *Cognitive science* 14.2 (1990), págs. 179-211.

[9] Jiwen Fan et al. "Substantial convection and precipitation enhancements by ultrafine aerosol particles". En: *Science* 359.6374 (2018), págs. 411-418.

[10] Kavita Ganesan. "ROUGE 2.0: Updated and Improved Measures for Evaluation of Summarization Tasks". En: *CoRR* abs/1803.01937 (2018).

[11] Tahmid Hasan et al. "XL-Sum: Large-Scale Multilingual Abstractive Summarization for 44 Languages". En: *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*. 2021, págs. 4693-4703.

[12] Yingmin Jin et al. "Building a highly functional Li1.3Al0.3Ti1.7(PO4)3/poly (vinylidene fluoride) composite electrolyte for all-solid-state lithium batteries". En: *Journal of Alloys and Compounds* 874 (2021), pág. 159890.

[13] Jacob Devlin Ming-Wei Chang Kenton y Lee Kristina Toutanova. "Bert: Pre-training of deep bidirectional transformers for language understanding". En: *Proceedings of naacL-HLT*. 2019, págs. 4171-4186.

[14] Mikhail V Koroteev. "BERT: a review of applications in natural language processing and understanding". En: *arXiv preprint arXiv:2103.11943* (2021).

[15] Faisal Ladhak et al. "WikiLingua: A New Benchmark Dataset for Cross-Lingual Abstractive Summarization". En: *EMNLP 2020*. 2020, págs. 4034-4048.

[16] Yann LeCun et al. "Gradient-based learning applied to document recognition". En: *Proceedings of the IEEE* 86.11 (1998), págs. 2278-2324.

[17] Yahui Li et al. "Exploring semantic awareness via graph representation for text classification". En: *Applied Intelligence* 53.2 (2022), págs. 2088-2097.

[18] Elizabeth D Liddy. "Encyclopedia of Library and Information Science". En: Marcel Decker, Inc, 2001. Cap. Natural language processing.

[19] Chin-Yew Lin. "ROUGE: A Package for Automatic Evaluation of Summaries". En: *Text Summarization Branches Out*. 2004, págs. 74-81.

[20] Yang Liu y Mirella Lapata. "Text Summarization with Pretrained Encoders". En: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2019, págs. 3730-3740.

[21] Yang Liu et al. "Topical word embeddings". En: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 29. 1. 2015.

[22] K Usha Manjari. "Extractive Summarization of Telugu Documents using TextRank Algorithm". En: *Proceedings of the Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*. 2020, págs. 678-683.

[23] Lluis Marquez, Lluis Padro y Horacio Rodriguez. "A machine learning approach to POS tagging". En: *Machine Learning* 39 (2000), págs. 59-91.

[24] Rada Mihalcea y Paul Tarau. "TextRank: Bringing Order into Text". En: *Proceedings of the 2004 Conference on Empirical Methods in Natural Language Processing*. 2004, págs. 404-411.

[25] Sabino Miranda, Alexander Gelbukh y Grigori Sidorov. "Generación de resúmenes por medio de síntesis de grafos conceptuales". En: *Revista Signos* 47.86 (2014), págs. 463-485.

[26] Rami Mohawesh et al. "Semantic graph based topic modelling framework for multilingual fake news detection". En: *AI Open* 4 (2023), págs. 33-41.

[27] Ramesh Nallapati, Feifei Zhai y Bowen Zhou. *SummaRuNNer: A Recurrent Neural Network based Sequence Model for Extractive Summarization of Documents*. 2016.

[28] Shashi Narayan et al. "Planning with Learned Entity Prompts for Abstractive Summarization". En: *Transactions of the Association for Computational Linguistics* 9 (2021), págs. 1475-1492.

[29] Wisam A Qader, Musa M Ameen y Bilal I Ahmed. "An overview of bag of words; importance, implementation, applications, and challenges". En: *Proceedings of the international engineering conference (IEC)*. 2019, págs. 200-204.

[30] Lawrence R Rabiner. "A tutorial on hidden Markov models and selected applications in speech recognition". En: *Proceedings of the IEEE* 77.2 (1989), págs. 257-286.

[31] Alec Radford y Karthik Narasimhan. "Improving Language Understanding by Generative Pre-Training". Preprint. 2018.

[32] Colin Raffel et al. "Exploring the limits of transfer learning with a unified text-to-text transformer". En: *Journal of machine learning research* 21.140 (2020), págs. 1-67.

[33] Colin Raffel et al. "Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer". En: *Journal of Machine Learning Research* 21.140 (2020), págs. 1-67.

[34] Juan Ramos et al. "Using tf-idf to determine word relevance in document queries". En: *Proceedings of the first instructional conference on machine learning*. Vol. 242. 1. 2003, págs. 29-48.

[35] Gerard Salton y Christopher Buckley. "Term-weighting approaches in automatic text retrieval". En: *Information processing & management* 24.5 (1988), págs. 513-523.

[36] Abigail See, Peter J. Liu y Christopher D. Manning. *Get To The Point: Summarization with Pointer-Generator Networks*. 2017.

[37] Alex Sherstinsky. "Fundamentals of recurrent neural network (RNN) and long short-term memory (LSTM) network". En: *Physica D: Nonlinear Phenomena* 404 (2020), págs. 132-306.

[38] Grigori Sidorov et al. "Syntactic N-grams as machine learning features for natural language processing". En: *Expert Systems with Applications* 41.3 (2014), págs. 853-860.

[39] Shouyou Song et al. "A Novel Text Classification Approach Based on Word2vec and TextRank Keyword Extraction". En: *Proceedings of the IEEE Fourth International Conference on Data Science in Cyberspace (DSC)*. 2019, págs. 536-543.

[40] Ilya Sutskever, Oriol Vinyals y Quoc V. Le. "Sequence to Sequence Learning with Neural Networks". En: *CoRR* abs/1409.3215 (2014).

[41] Gemma Team et al. "Gemma: Open models based on gemini research and technology". En: *arXiv preprint arXiv:2403.08295* (2024).

[42] Ashish Vaswani et al. "Attention is all you need". En: *Advances in Neural Information Processing Systems*. 2017, págs. 5998-6008.

[43] Mingye Wang et al. "T5-Based Model for Abstractive Summarization: A Semi-Supervised Learning Approach with Consistency Loss Functions". En: *Applied Sciences* 13.12 (2023).

[44] Qizhe Xie et al. "Unsupervised Data Augmentation for Consistency Training". En: *Advances in Neural Information Processing Systems*. 2020, págs. 6256-6268.

[45] Zhilin Yang et al. "XLNet: Generalized Autoregressive Pretraining for Language Understanding". En: *Advances in Neural Information Processing Systems*. Ed. por H. Wallach et al. 2019.

[46] Sarika Zaware et al. "Text Summarization using TF-IDF and Textrank algorithm". En: *Proceedings of the 5th International Conference on Trends in Electronics and Informatics (ICOEI)*. 2021, págs. 1399-1407.

[47] Jingqing Zhang et al. "PEGASUS: pre-training with extracted gap-sentences for abstractive summarization". En: *Proceedings of the 37th International Conference on Machine Learning*. 2020.

[48] Mingxi Zhang et al. "An Empirical Study of TextRank for Keyword Extraction". En: *IEEE Access* 8 (2020), págs. 178849-178858.

[49] Ming Zhong et al. "Extractive Summarization as Text Matching". En: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2020, págs. 6197-6208.