INSTITUTO DE BOLSAS Y MERCADOS ESPAÑOLES

TRABAJO FIN DE MÁSTER

Búsqueda Semántica en Documentos Financieros: aproximación basada en *Embeddings* y *Topic Modeling*

Autores:
Antonio BERNAL,
Alejandro MUÑOZ,
Borja SOLANO

Tutor:
Fernando DE LA CALLE

Trabajo de fin de máster presentado para obtener el título de Máster en Inteligencia Artificial aplicada a Mercados Financieros

26 de febrero de 2023

INSTITUTO DE BOLSAS Y MERCADOS ESPAÑOLES

Resumen

Máster en Inteligencia Artificial aplicada a Mercados Financieros

Búsqueda Semántica en Documentos Financieros: aproximación basada en *Embeddings* y *Topic Modeling*

por Antonio BERNAL, Alejandro MUÑOZ, Borja SOLANO

El trabajo propuesto se debe a la falta de un buscador especializado en el sector financiero, que permita encontrar de manera rápida y eficiente empresas cotizadas que estén relacionadas con temas específicos o noticias concretas.

Se presenta una metodología para analizar descripciones y *earnings calls* de empresas. El objetivo es identificar patrones semánticos y temas relevantes que puedan ser útiles para la toma de decisiones. La metodología se basa en el preprocesamiento de texto, la construcción de *embeddings* y el modelado por tópicos. Los resultados muestran que la metodología es capaz de identificar patrones semánticos y temas relevantes en dichos documentos. Además, se desarrolla una aplicación *web* que permite a los usuarios realizar búsquedas semánticas. Estos resultados sugieren que el análisis de lenguaje natural puede ser una herramienta valiosa para la toma de decisiones empresariales.

Keywords *– Natural Language Processing, Topic Modelling, Embeddings, Azure, Financial Documents*

Agradecimientos

Gracias a nuestras familias por todo el apoyo que nos han dado, gracias a los compañeros por la ayuda brindada durante estos meses y gracias al equipo docente del Instituto BME por los conocimientos que nos ha permitido obtener.

Índice general

Índice de figuras

Índice de cuadros

Abreviaciones

API	***A**pplication **P**rogramming **I**nterface*
BERT	***B**idirectional **E**ncoder **R**epresentations (from) **T**ransformers*
CBOW	***C**ontinuous **B**ag **O**f **W**ords*
CPU	***C**entral **P**rocessing **U**nit*
CSS	***C**ascading **S**tyle **S**eets*
CSV	***C**omma-**S**eparated **V**alues*
GB	***G**iga**B**yte*
GPU	***G**raphics **P**rocessing **U**nit*
HDBSCAN	***H**ierarchical **D**ensity-**B**ased **S**patial **C**lustering (of) **A**pplications (with) **N**oise*
HTML	***H**yper**T**ext **M**arkup **L**anguage*
IA	**I**nteligencia **A**rtificial
LDA	***L**atent **D**irichlet **A**llocation*
LSA	***L**atent **S**emantic **A**nalysis*
MB	***M**ega**B**yte*
MMR	***M**aximal **M**arginal **R**elevance*
NLP	***N**atural **L**anguage **P**rocessing*
PCA	***P**rincipal **C**omponent **A**nalysis*
RAM	***R**andom **A**ccess **M**emory*
SEC	***S**ecurities (and) **E**xchange **C**ommission*
SVD	***S**ingular **V**alue **D**ecomposition*
TF-IDF	***T**erm **F**recuency-**I**nverse **D**ocument **F**recuency*
UMAP	***U**niform **M**anifold **A**pproximation**P**rojection*
URL	***U**niform **R**esource **L**ocators*
UTC	***U**niversal **T**ime **C**oordinated*

Capítulo 1

Introducción

El tiempo es el recurso más valioso que existe, y la Inteligencia Artificial (IA), más allá de ofrecer soluciones innovadoras y disruptivas, ha dado la capacidad de reducir de manera drástica el tiempo invertido en la mayor parte de las tareas de resolución, investigación, comprensión o creatividad, realizadas actualmente por humanos.

De entre todas las infinitas soluciones que han llegado y están por venir, y gracias a la disciplina del *Natural Language Processing (NLP)* como subárea de la Inteligencia Artificial y la Lingüística, se ha centrado este trabajo en la búsqueda de acciones financieras, relacionadas con un tema específico. Dicho tema podría ser introducido mediante palabras clave o podría ser extraído de un texto, como el de una noticia.

1.1 Motivación

Los grandes proveedores de datos financieros suelen clasificar a las empresas por sectores. Por ejemplo, si se buscan compañías tecnológicas pueden encontrarse *Adobe, Advanced Micro Devices(AMD) o Microsoft*. En cambio, la primera de ellas se dedica al desarrollo de software para aplicaciones, la segunda a la fabricación de procesadores de computación y la tercera a desarrollar software para sistemas y servicios en la nube. *Microsoft* obtiene gran parte de sus ingresos por sus servicios *cloud*, al igual que le pasa a *Amazon* pero, ¿en qué sector se encuentra categorizada *Amazon*? En el sector consumo (FIGURA 1.1).

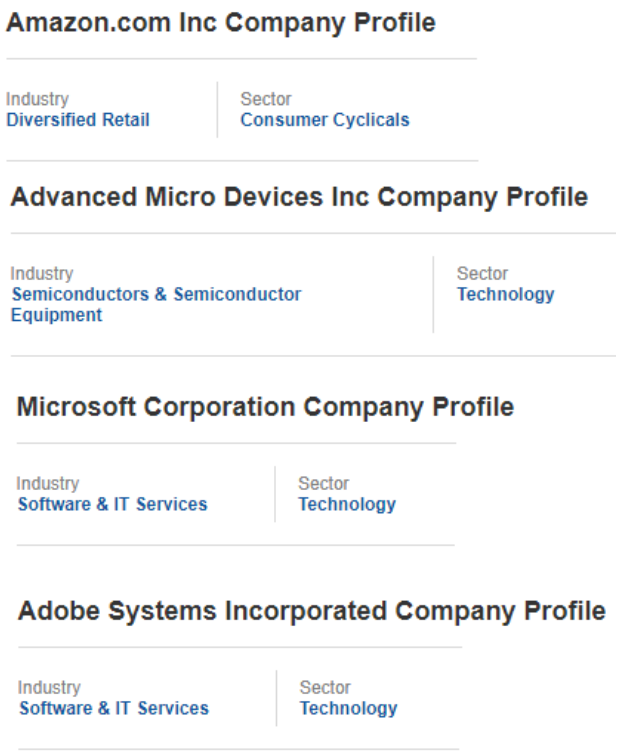

FIGURA 1.1: Clasificación de compañías extraídas de *Investing*.

¿Podría pensarse que las empresas están mal clasificadas? Pues no, no lo están, pero existe mucha información que se le escapa al inversor debido a ello. Si una persona ajena al mundo financiero quisiera invertir, podría tomar dos vías: pagar por la gestión y el asesoramiento, o gestionar sus propias inversiones. Si dicha persona tiene la convicción de que cierto sector o industria va a verse beneficiado o perjudicado en los próximos años y desea invertir, ¿cómo puede elegir la mejor opción?

En este trabajo, se pretende proporcionar al usuario el mayor universo de acciones cotizadas disponibles, relacionadas con el tema donde desee invertir. Suponiendo una situación en la que el descubrimiento de un medicamento fuera a causar un gran impacto y se deseara invertir en él, podría caerse en el error de pensar que las mejores posicionadas para invertir sólo fuesen las farmacéuticas que producen dicho medicamento. Sin embargo, podría darse el caso de que la mayor rentabilidad la diese una empresa logística que tiene el monopolio del transporte de dicho medicamento.

1.2 Objetivos

La finalidad de este trabajo consiste en desarrollar una herramienta financiera que permita a gestores, asesores e incluso a inversores *retail*:

- Encontrar compañías que puedan verse relacionadas con algún tema o noticia.
- Descubrir nuevas compañías.
- Asociar compañías a otras similares.
- Poder disponer de información actualizada.
- Acceso centralizado *webs* de cada compañía desde un único punto.
- Proporcionar herramientas gráficas interactivas.

1.3 Estructura de la Memoria

El **Capítulo** 1 proporciona una visión general de la motivación y los objetivos del proyecto. El **Capítulo** 2 presenta los fundamentos teóricos en los que se basa el proyecto y trabajos previos relacionados con él. El **Capítulo** 3 detalla toda la metodología empleada desde la obtención de los datos, preprocesamiento del texto y creación de modelos, hasta el despliegue *cloud* con la *API* y la *web*. En el El **Capítulo** 4 se evalúan los resultados obtenidos y en el **Capítulo** 5 se resumen las conclusiones de la memoria y se proponen vías para futuras investigaciones.

Capítulo 2

Fundamentos Teóricos y Trabajos Relacionados

Debido a la necesidad de gestionar grandes volúmenes de información financiera, diversas técnicas de minería de texto y aprendizaje automático se han aplicado para extraer información relevante de documentos.

Una de las técnicas más utilizadas en la búsqueda temática de acciones es el modelado por tópicos. El modelado por tópicos o *topic modelling*, son un conjunto de técnicas de *machine learning* que permiten comprimir el contenido de miles de documentos en un pequeño resumen compuesto por aquellos temas más frecuentes. Podría definirse como un modelo matemático no supervisado que toma como input una serie de documentos D y devuelve una serie de tópicos T que representan el contenido de D, de una manera precisa y coherente [1].

Se comenzó a hablar de ello en la década de los 90, y desde entonces han surgido una serie de hitos que marcan su evolución:

- En 1990 se publica el modelo *Latent Semantic Analysis (LSA*) [2]. Este modelo infiere la representación implícita de la semántica del texto, empleando la estadística. Guarda información sobre qué palabras se usan juntas y cuáles son comunes entre frases. La relación semántica será mayor cuando tengan más palabras en común. Aplica *Singular Value Decomposition (SVD)* que modela las relaciones entre palabras y frases para hallar patrones.
- En 2003 *Blei, Ng* y *Jordan* introducen el modelo *Latent Dirichlet Allocation* (*LDA*) [3]. Esta técnica asume que los documentos están representados como mezclas aleatorias de temas latentes, donde cada uno se caracteriza por una distribución sobre las palabras. Parte de una matriz cuyas

filas son documentos y las columnas son palabras. Como resultado devuelve dos matrices, la primera de documentos por temas y la segunda de temas por palabras.

- En 2013, *Tomas Mikolov* presenta el modelo *word2vec* [4]. Dicho modelo tiene dos variantes que se muestran en la FIGURA 2.1: *CBOW* donde la palabra central es precedida mediante el contexto y *Skip-gram* donde ocurre justo lo contrario. Se emplea una red neuronal con tres capas donde la primera capa (*embedding*) transformará cada palabra del contexto en un vector de *embeddings*, la segunda capa (*GlobalAveragePooling1D*) permite sumar los diferentes *embbedings* y por último, la capa densa que permite predecir la palabra objetivo.

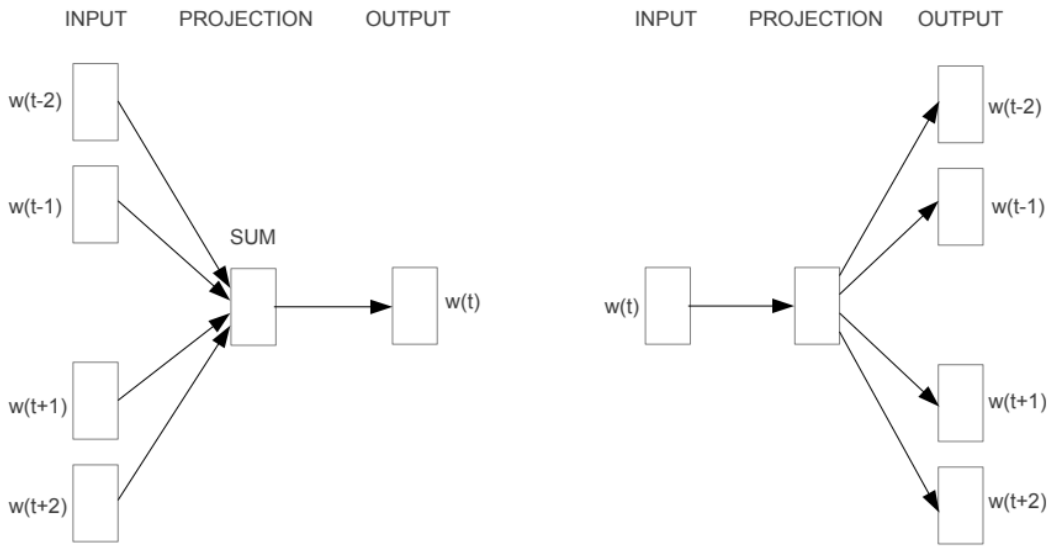

FIGURA 2.1: Tipos de *word2vec*.

- En 2017 se presenta la estructura de red neuronal de tipo *Transformer* [5]. La estructura de *Transformer* contiene un sistema de *encoding* y *decoding* donde se utiliza un modelo de *self-attention* como muestra la FIGURA 2.2.

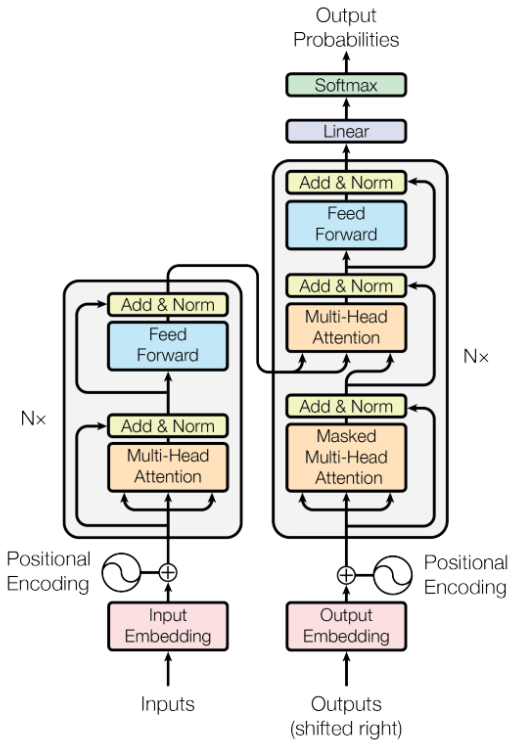

FIGURA 2.2: Arquitectura de un *Transformer*.

- En 2018 surge *Bidirectional Encoder Representations from Transformers (BERT)* [6], que a diferencia del *Transformer*, este modelo solamente contiene la parte del *encoder*.
- Un año después nace *Sentence-BERT* (*S-BERT*) [7]. *S-BERT* es una variante de *BERT* que permite calcular vectores de *embeddings* para oraciones completas utilizando una arquitectura siamesa de redes *BERT*.

  La idea principal de *S-BERT* es utilizar dos instancias de una red *BERT* para procesar dos frases mediante *contrastive learning*[1] comparando sus salidas para calcular una medida de similitud (FIGURA 2.3). Esta medida, se utiliza como un *embedding* de frase para tareas como búsqueda y recuperación de información, clasificación de texto, entre otras. Además, el modelo es más eficiente y escalable que otros modelos de procesamiento de lenguaje natural, lo que lo hace adecuado para su uso en aplicaciones en tiempo real.

[1]El *contrastive learning* es una técnica en la que se comparan dos frases y se trata de maximizar la distancia entre las similares y minimizar la distancia entre las diferentes. Esto permite al modelo aprender a generar *embeddings* que reflejen de manera precisa la similitud semántica entre frases.

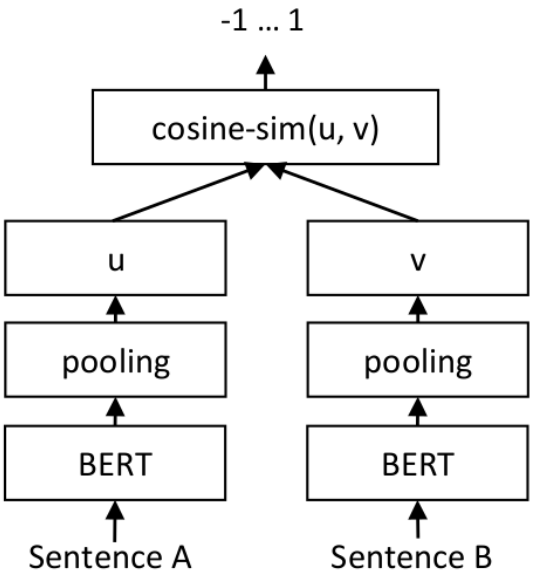

FIGURA 2.3: Estructura de *S-BERT*.

- En 2022 aparece *BERTopic* [8] que combina el modelado de tópicos y los *embeddings* de *S-BERT* para la identificación de temas relevantes en los documentos, su esquema se refleja en la FIGURA 2.4.

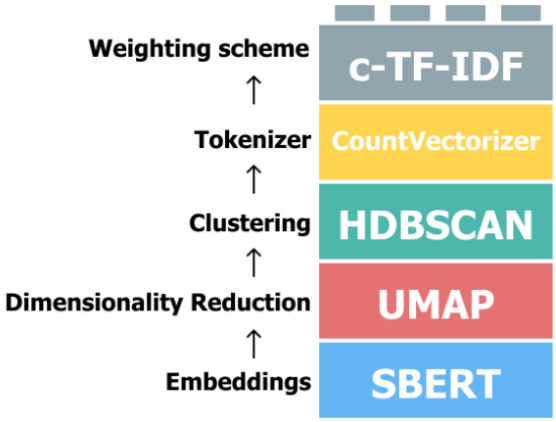

FIGURA 2.4: Estructura de *BERTopic*.

El modelado por tópicos ha sido y es utilizado en una gran variedad de campos. Como aplicaciones se encuentran la identificación de riesgos [9], el impacto de las noticias en las cotizaciones [10], extracción de información específica [11] o análisis de sentimientos [12].

Capítulo 3

Metodología

En este capítulo 3, se describe la metodología utilizada en el proyecto para implementar un sistema de búsqueda semántica y modelado por tópicos de documentos financieros, en particular, descripciones y *earnings calls* de empresas cotizadas. La metodología se divide en cuatro etapas principales: extracción de datos (3.1), preprocesamiento del texto (3.2), extracción de *embeddings* (3.3) y modelado por tópicos (3.4).

Se detallarán las herramientas y técnicas utilizadas en cada una de estas etapas, así como la infraestructura utilizada para implementar todo el sistema en la nube de *Microsoft Azure*[1] (3.5). Además, se presentará una descripción detallada de la *API*[2] (3.5.2) construida para la búsqueda semántica y de la aplicación web (3.5.3) para permitir a los usuarios hacer búsquedas y visualizar los gráficos interactivos del modelado por tópicos.

En resumen, este capítulo proporciona una descripción completa del proceso llevado a cabo para implementar el sistema y permitir a los usuarios buscar y visualizar la información de manera más eficiente y precisa.

3.1 Datos

La importancia de la calidad y cantidad de datos en cualquier proyecto de inteligencia artificial es innegable. Los científicos de datos dedican la mayor

[1] *Microsoft Azure* es una plataforma de computación en la nube que ofrece servicios de infraestructura, plataforma y software como servicio.

[2] Una *API (Application Programming Interface)* es un conjunto de protocolos, herramientas y rutinas para crear aplicaciones de software que permiten a distintos sistemas comunicarse e intercambiar datos entre sí.

parte de su tiempo a la tarea de recopilación y limpieza de datos, lo que demuestra la importancia de estos aspectos. Estos modelos se nutren y aprenden de los datos con los que se les alimenta, por lo que es fundamental garantizar que estos sean suficientes y de buena calidad para que el modelo pueda funcionar de manera efectiva.

En este trabajo, se ha dado especial atención a la búsqueda de datos, con el objetivo de obtener la información adecuada para el proyecto. Se han utilizado tanto datos estructurados como no estructurados, con un enfoque en el no estructurado. Aunque también se han recopilado datos estructurados sobre las compañías, su principal objetivo es proporcionar una información adicional y complementaria al usuario final de la aplicación.

Para llevar a cabo el trabajo, se han obtenido los datos de una fuente confiable y respetada: *Koyfin*, que es una plataforma de investigación financiera que brinda acceso a información detallada sobre compañías cotizadas en bolsa, incluidas sus descripciones y *earnings calls*. Este agregador de datos ofrece una gran cantidad de información, toda la cual obtiene de reputados proveedores de datos como *S&P Capital IQ*.

Se trabaja con varios tipos de datos: descripciones (3.1.1) y *earnings calls* (3.1.2) de empresas.

3.1.1 Descripciones

Las descripciones de empresas son resúmenes escritos que proporcionan información general sobre una compañía, incluyendo su historia, misión, productos o servicios, estructura organizativa y objetivos futuros. Estas descripciones suelen aparecer en los sitios *web* de las compañías, en prospectos de inversión y en otros materiales de marketing.

Las descripciones de empresas son importantes porque proporcionan una visión general de la compañía y ayudan a los inversores y otros interesados a comprender mejor su propósito y estrategia. También pueden ayudar a la compañía a establecer su marca y diferenciarse de sus competidores. Además, pueden ser un indicador de la cultura y los valores de la compañía, lo que puede ser importante para los empleados y otros accionistas.

En general, son una parte importante de la transparencia y la rendición de cuentas de las compañías y deben ser leídas y comprendidas por aquellos que buscan obtener información sobre una compañía en particular.

Obtención de Datos

La información se obtiene a través de una *API* de *Koyfin*. Esta incluye tanto el texto de las descripciones de las empresas como una serie de metadatos[3].

En el CUADRO 3.1, se puede ver con un ejemplo todos los metadatos que se obtienen de las empresas.

Metadato	Valor
name	Tesla, Inc.
ticker	TSLA
trading Currency	USD
exchange	NasdaqGS
company Country	US
trading Country	US
headquarters City	Austin
headquarters State	Texas
headquarters Website	www.tesla.com
year Founded	2003
workers	127.855
sector	Consumer Discretionary
industry	Automobiles
market Cap	357.015,5 M

CUADRO 3.1: Metadatos de descripciones para la empresa *Tesla*.

En cada llamada a la *API*, se obtiene información sobre 10.000 empresas a la vez. Después de obtener esta información, se realiza un formateo ligero para descargarla en formato *CSV*[4] y en formato *Pickle*[5]. Esto permite que la información obtenida se pueda analizar y procesar posteriormente.

Es importante tener en cuenta que, estos datos tienen el inconveniente de que no tienen fecha, es decir, que no se conoce exactamente cuándo se escribió esa descripción y si es fiel a las actividades y valores actuales de la empresa.

[3]Los metadatos son datos que, describen y proporcionan información adicional de las empresas.

[4]*CSV* (por sus siglas en inglés *Comma-Separated Values*) es un formato de archivo común utilizado para almacenar y intercambiar datos tabulares.

[5]El formato *Pickle* es una forma de serializar objetos *Python* en un formato binario compacto que puede ser fácilmente almacenado o transmitido entre diferentes sistemas.

En total se obtienen descripciones de **46.007** empresas, lo que supone aproximadamente 70 *MB*.

3.1.2 *Earnings Calls*

Las *earnings calls* son conferencias realizadas por las compañías para informar a los inversores y analistas financieros, sobre sus resultados financieros trimestrales o anuales. Durante estas conferencias, los ejecutivos de la compañía presentan un resumen de las ganancias, ingresos, gastos, estrategias futuras y responden a preguntas de los participantes.

Estas conferencias son importantes porque permiten a los inversores conocer el desempeño financiero de la compañía y tener una visión más clara de su dirección y perspectivas futuras. Las reacciones a las *earnings calls* pueden tener un impacto significativo en el precio de las acciones de la compañía en el corto plazo.

En general, son una parte importante de la transparencia y comunicación de las compañías con sus inversores y pueden ser un indicador valioso de su salud financiera y perspectivas a largo plazo.

Obtención de Datos

El proceso para obtener estas transcripciones de los *earnings calls*, consta de dos pasos:

1. **Obtención de los identificadores de las transcripciones de cada empresa.**
2. **Descarga del texto y metadatos de las nuevas transcripciones.**

Para el primer paso, se envía una solicitud a una *API* que devuelve los identificadores de todas las transcripciones disponibles de una empresa en particular. Es importante destacar que esta solicitud se realiza de manera asíncrona[6] lo cual permite disminuir el tiempo de ejecución del programa en un 80 % en comparación con la descarga secuencial, pero se limita a 10 peticiones por segundo para evitar saturar el servidor de *Koyfin*.

Además, todos los identificadores descargados se comparan con los últimos que se tenían para obtener dos archivos *CSV*: uno con los identificadores nuevos y otro con todo el historial de estos. De esta manera, se puede realizar

[6]Ejecución de tareas de manera independiente sin necesidad de esperar a que otra tarea termine antes de comenzar.

un seguimiento de los nuevos identificadores que aparecen y también tener un registro completo.

En el segundo paso, una vez obtenidos los identificadores de las nuevas transcripciones, se envía una solicitud a otra *API* para obtener el texto y metadatos de cada uno de ellos. Esta solicitud también se realiza de manera asíncrona, esta vez limitada a 4 peticiones por segundo ya que se están descargando archivos mas grandes que los anteriores. Después de descargar las transcripciones, se realiza un proceso de formateo para actualizar el banco de datos con la versión más reciente de cada transcripción.

Los *earnings calls* tienen unos metadatos específicos que se muestra en el CUADRO 3.2.

Metadato	Valor
transcript Id	2669494
company Id	27444752
announced Date	2022-10-02 16:00:00
event Date Time	2022-10-19 21:30:00
created At	2022-11-01 00:27:20
transcript Title	Tesla, Inc., Q3 2022 Earnings Call, Oct 19, 2022
duration (seconds)	3491.0
key Dev Id	1802897951
event Type	Earnings Calls
collection Type Name	Proofed Copy
presentation Type Name	Final
title	Tesla, Inc. - Q3 2022 Earnings Call

CUADRO 3.2: Metadatos de *earnings calls* para la empresa *Tesla*.

Cada vez que se realiza una transcripción de un *earnings call*, se pueden llevar a cabo varias etapas de edición y corrección para mejorar la precisión y la calidad del texto. Esto puede incluir el uso de herramientas de reconocimiento de voz para convertir la grabación en texto, seguido de una revisión manual por parte de los editores para corregir errores y mejorar la coherencia del texto.

Por lo tanto, en un mismo *earnings call*, puede haber varias versiones de la transcripción, cada una de ellas con diferentes niveles de corrección y calidad, cuya distribución podemos ver en la FIGURA 3.1.

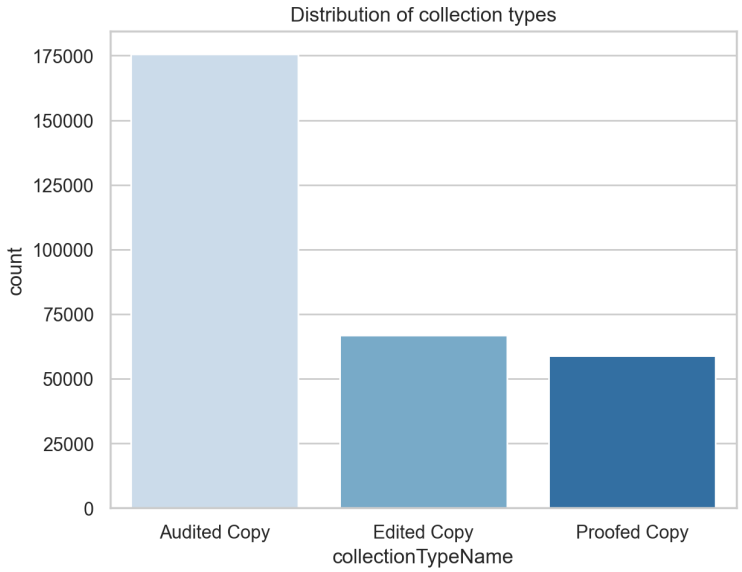

FIGURA 3.1: Distribución de los niveles de corrección y calidad de las transcripciones de los *earnings calls*.

Los tres tipos de niveles de corrección, de menor a mayor precisión en la transcripción son:

1. **Copia corregida (Proofed copy)**
2. **Copia editada (Edited copy)**
3. **Copia auditada (Audited copy)**

La **copia corregida**, es revisada para corregir errores ortográficos y gramaticales, y garantizar que la transcripción sea coherente y fácil de leer. Por lo general, es la versión final antes de su publicación. La **copia editada**, es revisada por un editor que hace correcciones y ajustes para mejorar la legibilidad y la comprensión. A menudo se utiliza para fines de marketing o para publicar en línea. La **copia auditada**, es revisada y corregida por una persona encargada de garantizar que la transcripción sea lo más precisa y completa posible. Por lo general, se utiliza para fines legales y se considera la versión más confiable.

El proceso de descarga y formateo de las transcripciones implica la comparación de cada nueva versión descargada con las versiones anteriores que se han descargado y almacenado en el banco de datos. De esta manera, se puede determinar si una versión más reciente de la transcripción está disponible, y si es así, se descarga y se almacena como la versión más actualizada. Esto garantiza que la información almacenada en el banco de datos sea lo más precisa y actualizada posible.

Durante un *earnings call*, los diferentes participantes hablan y discuten sobre diversos temas, y cada vez que alguien habla, se refleja en el registro, junto con el texto de lo que están diciendo. Este registro se almacena en una estructura de datos en forma de lista, en la que cada elemento de la lista corresponde a una intervención, con el nombre del orador y el texto que se ha dicho, como podemos observar en el CUADRO 3.3.

Speaker Name	Speaker Type	Text
Drew Baglino	Executives	We need to get 300 to 400 terawatt hours cells to accomplish our goal.
Elon Musk	Executives	Yes, there's roughly – to transition to sustainable energy, our rough calculation for both stationary and for vehicles is 300,000 to 400,000 gigawatt hours or 300 to 400 terawatt hours.
Drew Baglino	Executives	So when you're like 1 terawatt, that sounds like a lot, well, it's a lot of terawatt hours to get by.

CUADRO 3.3: Fragmento del *earnings call* de Tesla del *Q3* 2022.

Sin embargo, para poder trabajar con esta información de manera más eficiente, es útil convertir esta lista de intervenciones en una cadena de texto única, que contenga toda la conversación completa. Para hacer esto, se unen todas las intervenciones de un *earnings call* en una sola cadena de texto, eliminando las etiquetas que indican quién está hablando en cada momento. De esta forma, el resultado final es un único texto que contiene toda la conversación de un *earnings call*, que se puede utilizar para hacer análisis y estudios más detallados.

Al contrario que con las descripciones, ahora si que se tienen datos de las fechas en las que se han realizado, por lo que se puede seguir la evolución temporal de los temas de los que hablan las empresas en sus *earnings calls*.

Se puede observar en la FIGURA 3.2, que a medida que pasan los años, la cantidad de información disponible aumenta de forma lineal. El número total de transcripciones de *earnings calls* es de **1.041.938**, que después de filtrar por la última versión de corrección de cada transcripción, nos quedan **325.436** documentos. Estos datos ocupan aproximadamente 44 *GB*.

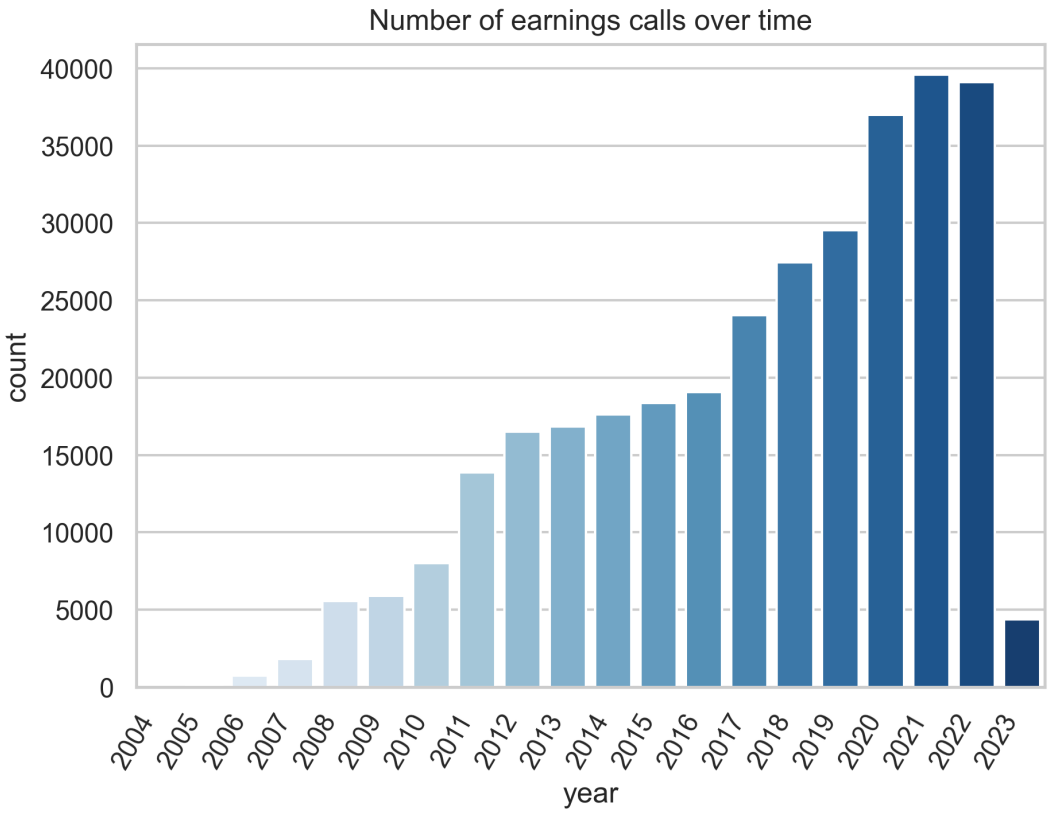

FIGURA 3.2: Evolución temporal del número de *earnings calls*, hasta febrero de 2023.

3.1.3 Universo de Empresas

Como se ha mencionado previamente, se ha trabajado con un universo de 46.007 empresas de todo el mundo, sin hacer distinciones por su capitalización, precio, índice de pertenencia o cualquier otra característica. Debido a la imposibilidad de mencionarlas individualmente, se presentarán estadísticas que resalten los aspectos más relevantes.

País

En términos generales se puede hablar de un universo global, ya que hay empresas de 134 países diferentes. En el CUADRO 3.4, se pueden ver los 15 países con mayor representación.

País	Número compañías
Estados Unidos	7634
China	4958
India	4033
Japón	3366
Canadá	3246
Taiwán	1718
Australia	1625
Corea del Sur	1602
Reino Unido	1458
Hong Kong	1454
Malasia	874
Suecia	837
España	818
Tailandia	719
Alemania	685

CUADRO 3.4: Países más representativos.

Sector

En la FIGURA 3.3, se puede ver que las empresas se distribuyen en 11 sectores.

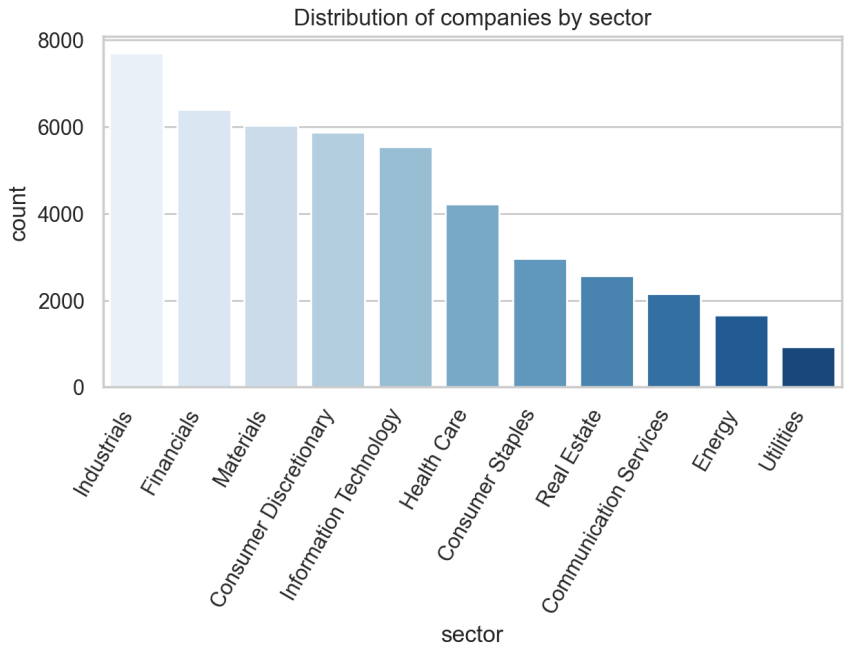

FIGURA 3.3: Distribución de empresas por sector.

El sector industrial es el de mayor relevancia, seguido del financiero y el de materiales. La menor parte de las compañías se dedica al sector de la energía o *utilities*.

Capitalización

Las capitalizaciones de las empresas seleccionadas quedan expuestas en el CUADRO 3.5.

	Capitalización (Millones €)
Media	2.139,7
Desviación	20.842,6
25 %	12,8
50 %	84,3
75 %	580,3
Máximo	2.413.638,2

CUADRO 3.5: Estadísticos sobre la capitalización.

De este análisis se extrae que la gran parte de las empresas son pequeñas o medianas y el pequeño porcentaje restante corresponde a las grandes (aquellas que superan los 10 *Billions* de capitalización).

Año de Fundación

En el siguiente gráfico (FIGURA 3.4) se muestra el número de empresas fundadas en cada año. Es curioso ver como el gráfico no está centrado y esto se debe a que la compañía más antigua de la que se tiene constancia es del año 1308, se trata de la cervecera alemana *Aktienbrauerei Kaufbeuren*. Por otra parte, como puede observarse el mayor número de empresas surgen en 1999 y 2000, sumando entre ambos años, una cantidad de 2.220 nuevas empresas.

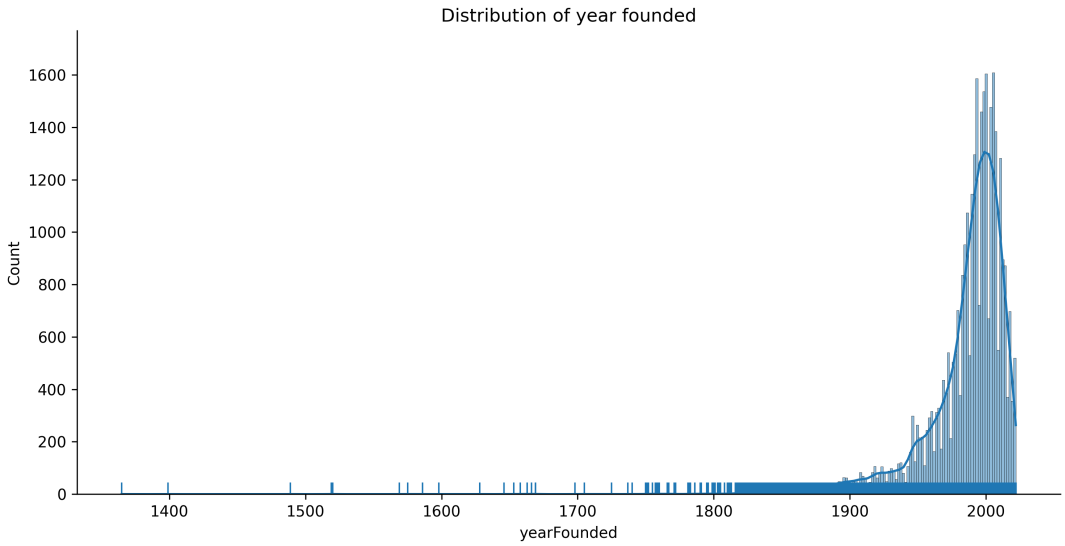

FIGURA 3.4: Distribución del año de fundación de las empresas.

3.2 Preprocesamiento de Texto

El preprocesamiento de texto es una fase esencial para la búsqueda semántica. Consiste en realizar una limpieza básica del texto, para obtener una versión más estructurada y uniforme que facilite su posterior análisis.

En este caso, se llevó a cabo una eliminación de líneas vacías y espacios en blanco adicionales para simplificar el texto y hacerlo más legible. También se estableció la condición que no se podían romper oraciones completas para facilitar la lectura y el análisis. Por último, se eliminaron ciertas cadenas de caracteres no relevantes que podrían interferir en la tarea de búsqueda semántica.

Adicionalmente, se dividió el texto en fragmentos de longitud máxima de 100 palabras, ya que los modelos utilizados suelen tener un límite máximo de *tokens*[7], 256 en este caso. Esto se hizo porque en un texto de una descripción o un *earnings call* se van a tratar varios temas, y al dividirlo en partes más pequeñas, se le esta ayudando a capturar todos los temas de ese texto global.

Los textos de las transcripciones de *earnings calls*, tienen una media de 41.878 palabras y 27.190 de desviación. Por otro lado, los de las descripciones tienen una media de 762 palabras y 450 de desviación. Es decir, que los textos de los *earnings calls* son notablemente mas grandes que lo de las descripciones,

[7]En *NLP*, un *token* se refiere a una secuencia de caracteres (como palabras o signos de puntuación) que se agrupan como una sola unidad para su análisis.

aproximadamente un **5.400 %**. En la FIGURA 3.5, se puede observar las distribuciones de las longitudes de los textos para ambos tipos de documentos.

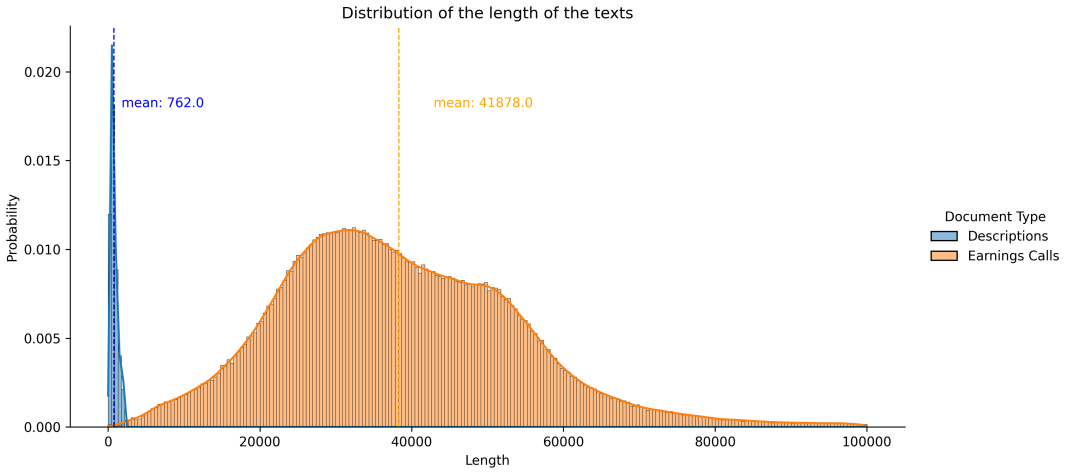

FIGURA 3.5: Comparación de la distribución de las longitudes de los textos de descripciones y *earnings calls*.

3.3 *Embeddings*

La creación de *embeddings* es el proceso de transformar el texto de cada documento en un vector de números, que represente el significado semántico del texto. Para crearlos, se utiliza un modelo preentrenado de lenguaje natural, en este caso, el modelo es el *S-BERT all-MiniLM-L6-v2*.

3.3.1 Algoritmo

El proceso de obtención de *embeddings* se divide en las siguiente partes:

- **Tokenización.**
- **Agregación de información de contexto.**
- **Codificación.**
- **Cálculo del *embedding*.**

En primer lugar, la cadena de texto se divide en *tokens* y se asigna un índice a cada *token* para representarlo numéricamente.

Luego, se agrega información adicional a cada *token* para proveer contexto acerca de su papel en la frase, como la información de segmentación o la información de posición.

Después, cada *token* y su información adicional se pasan a través de la capa de codificación, situada en el interior de la arquitectura *BERT* y está preentrenada en grandes cantidades de datos. La capa de codificación convierte cada *token* en un vector denso de alta dimensión.

Por último, la salida de la capa de codificación se utiliza como el *embedding* de la frase. Este *embedding* se puede utilizar para realizar tareas de procesamiento de lenguaje natural, como la búsqueda y recuperación de información, la clasificación de texto, y construir modelos por tópicos.

3.3.2 Modelo

La razón principal para utilizar un modelo preentrenado como el *all-MiniLM-L6-v2* en lugar de entrenar un modelo desde cero, es que el modelo preentrenado ya ha sido entrenado en grandes cantidades de datos, lo que permite una transferencia de aprendizaje eficiente.

Este modelo, ha sido entrenado en una variedad de corpus de texto que incluyen noticias, artículos de investigación y mucho más. Aunque no se haya entrenado específicamente con datos financieros, este modelo puede capturar características generales del lenguaje natural que se pueden aplicar en una variedad de dominios, incluyendo el financiero.

Además, entrenar un modelo desde cero requiere una cantidad significativa de recursos computacionales y de tiempo, así como un gran conjunto de datos para el entrenamiento. Para hacer esto, se necesitaría un conjunto de datos etiquetados con consultas y sus respectivos documentos relevantes, y utilizar este conjunto de datos para entrenar el modelo con una tarea de clasificación binaria: relevante o no relevante. Pero lamentablemente no se cuenta con datos etiquetados, y etiquetarlos de forma manual llevaría mucho tiempo.

El *all-MiniLM-L6-v2*, se basa en el modelo *nreimers/MiniLM-L6-H384-uncased*, tiene una dimensión de 384 y una longitud máxima de secuencia de 256 *tokens*. El modelo se ha entrenado en un conjunto de datos amplio y diverso de más de mil millones de pares de entrenamiento, y tiene *embeddings* normalizados, lo que lo hace adecuado para funciones de puntuación como el producto escalar, distancia coseno o distancia euclídea.

La decisión de elegir el modelo *all-MiniLM-L6-v2* en lugar del modelo *all-mpnet-base-v2* u otros, se basó en un compromiso entre el rendimiento del modelo y consideraciones prácticas como la velocidad y el tamaño de este.

Aunque el modelo *all-mpnet-base-v2* puede ser mejor a la hora de crear *embeddings* y realizar tareas de búsqueda semántica, también es un modelo mucho más grande, lo que puede dificultar y ralentizar su despliegue en determinados contextos, como en dispositivos móviles o en entornos con pocos recursos.

En cambio, el modelo *all-MiniLM-L6-v2* tiene un tamaño menor, lo que lo hace más rápido y ligero, lo que puede ser más adecuado para ciertas aplicaciones en las que la velocidad y el tamaño del modelo son consideraciones importantes.

Como se puede observar en el CUADRO 3.6, el modelo *all-MiniLM-L6-v2*, respecto al *all-mpnet-base-v2*, es:

- **407 % mejor** en velocidad.
- **425 % mejor** en tamaño del modelo.
- **2,2 % peor** creando embeddings.
- **15,1 % peor** creando en búsqueda semántica.
- **7,7 % peor** de media en rendimiento.

Como se mejora notablemente en velocidad y no se perjudica mucho el rendimiento, es razonable utilizar el modelo *all-MiniLM-L6-v2*.

Model Name	Performance Sentence Embeddings[8]	Performance Semantic Search[9]	Avg. Performance	Speed[10]	Model Size (*MB*)
all-mpnet-base-v2	69.57	57.02	63.30	2800	420
all-MiniLM-L12-v2	68.70	50.82	59.76	7500	120
all-MiniLM-L6-v2	68.06	49.54	58.80	14200	80

CUADRO 3.6: Comparación de modelos preentrenados de *S-BERT*.

[8]Rendimiento medio en la codificación de frases en 14 tareas diversas de varios ámbitos.
[9]Rendimiento en 6 tareas diversas de búsqueda semántica, codificando en textos de hasta 512 *tokens*.
[10]Velocidad de codificación (frases/segundo) en una GPU V100.

3.4 Modelado por Tópicos

Una vez obtenidos los *embeddings* para cada texto de las empresas, se procede a realizar un modelado por tópicos para extraer información relevante y significativa de los textos. El modelado por tópicos es una técnica de procesamiento de lenguaje natural que se utiliza para descubrir patrones latentes en un conjunto de documentos y agruparlos en temas o categorías. La idea es que los documentos que tratan temas similares tendrán palabras en común y, por lo tanto, estarán cerca entre sí en un espacio vectorial.

Se han utilizado únicamente los datos de las descripciones de las 46.007 empresas. Como resultado de aplicar el modelado por tópicos, las empresas se ha agrupado en 149 tópicos.

3.4.1 Algoritmo

Se va a utilizar el algoritmo *BERTopic* para realizar el modelado por tópicos a partir de los *embeddings* obtenidos (3.3). La implementación de *BERTopic* consta de 6 pasos principales:

1. **Cálculo del *embeddings*.**
2. **Reducción de la dimensionalidad.**
3. **Clúster de documentos.**
4. ***Bag-of-words*.**
5. **Representación temática.**
6. **Relevancia marginal máxima.**

En primer lugar, se coienza por convertir los documentos en representaciones numéricas. Aunque existen muchos métodos para hacerlo, se utiliza el modelo ***S-BERT all-MiniLM-L6-v2*** ya calculado anteriormente (3.3).

En segundo lugar, después de convertir los documentos en representaciones numéricas, es necesario **reducir la complejidad dimensional** de estas representaciones. La alta dimensionalidad puede ser un obstáculo para los algoritmos de agrupamiento debido a la *maldición de la dimensionalidad*[11]. Hay

[11]La *maldición de la dimensionalidad* se refiere al aumento exponencial de: la dispersión de los datos y la complejidad computacional, a medida que crece el número de dimensiones de un conjunto de datos, lo que dificulta el análisis y la extracción de información significativa.

diferentes formas de lograr una reducción dimensional, como *PCA*[12], pero *BERTopic* por defecto utiliza *UMAP*[13]. Esta técnica es capaz de preservar cierta parte de la estructura local y global de los datos mientras los simplifica. Mantener esta estructura es crucial, ya que contiene información valiosa sobre la similitud semántica entre los documentos y es esencial para crear agrupaciones precisas.

En tercer lugar, una vez reducidas las representaciones dimensionales, es posible **agrupar los datos**. Para ello, se utiliza *HDBSCAN*[14], una técnica de clustering basada en la densidad de los datos. Este, tiene la capacidad de identificar agrupaciones de diferentes formas y también es capaz de detectar valores atípicos, evitando así la asignación forzada de documentos a clústers que no les corresponden. Esto mejorará la calidad de la representación temática final, ya que disminuirá la cantidad de ruido en los datos.

En cuarto lugar, una vez se han agrupado los documentos en clústers con la técnica *HDBSCAN*, es necesario seleccionar una técnica para **representar temáticamente cada grupo**. Para ello, se utiliza el *Bag-of-words*[15], que consiste en combinar todos los documentos de un grupo en un solo documento más largo, que representa el clúster. Luego, se cuenta la frecuencia con la que aparece cada palabra en el documento compuesto. De esta manera, se obtiene una representación llamada *Bag-of-words* en la que se indica la frecuencia de cada palabra en el clúster.

Esta representación es importante porque se centra en el nivel de tema y no en el nivel de documento, lo que significa que se está interesado en las palabras que describen el tema general del clúster en lugar de los documentos individuales. Además, no se hacen suposiciones sobre la estructura de los clústers y la representación presenta normalización *L1*[16] para tener en cuenta

[12] El *PCA (Principal Component Analysis)* es una técnica de reducción de la dimensionalidad que transforma los datos de alta dimensión en un espacio de menor dimensión preservando la información más importante.

[13] *UMAP (Uniform Manifold Approximation and Projection)* es una técnica no lineal de reducción de la dimensionalidad que mapea datos de alta dimensión en un espacio de baja dimensión preservando tanto la estructura global como la local.

[14] *HDBSCAN (Hierarchical Density-Based Spatial Clustering of Applications with Noise)* es un algoritmo de agrupación basado en la densidad que puede descubrir agrupaciones de formas y densidades variables en los datos, al tiempo que identifica puntos de ruido.

[15] El modelo *Bag-of-words* es una forma de representar los datos de texto como una colección de recuentos de palabras, ignorando la gramática y el orden de las palabras, pero capturando la frecuencia de cada palabra en un documento.

[16] La normalización *L1* es una técnica de normalización que escala los valores de un vector de modo que la suma de sus valores absolutos sea igual a 1, lo que facilita la comparación de la importancia relativa de diferentes características.

los clústers de diferente tamaño.

En quinto lugar, se encuentra el paso de la **representación temática**, que trata de identificar las características que diferencian a un clúster de otro. Es decir, se quiere conocer qué palabras son comunes en un clúster y no tanto en los demás. Para hacer esto, se modifica el algoritmo *TF-IDF*[17] para que tenga en cuenta los temas o los clústers en lugar de los documentos individuales.

El proceso consiste en tratar todos los documentos de un mismo clúster como un solo documento y aplicar *TF-IDF* en él. De esta forma, se obtiene la puntuación de importancia de las palabras dentro de un clúster en particular. Las palabras más importantes dentro de un clúster son las que mejor describen el tema al que pertenece. Este modelo se llama *c-TF-IDF* (*TF-IDF* basado en clases). La ecuación del algoritmo *c-TF-IDF* es:

$$c\text{-}TF\text{-}IDF_{x,c} = TF_{x,c} \cdot IDF_{x,c} \tag{3.1}$$

Donde:

$c\text{-}TF\text{-}IDF_{x,c}$ es la puntuación de importancia de la palabra x en la clase c. $TF_{x,c}$ es la frecuencia de la palabra x en la clase c.

$IDF_{x,c}$ es la representación *IDF* basada en clases de la palabra x en la clase c, dada por:

$$IDF_{x,c} = \log \frac{N_c}{n_x} \tag{3.2}$$

Donde:

N_c es el número total de documentos en la clase c.

n_x es el número de documentos en la clase c que contienen la palabra x.

En sexto y último lugar, la Relevancia Marginal Máxima[18] es una técnica utilizada para **mejorar la coherencia y diversificación de las palabras en una representación temática**. A través de esta técnica, se busca encontrar las palabras más coherentes que describan un tema similar, sin que haya demasiado

[17] *TF-IDF (Term Frequency-Inverse Document Frequency)* es un sistema de ponderación que mide la importancia de cada palabra en un documento aumentando la frecuencia de la palabra por la frecuencia inversa de su aparición en el documento.

[18] *Maximal Marginal Relevance (MMR)* es un método de clasificación de algoritmos que busca maximizar la diversidad y relevancia de los resultados obtenidos.

solapamiento entre ellas. De este modo, se pueden eliminar aquellas palabras que no aportan al tema y mejorar la representación de este.

Además, esta técnica también se puede utilizar para reducir el número de sinónimos en la representación temática. A veces, varias formas de una misma palabra pueden aparecer en la representación, y para mejorar la diversidad entre las palabras, se puede utilizar este algoritmo para encontrar aquellas que son similares y a la vez diferentes entre sí.

3.4.2 Herramientas Gráficas

Una de las grandes ventajas de utilizar el modelado por tópicos es que se pueden visualizar los resultados de una manera muy intuitiva y fácil de interpretar. Se proponen varias herramientas de visualización para explorar los tópicos y los documentos asociados a cada uno de ellos.

La primera es una herramienta de representación gráfica de los documentos en 3D (FIGURA 3.6), que permite visualizar en un espacio tridimensional los *embeddings* correspondientes a cada texto de empresa, de tal forma que las empresas cuyos textos tienen características similares estarán agrupadas en áreas cercanas. Esta herramienta es muy útil para identificar patrones y relaciones entre los textos de las empresas y su representación visual en 3D facilita la identificación de áreas o clústers en los que se agrupan empresas con características similares.

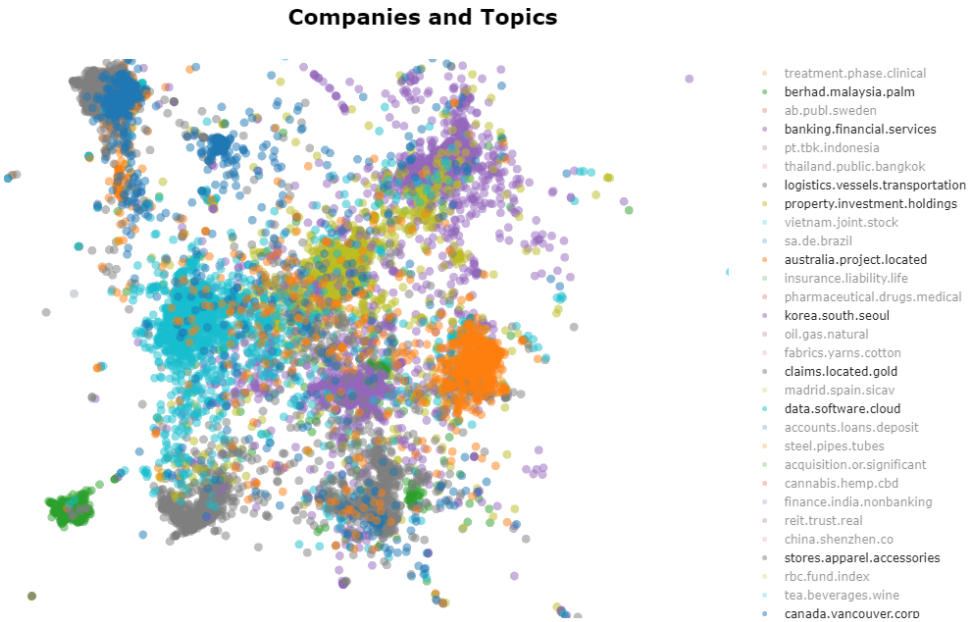

FIGURA 3.6: Herramienta de representación de los documentos en 3D.

Otra herramienta interesante, es la visualización de la frecuencia de aparición de los tópicos en diferentes agrupamientos de los datos. Esta herramienta, permite analizar la distribución de los tópicos en los diferentes grupos: por país, mercado, sector o industria (FIGURA 3.7). De esta manera, se puede obtener información adicional sobre los patrones temáticos que existen en los datos.

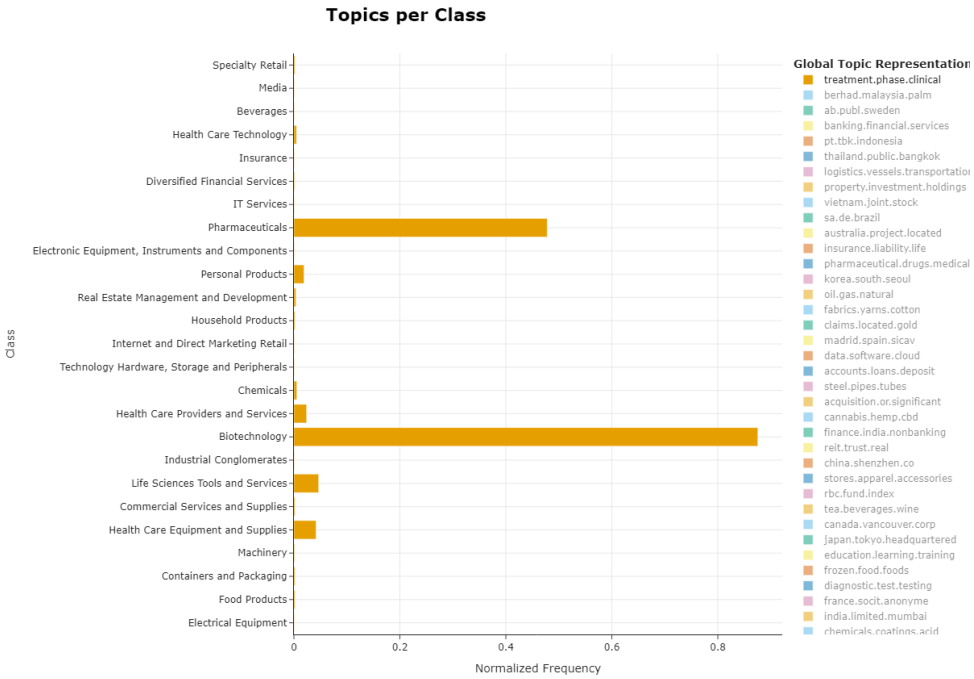

FIGURA 3.7: Herramienta de representación de tópicos por industria.

Al analizar la distribución de los tópicos en diferentes grupos, es posible entender mejor las tendencias y patrones en los datos. Por ejemplo, se puede utilizar esta herramienta para analizar si ciertos tópicos son más comunes en ciertos sectores, o en empresas de un país en particular. Esto permite tomar decisiones de inversión más informadas y precisas, y ayuda a identificar oportunidades interesantes.

3.5 Despliegue *Cloud*

Se decidió implementar todo en *Microsoft Azure*, para automatizar el proceso de extracción de datos (3.1), preprocesamiento del texto (3.2), extracción de *embeddings* (3.3) y modelado por tópicos (3.4). Además, de construir y desplegar una *API* (3.5.2) para acceder a los *embeddings* de manera rápida, y servir estos a una aplicación web (3.5.3) desde la cual los usuarios podrán realizar

búsquedas semánticas y visualizar las herramientas gráficas del modelado por tópicos.

El motivo principal de migrar todo a la nube y automatizarlo, es debido a que a medida que pase el tiempo aumentara el número de datos disponibles: habrá nuevas *earnings calls* de empresas, y se fundarán nuevas empresas de las que podremos obtener sus descripciones.

En la FIGURA 3.8, se puede ver el diagrama completo de la arquitectura *cloud*, la cual se ira explicando por partes en los siguientes apartados (3.5.1, 3.5.2 y 3.5.3).

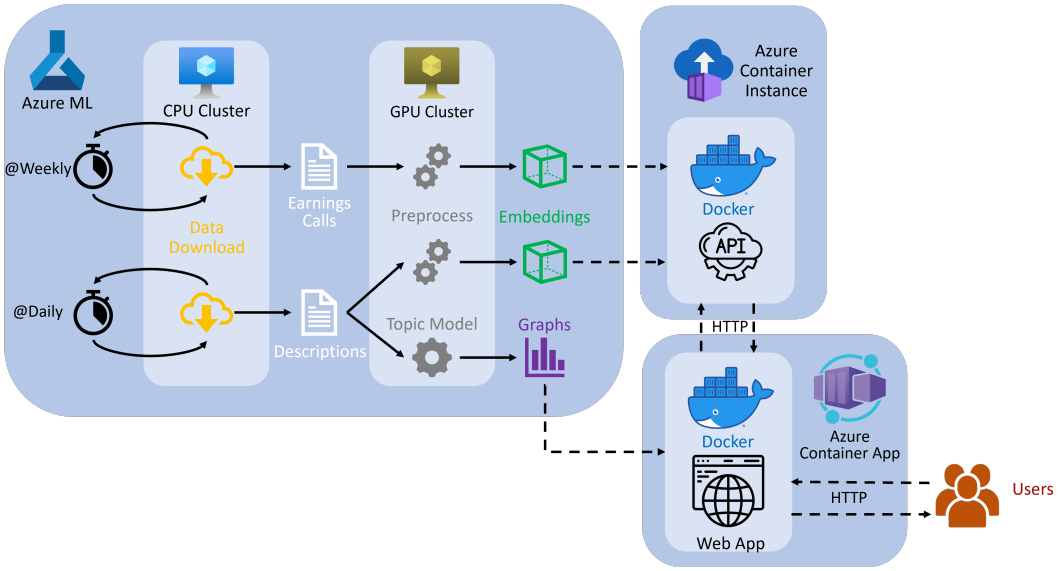

FIGURA 3.8: Diagrama completo de la arquitectura *cloud*.

3.5.1 *Azure Machine Learning*

Azure Machine Learning es una plataforma de *Microsoft* que permite la creación y ejecución de modelos de aprendizaje automático a gran escala. Uno de los componentes clave de la plataforma son los *pipelines*, que permiten automatizar y orquestar el flujo de trabajo de un proyecto de aprendizaje automático.

Para automatizar el proceso de análisis de datos, se han implementado 6 *pipelines* divididos en 3 grupos principales: descripciones, *earnings calls* y modelado por tópicos (FIGURA 3.9).

El primer grupo, **descripciones**, consta de 2 *pipelines*. El primer *pipeline* se encarga de extraer los datos de descripciones, mientras que el segundo realiza

el preprocesamiento y cálculo de *embeddings* de las descripciones. Estos 2 *pipelines*, se ejecutan diariamente de lunes a viernes a las 23:15 y 23:45 *UTC*[19] respectivamente. Este grupo tarda en total 20 minutos en ejecutarse, 7 para los datos y 13 para el preprocesamiento y *embeddings*.

El segundo grupo, ***earnings calls***, consta de 3 *pipelines*. El primer *pipeline* se encarga de extraer los identificadores de los *earnings calls*, el segundo *pipeline* se encarga de descargar los datos de los nuevos *earnings calls*, y el tercero realiza el preprocesamiento y cálculo de *embeddings* de estos. Estos datos son más complejos y voluminosos, por lo que se procesan únicamente los domingos para evitar saturar los recursos computacionales. Se ejecutan a las 3:30, 12:30 y 14:30 *UTC* respectivamente. Este grupo tarda en total 160 minutos en ejecutarse, 80 para los identificadores, 20 para la descarga y 60 para el preprocesamiento y *embeddings*.

El tercer grupo, **modelado por tópicos**, consta de un único *pipeline*. Este, se encarga de crear el modelo *BERTopic* y generar las herramientas gráficas. Se lleva a cabo semanalmente los sábados a las 3:30 *UTC* y tarda 50 minutos aproximadamente.

Cabe destacar que para los *pipelines* que calculan los *embeddings* se utilizan clústers con *GPU*[20]. Lo cual permite disminuir exponencialmente el tiempo de ejecución de las tareas, frente a calcularlos con *CPU*[21].

Una vez que se han generado los modelos con los textos y *embeddings* de las descripciones y *earnings calls*, estos son registrados en *Azure Machine Learning* para poder ser utilizados en la búsqueda semántica de información. Además, se registran datos y gráficas de la parte de modelado por tópicos.

[19] *UTC* son las siglas de Tiempo Universal Coordinado y es la principal norma horaria utilizada para regular los relojes y el cronometraje en todo el mundo.

[20] Una *GPU*, o unidad de procesamiento gráfico, es un circuito electrónico especializado diseñado para procesar y renderizar gráficos e imágenes con rapidez y eficacia.

[21] Una *CPU*, o Unidad Central de Procesamiento, es el componente principal de un ordenador que realiza la mayoría de las tareas de procesamiento y ejecuta las instrucciones de un programa informático.

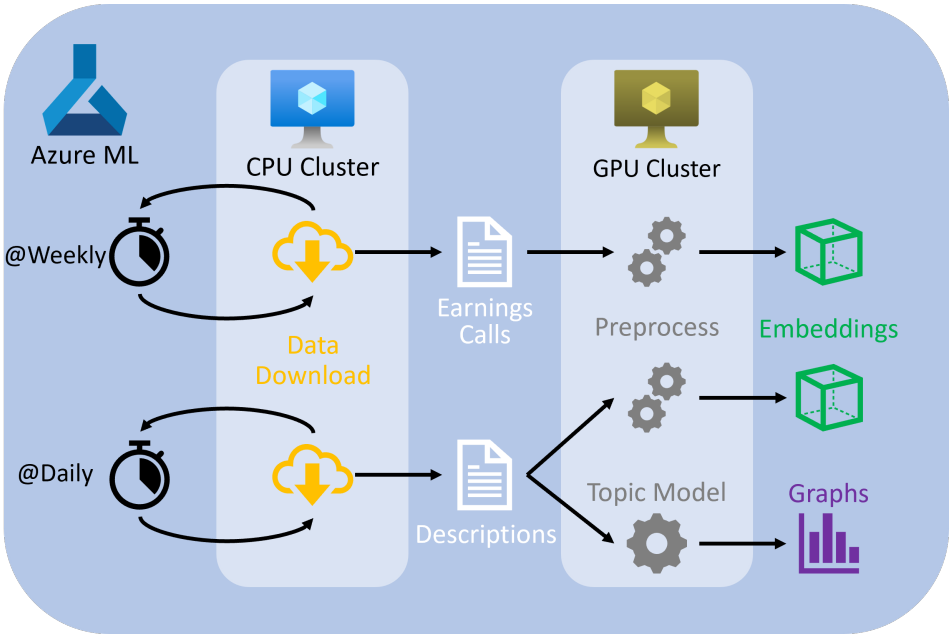

FIGURA 3.9: Diagrama de la arquitectura de *Azure Machine Learning*.

3.5.2 *API*

Para la parte de búsqueda semántica, se han registrado los modelos con los textos y *embeddings* de las descripciones y *earnings calls* en *Azure*. Estos modelos en conjunto ocupan un tamaño considerable, aproximadamente 60 *GB* y en constante crecimiento a medida que haya más datos. Para que un usuario en internet pueda acceder a estos modelos de manera rápida, se ha construido una *API* con *FastAPI*[22].

La *API* está desplegada en un contenedor con *Docker*[23], en *Azure Container Instances*[24]. Este contenedor, cuenta con 2 *CPUs* y 6 *GBs* de *RAM*, que utiliza para cargar los modelos en memoria una sola vez y permitir realizar llamadas a ellos con tiempos de respuesta de décimas de segundo. De esta manera, los usuarios pueden hacer consultas a través de la *API* y obtener resultados de forma rápida y eficiente sin tener que preocuparse por la complejidad técnica de la búsqueda semántica (FIGURA 3.10).

[22] *FastAPI* es un entorno *web* moderno y rápido (de alto rendimiento) para construir *APIs* con *Python*, utilizando el paradigma de programación asíncrona.

[23] *Docker* es una plataforma de código abierto que simplifica el proceso de creación, despliegue y ejecución de aplicaciones al permitir que se ejecuten en contenedores que pueden moverse fácilmente entre distintos entornos.

[24] *Azure Container Instances (ACI)* es un servicio basado en la nube de *Microsoft* que proporciona una forma rápida y sencilla de ejecutar contenedores *Docker* en *Azure* sin gestionar la infraestructura subyacente.

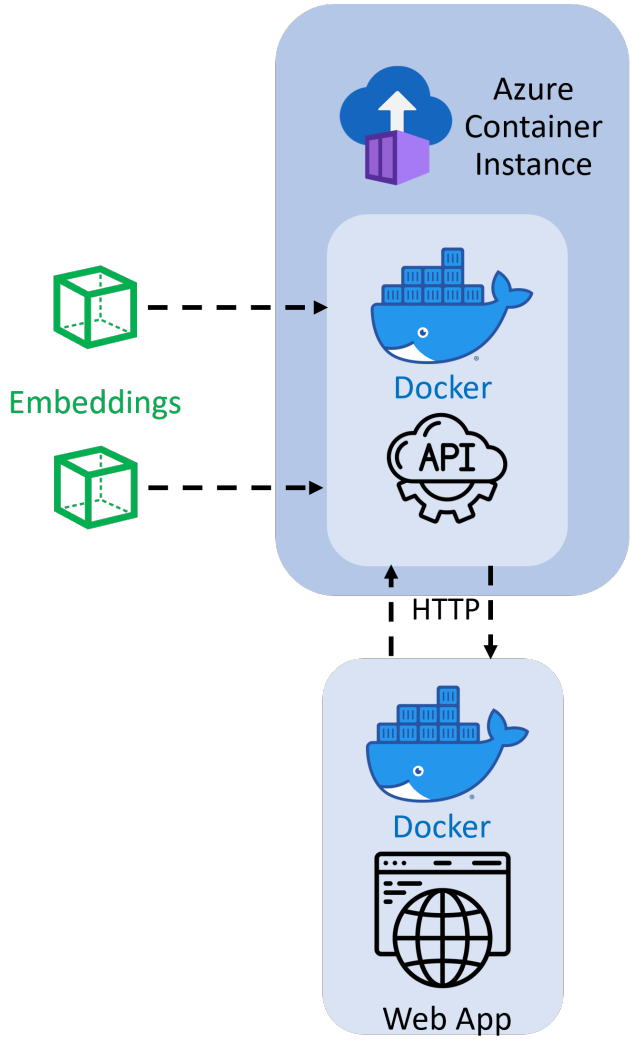

FIGURA 3.10: Diagrama de la arquitectura de la *API*.

La *API* funciona de la siguiente forma: cuando un usuario escribe un texto en la aplicación *web*, esta hace una llamada a la *API*. Ese texto se convierte en un vector de *embedding* utilizando el modelo de codificación de texto previamente entrenado (*all-MiniLM-L6-v2*).

Luego, ese vector de *embedding* se normaliza. Esto es importante para que todos los vectores de *embedding* tengan una magnitud unitaria y, por lo tanto, puedan ser comparados directamente utilizando la función de similitud de producto escalar, que funciona mas rápido que la similitud coseno.

Por último, se compara el vector de entrada con todos, y se devuelven los N documentos más parecidos en forma de diccionarios, donde cada diccionario contiene el contenido del documento y la puntuación de relevancia asociado.

3.5.3 Aplicación *web*

Arquitectura

Para que usuarios puedan realizar la búsqueda semántica y visualizar gráficos interactivos del modelado por tópicos, se construye una aplicación *web*. Para ello, se ha creado un contenedor de *Docker* con la aplicación *web* y se ha desplegado en *Azure Container Apps*[25]. Este contenedor tiene acceso a la *API* previamente construida y se comunica con ella para realizar las búsquedas semánticas. Además, también accede a los datos y gráficas de los modelos de tópicos, registrados en *Azure* para mostrarlos en la interfaz de usuario (FIGURA 3.11).

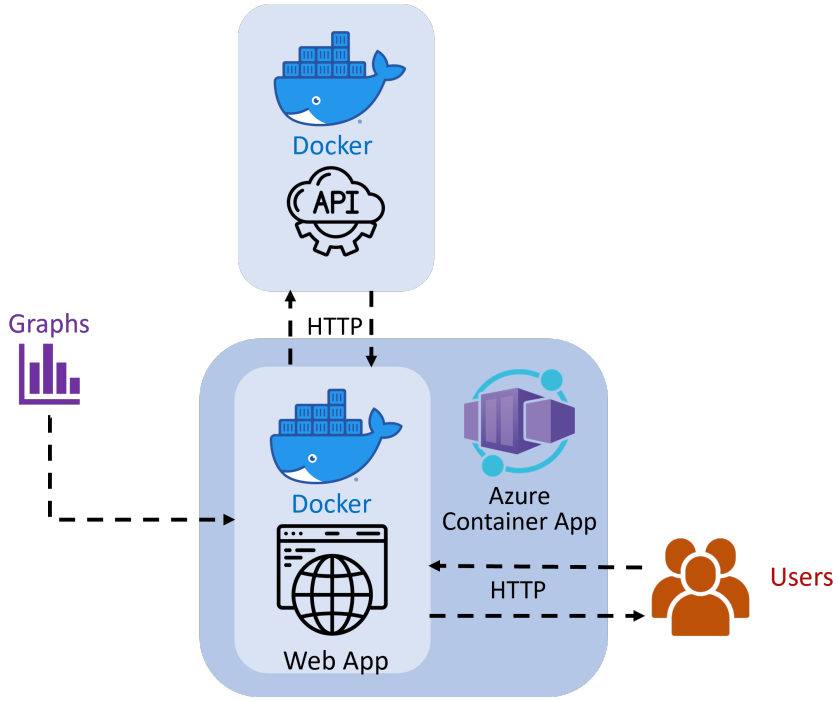

FIGURA 3.11: Diagrama de la arquitectura de la aplicación *web*.

Azure Container Apps permite desplegar contenedores de forma rápida y sencilla, lo que simplifica el proceso de implementación y mantenimiento de la aplicación. Además, ofrece funcionalidades adicionales, como auto escalamiento y alta disponibilidad, lo que garantiza que la aplicación esté siempre disponible para los usuarios.

[25] *Azure Container Apps* es un servicio *serverless* de *Microsoft* que permite a los desarrolladores desplegar y gestionar fácilmente aplicaciones en contenedores como servicios totalmente gestionados en *Azure*.

Interfaz

La manera que tiene el usuario de interactuar con el modelo de lenguaje natural es mediante una aplicación *web* hecha en *Flask*[26] y programada en *HTML*[27], *CSS*[28] y *JavaScript*[29]. Para familiarizar al usuario con la aplicación, se va a realizar una explicación general de cada una de sus partes.

La FIGURA 3.12 corresponde con la página principal, que consta de dos elementos: el buscador y los gráficos. El **buscador** es a través del cual los usuarios puedan realizar las búsquedas semánticas, y el botón de **gráficos** (*charts*) les llevara a una página con herramientas gráficas interactivas.

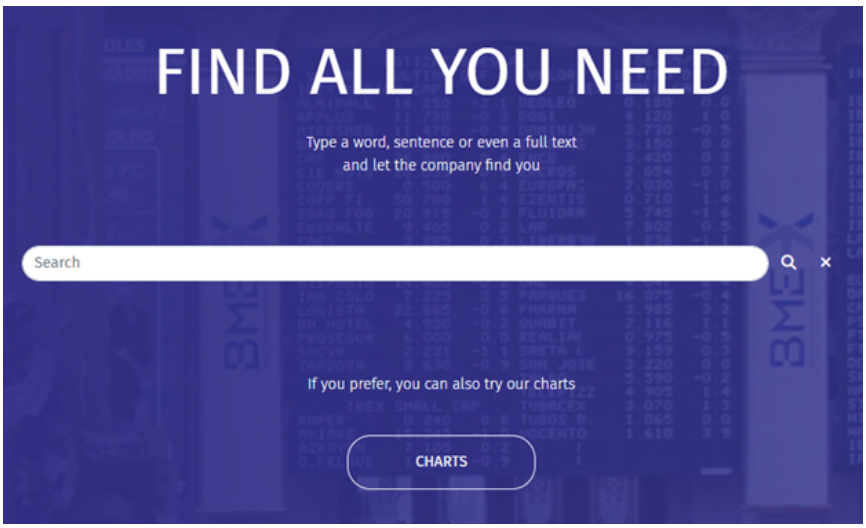

FIGURA 3.12: Página principal de la aplicación *web*.

A continuación (FIGURA 3.13), se va a simular el caso en el que el usuario comienza una búsqueda introduciendo *medioambiente y sostenibilidad*. Un formulario es enviado mediante el cual se redirigirá a la misma ruta, pero con unos parámetros contenidos en la *URL* por lo que la apariencia de la *web* cambia.

[26] *Flask* es una *framework* de *Python* utilizado para desarrollar aplicaciones *web* ligeras y escalables, que proporciona herramientas y funcionalidades básicas y flexibilidad para añadir extensiones según las necesidades del proyecto.

[27] *HTML (HyperText Markup Language)*, hace referencia al lenguaje de marcado para la elaboración de páginas web.

[28] *CSS (Cascading Style Sheets)* es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en *HTML*.

[29] Lenguaje de programación orientado a objetos comúnmente utilizado para crear efectos interactivos en páginas *web*.

FIGURA 3.13: Resultados de la búsqueda.

En la parte superior izquierda hay un botón que dirige a la parte de herramientas gráficas y en la barra de buscador se mantiene escrito lo que el usuario introdujo. En la parte de abajo del buscador, aparece el texto *Searching for: environment and sustainability*, esto se debe a que las descripciones y los *earnings calls* con los que se crearon los *embeddings* del modelo estaban en inglés.

Por ello, se ha incorporado un traductor en línea que detecta el idioma en el que el usuario escribe y lo traduce al inglés. Esto se deja visible para que el usuario sea consciente de la traducción que se ha realizado. Solo desaparecerá cuando se haya superado un límite de palabras, ya que se considera que no tiene sentido que aparezca un párrafo muy largo traducido.

Seguidamente se muestran todas las compañías relacionadas con las palabras introducidas en formato de tarjeta. Este formato es el que se muestra en la FIGURA 3.14.

FIGURA 3.14: Formato de la tarjeta.

En la parte superior izquierda se encuentra, expresada en porcentaje, la puntuación que el modelo otorga a esa compañía. Las tarjetas salen ordenadas en base a esta puntuación, siendo la primera la más alta.

En la parte superior derecha puede verse un botón conteniendo *similar companies*. La finalidad de este botón, es ver cuáles son las empresas que más se asemejan a ese texto de esa compañía.

En la parte central tienen el nombre de la compañía y entre paréntesis su *ticker*. El nombre contiene un hipervínculo a la página *web* oficial de la compañía, que es el sitio donde podrá obtener la información más completa sobre ella.

Justo debajo, pueden ver una serie de iconos con palabras o números. Estos iconos representan, comenzando desde la izquierda:

- **Divisa**
- **País**
- **Mercado**
- **Sector**
- **Industria**
- **Capitalización**
- **Tipo de documento (descripción o *earnings call*)**

En la FIGURA 3.13, se observa en primera posición una tarjeta del tipo *earnings call*. La estructura es la misma que la de las descripciones (FIGURA 3.14),

salvo que la parte central de la tarjeta contiene el nombre del *earnings call* y su correspondiente fecha.

Por otro lado, en la parte de herramientas gráficas (FIGURA 3.15), aparecen en la barra superior todos los gráficos disponibles.

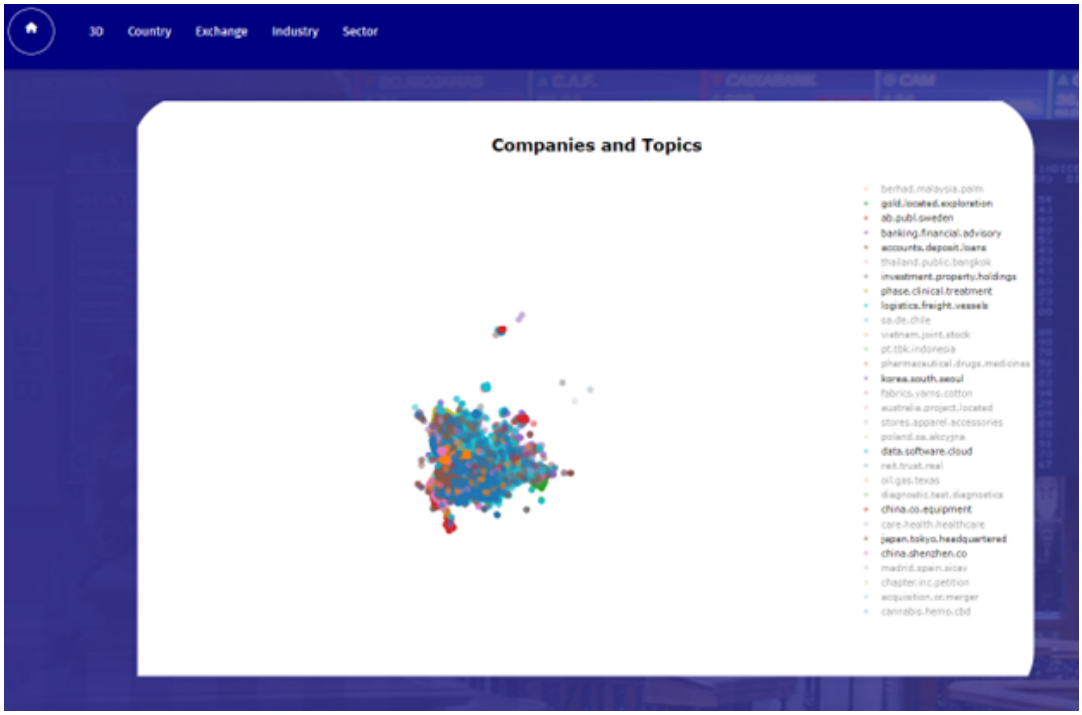

FIGURA 3.15: Página con las herramientas gráficas, se muestra el gráfico 3D.

Por defecto, el que se muestra es el 3D, pero también se pueden visualizar los que muestran la distribución de los tópicos por grupo (país, mercado, sector o industria).

Capítulo 4

Resultados

4.1 Búsqueda Semántica

Se ha puesto a prueba el modelo mediante 60 *inputs* distintos, sobre temas que pudieran tener relación con empresas cotizadas. Y de cada tema a analizar, 10 datos de entrada distintos.

Estos datos de entrada empleados se han diferenciado en 6 temas principales: «Biotecnología», «Inteligencia Artificial», «Ciberseguridad», «Industria Automovilística», «Energía Sostenible» y «Criptodivisas».

En las siguientes FIGURAS (4.1, 4.2, 4.4, 4.3 y 4.5), se muestran ejemplos de los tipos de datos usados para cada tópico y su primer resultado incluyendo el *rating*:

- Un tema principal y definición del mismo.

Cybersecurity	**Cybersecurity refers to the practice of protecting devices, networks, and sensitive information from unauthorized access or attack by malicious actors, such as hackers, cybercriminals, and state-sponsored actors.**
0.72 - Booz Allen Hamilton Holding Corporation Cybersecurity is a really broad industry term at its fundamental, it's the practice of defending computer systems and electronic devices from malicious attacks. What's been interesting with cybersecurity, and we've been in this game for many, many years, is how it's evolved. In the early days, cybersecurity was something that was very much of a compliance-based exercise.	0.83 - Booz Allen Hamilton Holding Corporation Cybersecurity is a really broad industry term at its fundamental, it's the practice of defending computer systems and electronic devices from malicious attacks. What's been interesting with cybersecurity, and we've been in this game for many, many years, is how it's evolved. In the early days, cybersecurity was something that was very much of a compliance-based exercise.

FIGURA 4.1: Ejemplo «Ciberseguridad». Tema, definición y sus primeros resultados.

- Dos subtemas específicos y sus respectivas definiciones.

Quantum Cryptography	**Quantum Cryptography is a field of study that combines principles of quantum mechanics with cryptography to create unbreakable encryption that is not susceptible to hacking or eavesdropping.**
0.68 - Toshiba Corporation It is said is that when the Quantum computer is accomplished and become available, then the current mathematical type of encryption technology will easily be breached and broken. And we are the leading manufacturer of quantum cryptography, a communication method that prevents the fact of this type of encryption. We are convinced that the time will also come when our communications will be protected by the quantum. But the technology of quantum communication does not stop there.	0.76 - Toshiba Corporation It is said is that when the Quantum computer is accomplished and become available, then the current mathematical type of encryption technology will easily be breached and broken. And we are the leading manufacturer of quantum cryptography, a communication method that prevents the fact of this type of encryption. We are convinced that the time will also come when our communications will be protected by the quantum. But the technology of quantum communication does not stop there.

FIGURA 4.2: Primer ejemplo «Ciberseguridad». Subtema, definición y sus primeros resultados.

Cyber Insurance	**Cyber Insurance is a type of insurance policy that is designed to help businesses and individuals protect against Internet-based risks, such as data breaches, cyber attacks, and other types of online threats.**
0.84 - Arch Capital Group Ltd . Cyber insurance has become increasingly important to our insurers globally, and we have substantially increased our support because quite simply, we believe that today's cyber market has changed for the better.The most important development over the past several quarters is that the alignment between clients and insurance companies have significantly	0.84 - Beazley plc And this is a huge commitment that we're making as a team and as a company, which is we want to give the best client experience on cyber. And I don't say cyber insurance on purpose. I think it starts from when we first received a quote and a submission through the policy life cycle through to a client that wants to help with other services and risk management

FIGURA 4.3: Segundo ejemplo «Ciberseguridad». Subtema, definición y sus primeros resultados.

- Dos titulares sobre noticias y sus resúmenes, relativo a ambos subtemas.

New Quantum Encryption System Takes Security to Unprecedented Levels	**Researchers at the National Institute of Standards and Technology (NIST) have developed a new quantum encryption system that is able to protect data from even the most advanced cyber attacks. The new system, which is based on "entangled photons," ensures that any attempt to intercept or eavesdrop on the communication will automatically change the state of the photons, thereby alerting the sender and rendering the message unintelligible to the interceptor. The research was published in the journal Nature Communications.**
0.75 - International Business Machines Corporation This will help us move forward towards our road map to deliver 1,000-plus qubit system next year and a 4,000-plus qubit system in 2025.One of the implications of quantum computing will be the need to change how information is encrypted. We are proud that technology developed by IBM and our collaborators has been selected by NIST as the basis of the next generation of quantum-safe encryption protocols.In another example of innovation, our new z16 system became generally available in the second quarter.	0.71 - International Business Machines Corporation This will help us move forward towards our road map to deliver 1,000-plus qubit system next year and a 4,000-plus qubit system in 2025.One of the implications of quantum computing will be the need to change how information is encrypted. We are proud that technology developed by IBM and our collaborators has been selected by NIST as the basis of the next generation of quantum-safe encryption protocols.In another example of innovation, our new z16 system became generally available in the second quarter.

FIGURA 4.4: Primer ejemplo «Ciberseguridad». Titular, noticia del subtema y sus primeros resultados.

Cyber insurers begin to count the cost of ransomware	**Cyber insurers are counting the cost of recent ransomware attacks, which have affected a growing number of businesses worldwide. Insurance companies are facing rising payouts, as the frequency and severity of cyber attacks continue to increase. A recent report from insurance company Hiscox found that cyber crime is now the second most reported economic crime, after theft.**
0.76 - American International Group, Inc. You have to look at ransomwares today, but like what's next and what industries have systemic impact and trying to do the best we can to make sure that we understand the value that we're delivering to our clients in terms of risk assessment.I think that's a big part, is that if you -- you have companies asking sometimes like what's the value of purchasing the cyber	0.82 - Fortinet, Inc. We met with -- our CFO met with a cyber insurance company -- insurance company that does cyber. And basically, he told -- the insurance company told them that they turn down 95% of the policy request they receive for cyber insurance.That's not a cyber insurance business, by the way. It's not -- but they can't -- they're trying to understand the risk of -- they

FIGURA 4.5: Segundo ejemplo «Ciberseguridad». Titular, noticia del subtema y sus primeros resultados.

Se ha conseguido un buscador semántico en informes financieros funcional, dado que el rango de *ratings* conseguido con todos los tópicos anteriormente mencionados varía entre rangos de 0.50 y 0.84. Valores del todo aceptables, al haber encontrado coincidencias realmente significativas y útiles a pesar de la singularidad de los *inputs*.

Los mejores resultados se consiguen no siendo extremadamente extensos o inconexos en la información dada como *input*. Las noticias y sus titulares consiguen un *rating* algo inferior a los temas y sus definiciones, debido a la gran cantidad de palabras dispares que quizás aportan confusión.

El algoritmo detecta con rapidez y facilidad los documentos de empresas en los que aparece el tópico introducido como *input*, independientemente de la profundidad o complejidad de este y sin la necesidad de ser extremadamente explícito en lo que se pretende buscar, puesto que la red interpreta con una gran precisión lo que deseamos localizar. Intentar definir con excesiva exactitud lo que buscamos, utilizando definiciones, no siempre consigue un mayor *rating* en comparación con utilizar palabras clave sueltas. En muchos casos se solapan los *outputs* recibidos. Esto demuestra que la red neuronal realiza correctamente su función, interpretando el texto en su conjunto con facilidad.

Se puede encontrar que el algoritmo muestre varias veces la misma empresa como *output*, dado un mismo *input*, al poder encontrarse documentos distintos de la misma empresa en los que se trata el mismo tópico en cuestión.

Aunque las noticias puedan incluir explícitamente nombres de empresa cotizadas, el algoritmo no se centra únicamente en mostrar estas, sino que dependiendo de la relación que se tenga con el tópico, mostrará todas las empresas relacionadas. Pudiendo incluso obtener mayor *rating* en empresas distintas a las explícitamente nombradas en el *input* como noticia.

El algoritmo también muestra empresas que pueden no tener ninguna relación actual con el *input* introducido. Esto es debido a que quizás en alguno de los documentos de la empresa, se discutiera el tópico en cuestión, aunque no tengan relación directa con ello. Por lo que no solo detecta empresas que tengan que ver directamente con el tópico, sino también empresas que hayan discutido temas relacionados, independientemente de que hayan sido ya implementados, lo estén discutiendo de cara al futuro de la compañía, o únicamente hayan sido mencionados en alguno de sus informes por cualquier otro motivo.

4.2 Modelado por Tópicos

Al haberse alimentado el modelo únicamente mediante las descripciones de las empresas, se ha observado que este ha decidido segmentar la base de datos de 46 mil empresas, en 149 tópicos distintos. Éstos parecen desglosarse en tres categorías principales, predominando de la primera:

- Diferenciando por **sectores o industrias**, véase: «data.software.cloud» (FIGURA 4.6), «biotech.biosciences.biotechnology», «green.energy.inc», «banking.financial.services», «information.smart.technology» o «blockchain.bitcoin.cryptocurrency».

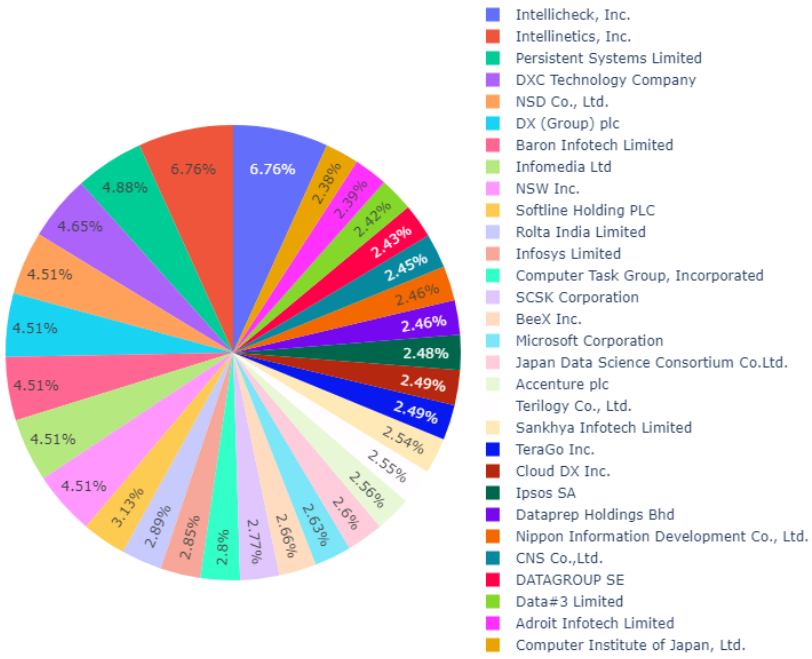

FIGURA 4.6: Tópico «data.software.cloud» y sus 30 compañías más representativas.

- Diferenciando por **zonas geográficas o ciudades**, véase: «ag.germany.switzerland» (FIGURA 4.7), «spa.italy.milan», «korea.south.seoul », «saudi.arabia.riyadh», «asa.norway.denmark» o «california.san.inc».

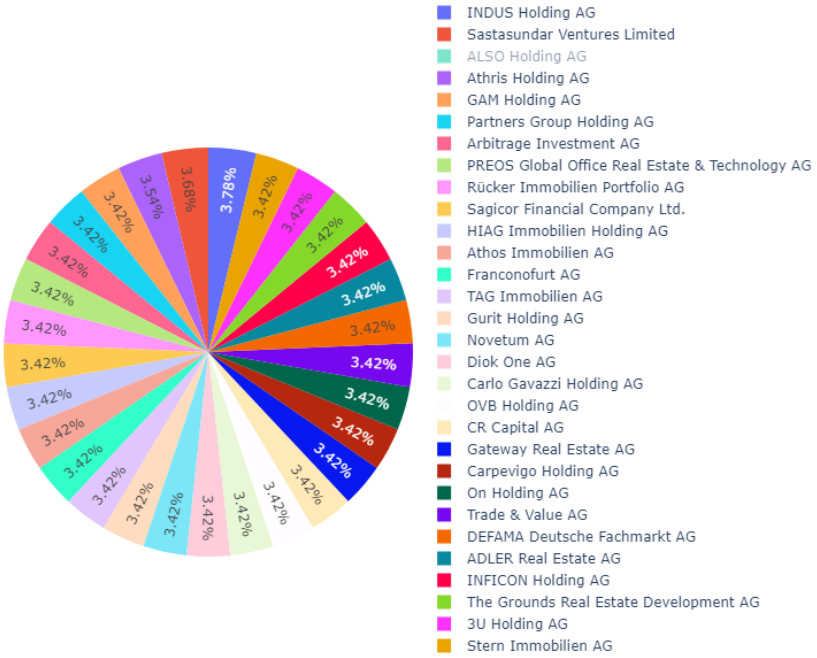

FIGURA 4.7: Tópico «ag.germany.switzerland» y sus 30 compañías más representativas.

- Una **mezcla de ambos**, véase: «taiwan.video.modules» (FIGURA 4.7), «stores.japan.foods», «banking.qatar.arab», «electric.china.power», o «madrid.inversiones.sicav».

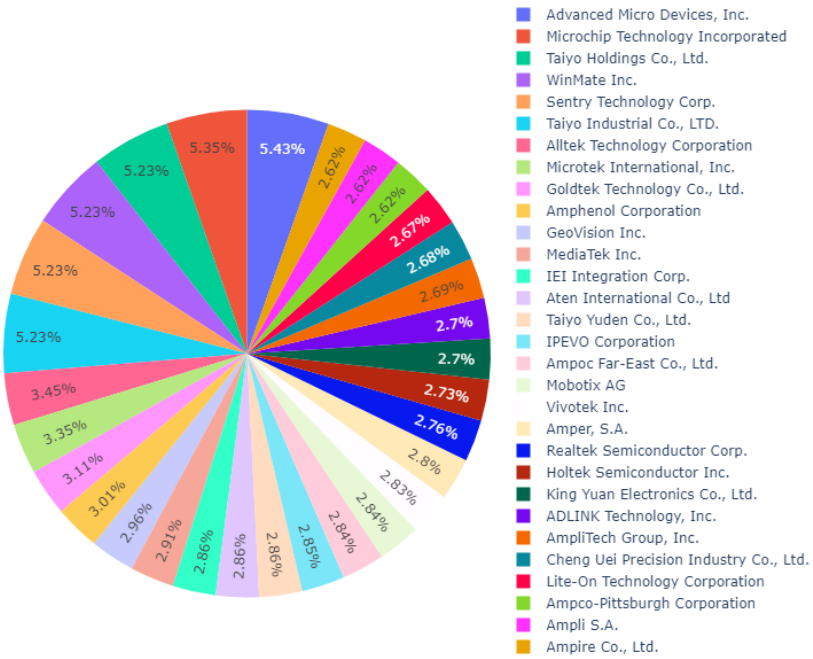

FIGURA 4.8: Tópico «taiwan.video.modules» y sus 30 compañías más representativas.

Haciendo hincapié en el ejemplo de la FIGURA 4.6, se puede analizar más de cerca, a una de las empresas que lo componen, *Microsoft Corporation* (FIGURA 4.9), y como ésta pertenece en su mayoría al tópico «data.software.cloud».

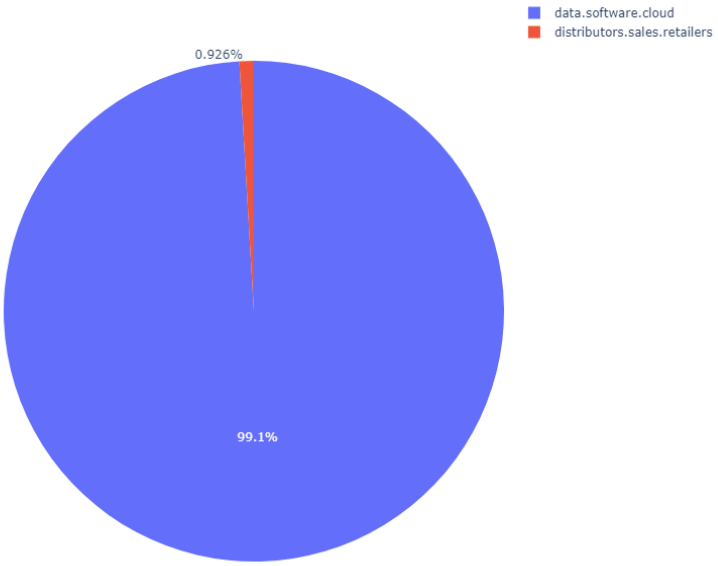

FIGURA 4.9: Tópicos pertenecientes a *Microsoft Corporation*.

Sin embargo, analizando una empresa que pudiera ser competencia directa como *Amazon.com, Inc.*(FIGURA 4.10), al menos en cuanto a servicios *cloud* se refiere, no devuelve nada en lo referente al tópico «data.software.cloud».

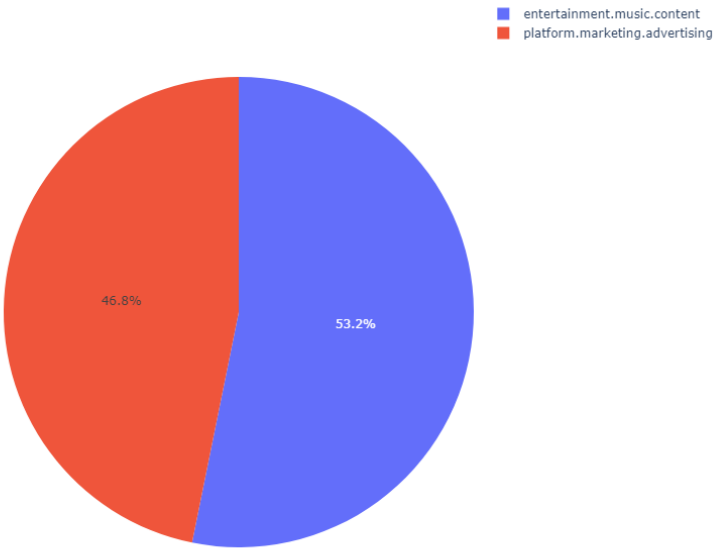

FIGURA 4.10: Tópicos pertenecientes a *Amazon.com, Inc.*.

Tal situación quizás se deba a que dentro de la descripción de la empresa *Amazon.com, Inc.*, lo único que aparece referente a servicios *cloud* es la palabra *"Amazon Web Services (AWS)"* y alguna escueta descripción como: *"the company provides compute, storage, database, analytics, machine learning, and other services"* y quizás el algoritmo no lo sepa identificar con servicios *cloud*. Del mismo modo que no se han usado los *earnigns calls* obtenidos del último año para la clasificación por tópicos, únicamente las descripciones.

Otro ejemplo de éxito, proveniente en su mayoría en parte del tópico «taiwan.video.modules» de la FIGURA 4.7 y del tópico «california.san.inc», sería la empresa *Advanced Micro Devices, Inc. (AMD)* (FIGURA 4.11). Estos resultados concuerdan con la referencia a una compañía estadounidense de semiconductores con sede en Santa Clara, California, que desarrolla procesadores de computación y productos tecnológicos similares.

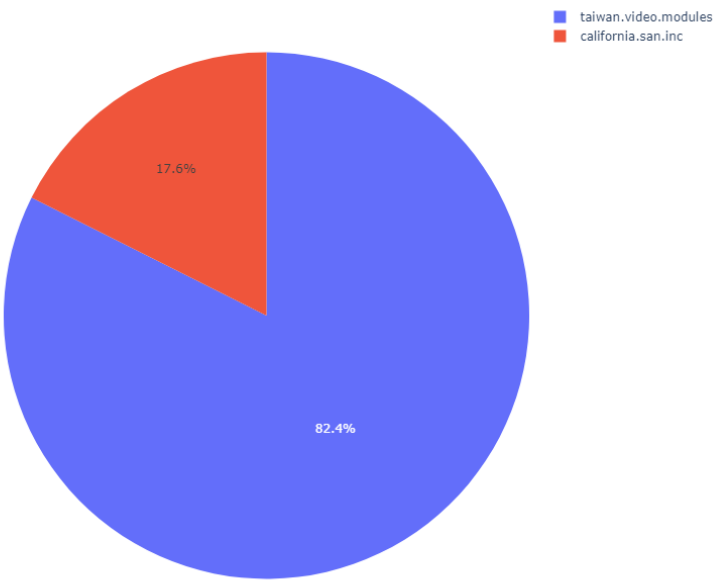

FIGURA 4.11: Tópicos pertenecientes a *Advanced Micro Devices, Inc. (AMD)*.

Como último ejemplo, y por concluir la referencia a las empresas mencionadas en la parte introductoria de este trabajo, se analiza *Adobe Inc.* (FIGURA 4.12). Empresa líder mundial en soluciones de Medios digitales, Experiencia digital, Marketing y Publicidad, tanto para el público profesional como para el *retail*, gracias a la gran cantidad de herramientas que ofrece. Así pues, y en

concordancia con la descripción, se puede observar la pertenencia al tópico «platform.marketing.advertising», así como a «distributors.sales.retailers».

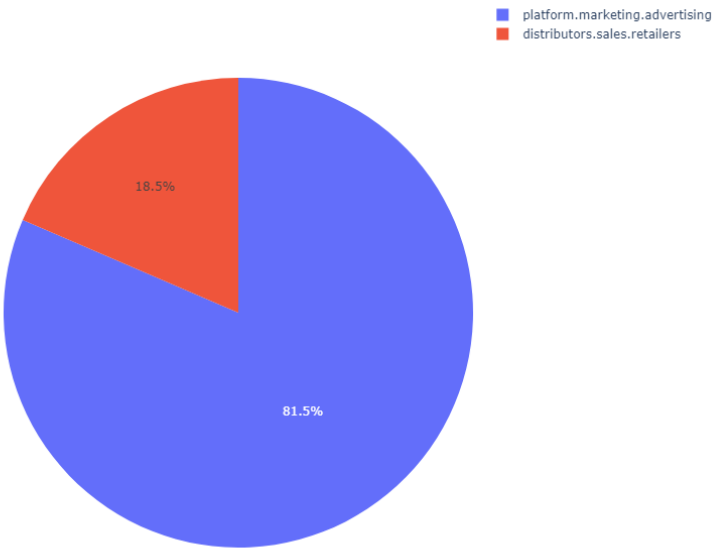

FIGURA 4.12: Tópicos pertenecientes a *Adobe Inc.*.

Estos resultados, entre otros, y a pesar de la gran concordancia con la realidad, dan a entender por la carencia de algunos tópicos en la composición de las empresas que quizás necesitarían complementarse mediante algún otro documento informativo, de cara a completar la información y mejorar la clasificación y agrupación de las empresas.

Capítulo 5

Conclusiones

En este apartado se pretende dar un enfoque global al trabajo realizado, destacando ventajas y también inconvenientes puesto que eso es lo que permitirá mejorar.

Como aspectos positivos resaltan el empleo de herramientas de procesamiento de lenguaje natural que han permitido comprender el contexto semántico de las palabras y frases en un texto. A su vez, ha quedado demostrado que el modelo es capaz de encontrar información relacionada de manera efectiva y eficiente en una gran cantidad de documentos en tiempo real y siendo altamente escalable. Todo ello se ha podido desplegar en la nube, permitiendo que cualquier persona del mundo lo utilice, por tan solo un coste mensual de 95€ (75€ la *API* y 20€ el resto).

Como aspectos negativos, una de las limitaciones del trabajo es que el modelo depende en gran medida de la calidad de los datos utilizados para entrenarlo. Si los datos de entrenamiento son incompletos o inexactos, el modelo puede producir resultados incorrectos. Una manera de solucionar esto sería introduciendo nuevos documentos como podrían ser los ya conocidos 10-K de la *SEC* o noticias. También una manera de mejorar los resultados del modelado de tópicos sería alimentarlo con datos de *earnings calls*.

En conclusión, después de un análisis detallado, se puede observar que el modelo en cuestión presenta ciertas limitaciones y no puede ser considerado como un sistema perfecto. Sin embargo, los resultados obtenidos a través de su implementación han demostrado ser satisfactorios y cumplen con los objetivos establecidos previamente. Es importante destacar que, a pesar de las posibles mejoras que podrían ser implementadas en el futuro, el modelo ha demostrado ser una herramienta efectiva en el ámbito en el que se ha aplicado.

Bibliografía

[1] R. Churchill y L. Singh, «The evolution of topic modeling», *ACM Computing Surveys*, vol. 54, n.º 10s, págs. 1-35, 2022.

[2] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer y R. Harshman, «Indexing by latent semantic analysis», *Journal of the American society for information science*, vol. 41, n.º 6, págs. 391-407, 1990.

[3] D. M. Blei, A. Y. Ng y M. I. Jordan, «Latent dirichlet allocation», *Journal of machine Learning research*, vol. 3, n.º Jan, págs. 993-1022, 2003.

[4] T. Mikolov, K. Chen, G. Corrado y J. Dean, «Efficient estimation of word representations in vector space», *arXiv preprint arXiv:1301.3781*, 2013.

[5] A. Vaswani, N. Shazeer, N. Parmar et al., «Attention is all you need», *Advances in neural information processing systems*, vol. 30, 2017.

[6] J. Devlin, M.-W. Chang, K. Lee y K. Toutanova, «Bert: Pre-training of deep bidirectional transformers for language understanding», *arXiv preprint arXiv:1810.04805*, 2018.

[7] N. Reimers e I. Gurevych, «Sentence-bert: Sentence embeddings using siamese bert-networks», *arXiv preprint arXiv:1908.10084*, 2019.

[8] M. Grootendorst, «BERTopic: Neural topic modeling with a class-based TF-IDF procedure», *arXiv preprint arXiv:2203.05794*, 2022.

[9] X. Zhu, S. Y. Yang y S. Moazeni, «Firm risk identification through topic analysis of textual financial disclosures», en *2016 IEEE Symposium Series on Computational Intelligence (SSCI)*, IEEE, 2016, págs. 1-8.

[10] S. Feuerriegel, A. Ratku y D. Neumann, «Analysis of how underlying topics in financial news affect stock prices using latent dirichlet allocation», en *2016 49th Hawaii International Conference on System Sciences (HICSS)*, IEEE, 2016, págs. 1072-1081.

[11] G. Li, X. Zhu, J. Wang, D. Wu y J. Li, «Using lda model to quantify and visualize textual financial stability report», *Procedia computer science*, vol. 122, págs. 370-376, 2017.

[12] D. Araci, «Finbert: Financial sentiment analysis with pre-trained language models», *arXiv preprint arXiv:1908.10063*, 2019.