Modelos de Deep learning para la clasificación de textos en los objetivos de desarrollo sostenible

Diego Felipe Carvajal Lombo

Universidad de los Andes

Departamento de Ingeniería de Sistemas y Computación

Ingeniería de Sistemas y Computación

Bogotá D.C., Colombia

2023

Modelos de Deep Learning para la clasificación de textos en los Objetivos de Desarrollo Sostenible

Diego Felipe Carvajal Lombo

Tesis presentada como requisito parcial para optar al título de:

Ingeniero de Sistemas y Computación

Directora:

Ph.D. Haydemar Nuñez Castro

Grupo de Investigación:

COMIT

Universidad de los Andes

Departamento de Ingeniería de Sistemas y Computación

Ingeniería de Sistemas y Computación

Ciudad, Colombia

2023

Para mi padre, quien fue la persona que sembró todos los valores y enseñanzas para convertirme en la persona que soy hoy en día.

Resumen

Este estudio se centra en el tratamiento de comentarios y reseñas de los Objetivos de Desarrollo Sostenible (ODS) mediante un modelo de clasificación basado en técnicas de Deep Learning, implementando un modelo pre entrenado para el entendimiento de los datos, una red neuronal para clasificarlos y técnicas de aumentación de datos. Los resultados destacan una notable mejora en la precisión, especialmente en categorías con datos limitados, además de una precisión del 91.15% a la hora de clasificar. La implementación culmina en una aplicación web que permite a los usuarios ingresar textos o cargar archivos para obtener su clasificación respecto a los ODS, contribuyendo así a una interpretación más efectiva y análisis de información clave para el logro de los ODS.

Palabras clave: Machine Learning, Deep Learning, Aumentación de textos.

Abstract

This study focuses on the processing of comments and reviews of the Sustainable Development Goals (SDGs) using a classification model based on Deep Learning techniques, implementing a pre-trained model for data understanding, a neural network for classification and data augmentation techniques. The results highlight a remarkable improvement in accuracy, especially in categories with limited data, in addition to an accuracy of 91.15% when classifying. The implementation culminates in a web application that allows users to enter text or upload files to obtain their classification against the SDGs, thus contributing to more effective interpretation and analysis of key information for the achievement of the SDGs.

Keywords: Machine Learning, Deep Learning, Text Augmentation.

Contenido

Pág.

Introducción

La organización de las Naciones Unidas (ONU) nace a finales del año 1945, recién acabada la segunda guerra mundial. Luego de este fatídico hito en la historia de la humanidad, se propone crear una organización comprometida con el mantenimiento de la paz y la seguridad internacional. El número de países integrantes de la ONU actualmente es de 193. Estos países trabajan en conjunto en asuntos como el derecho internacional, la paz y la seguridad, el desarrollo económico y social, los asuntos humanitarios y los derechos humanos [1].

Los Objetivos de Desarrollo Sostenible (ODS) son un conjunto de objetivos globales para que, en términos generales, los diferentes países apunten a erradicar la pobreza, proteger el planeta y asegurar la prosperidad para todos como parte de una nueva agenda[2]. Estos objetivos tienen como ideal garantizar que para el 2030 todas las personas disfruten de paz y prosperidad. Los 17 ODS están integrados, es decir, reconocen que la acción en un área afectará los resultados en otras áreas y que el desarrollo debe equilibrar la sostenibilidad social, económica y ambiental [3].

En Colombia, se ha puesto en marcha la agenda y para esto dispone de espacios de diálogo permanente de todos los sectores del gobierno nacional, los gobiernos departamentales y municipales. Además, generan iniciativas en las que haya mayor apropiación por parte de los demás actores de la sociedad [4].

Debido a su importancia a nivel global, es muy importante para todas las naciones poder medir el crecimiento o decrecimiento de los mismos objetivos, esto con el fin de poder tomar las medidas y decisiones necesarias para que en el año 2030 se pueda obtener una mejora tanto teórica como práctica en las diferentes áreas que cubren los ODS y mejorar la calidad de vida de la población global.

Ante este desafío, la inteligencia artificial (IA), y en especial, el Machine Learning (ML) y el Deep Learning (DL) aparece como una colaboración desmesurada para los distintos países para lograr cumplir los objetivos. La Inteligencia Artificial se puede definir como la combinación de algoritmos planteados con el propósito de crear máquinas que presenten

las mismas capacidades que el ser humano [5]. Dentro de esta gran ciencia, existe la rama de Machine Learning, ese a través de algoritmos, dota a ordenadores de la capacidad de identificar patrones en datos masivos y elaborar predicciones y elaborar predicciones [6]. Y para culminar, dentro de esta rama se encuentra el Deep Learning, que es esencialmente una red neuronal que intenta simular el comportamiento del cerebro humano, ingiriendo y procesando datos no estructurados, como texto e imágenes, y con esto automatiza la extracción de características eliminando la dependencia humana [7].

Ya existen varias formas en las cuales la tecnología, y en especial, la Inteligencia Artificial están ayudando alrededor del mundo para el cumplimiento de los ODS. Por ejemplo, la IA posibilita el desarrollo de una agricultura sostenible mediante sistemas de soporte para toma de decisiones al sugerirles a los pequeños agricultores la mejor variedad para cultivar [8].

Otro ejemplo del uso de la Inteligencia artificial para el beneficio de los ODS lo menciona Alejandro Arango, líder de transformación digital y analítica avanzada de Ecopetrol, que plantea que la IA puede aportar con el cumplimiento de los ODS relacionados con la salud, gracias a la interconectividad entre gran parte de los hospitales se puede monitorea y mapear en tiempo real las condiciones globales de salud [8].

En este proyecto nos enfocaremos en mejorar el procesamiento de textos/comentarios de las personas para poder clasificar su contenido entre los 16 diferentes ODS, para esto nos basaremos en un proyecto de grado de maestría realizado previamente en el cual se reflejan dificultades para la clasificación de categorías de ODS con pocos datos/muestras existentes.

1. Descripción

En este capítulo se va a desarrollar el contexto y el planteamiento del problema a resolver en el proyecto de grado.

1.1 Contexto

La necesidad de comprender y abordar los Objetivos de Desarrollo Sostenible (ODS) establecidos por las Naciones Unidas en su Agenda 2030 ha llevado a la implementación de diversas estrategias para analizar la percepción y la participación de la comunidad en torno a estos objetivos. El proyecto se enfoca en la clasificación de textos, específicamente comentarios y opiniones recopilados a través de encuestas realizadas en territorios vinculados a los Objetivos de Desarrollo Sostenible, cuyas clasificaciones serán los mismos ODS.

El principal contexto de este trabajo es un proyecto de grado de maestría anterior, el cual se enfocó en Desarrollar un modelo que permita la clasificación de textos obtenidos mediante encuestas realizadas en territorios en los ODS planteados en la Agenda 2030 de las naciones unidas, haciendo uso de diferentes técnicas de aprendizaje profundo [9].

1.2 Descripción del problema

En el trabajo de maestría realizado previamente, se encontraron diversas dificultades en el modelo de clasificación planteado. El principal problema que se observa son los bajos porcentajes de aciertos del modelo en ciertos objetivos específicos, el común denominador que se puede identificar es que estos objetivos eran aquellos con la menor cantidad de datos/textos recolectados. Esto nos quiere decir que probablemente el modelo tenga una dificultad para identificar y clasificar textos cuya temática principal sean los objetivos con pocos datos suministrados.

El segundo problema que se puede identificar es que, dado la similitud de los mismos objetivos de desarrollo sostenible, se puede encontrar que hay textos que pueden tocar

temáticas relacionadas o parecidas, por lo que es posible que el modelo planteado se “confunda” al momento de clasificar los textos y seleccione un objetivo que puede tener cierto contenido de este, pero no es seleccionado como el objetivo principal. Es decir, existen textos que pueden tocar diversas temáticas o tener una multietiqueta.

2. Marco teórico

2.1 Definiciones clave

2.1.1 Procesamiento de lenguaje natural

Este se define como el campo de la Inteligencia Artificial que se consiste en transformar el lenguaje natural, en un lenguaje formal, como el de la programación para que los ordenadores puedan procesar [10]. En otras palabras, trata de dotar a las computadoras con la capacidad de comprender el lenguaje natural del ser humano.

2.1.2 Modelos pre entrenados para procesamiento de lenguaje (Transformers)

Existen distintos modelos pre entrenados de libre uso los cuales nos proporcionan automáticamente un procesamiento de lenguaje natural para poder ser usado más fácilmente posteriormente para distintas tareas como pueden ser la clasificación, predicción y demás. Dentro de estos hay varias modelos y algoritmos, aquí se va a definir brevemente el que fue usado en el proyecto:

- **BERT:** Viene de sus siglas en inglés *Bidirectional Encoder Representations from Transformers,* es un algoritmo proporcionado por Google que utiliza una arquitectura de transformadores para aprender representaciones bidireccionales de texto, lo que permite comprender mejor el contexto de las palabras dentro de una oración o párrafo [11].
- **Embeddings:** Los embeddings son una técnica de procesamiento de lenguaje natural que convierte el lenguaje humano en vectores matemáticos. Estos vectores son una representación del significado subyacente de las palabras, lo que permie que las computadoras procesen el lenguaje de manera más efectiva [12]. Un embedding se refiere a esta representación numérica de las palabras en un espacio vectorial, lo que facilita que las máquinas procesen y comprendan el lenguake
- **Búsqueda de hiperparámetros:** Se refiere a la tarea de encontrar la combinación óptima de valores para los parámetros que son imposible de que un modelo de Machine Learning aprenda durante el entrenamiento. Los hiperparámetros son valores ajustables que permiten controlar el proceso de entrenamiento de un

modelo. Por ejemplo, con redes neuronales, puede decidir el número de nodos de cada capa [13].

2.1.3 Redes neuronales

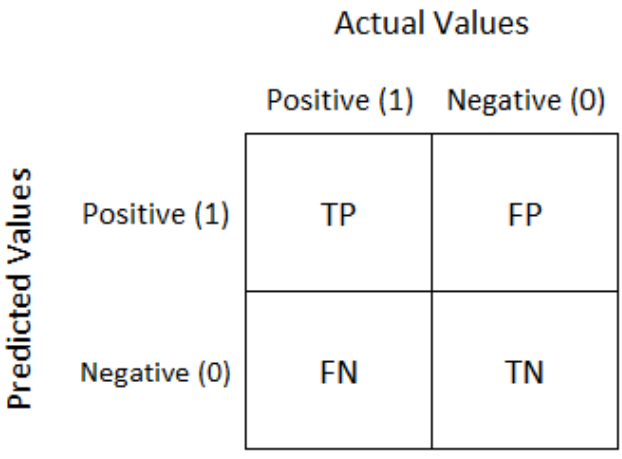
Las redes neuronales son un método de la inteligencia artificial que enseña a los computadores a procesar datos de una manera que está inspirada en el cerebro humano [14]. Están formadas por capas de nodos, que contienen una capa de entrada, una o varias capas ocultas o intermedias, y una capa de salida. Cada nodo se conecta a otro y tiene un peso y un umbral asociados. Si la salida de un nodo individual está por encima del valor del umbral especificado, dicho nodo se activa y envía datos a la siguiente capa de la red [15].

2.1.4 Métricas de desempeño

A continuación, se muestran las métricas empleadas para evaluar el desempeño del modelo:

- Matriz de confusión: Esta es una tabla que muestra el desempeño de un algoritmo. En las filas se representa el número de predicciones de cada clase, mientras que en las columnas se muestran las instancias de la clase real [16].

Figura 1: Matriz de confusión



- Recall: Es la relación que se encuentra entre los datos en los cuales la predicción/agrupación fue positiva (Totales Positivos) sobre la cantidad de datos

positivos en el conjunto de datos. Es decir, de todo lo que es positivo (o negativo), ¿cuánto está bien clasificado?

$$Recall = \frac{TP}{TP + FN}$$

- Precisión: Es la relación que se encuentra entre los datos en los cuales la predicción/agrupación fue positiva (Totales Positivos) sobre la cantidad de datos positivos arrojados por el modelo. Es decir, de los que el modelo dice que es positivo (o negativo) ¿Cuánto es verdad?

$$Precisión = \frac{TP}{TP + FP}$$

- F1-Score: Esta métrica combina el Recall y la Precisión. Este valor es la media ponderada de las dos métricas anteriores. Por lo que, entre mayor sea el F1, mayor será la precisión y Recall, lo que significa que un modelo tiene un mejor rendimiento.

$$F1 - Score = \frac{2 * Precisión * Recall}{Precisión + Recall}$$

2.2 Estado del arte

Como se mencionó en la introducción, este trabajo se basa en un proyecto de grado de maestría el cual consiguió realizar un algoritmo de clasificación de textos en los Objetivos de Desarrollo Sostenible, usando una variación del modelo pre entrenado de Google *BERT* llamado "*DistilRoBERTa*". Y luego de usar este modelo pre entrenado se construyó una red neuronal cuyas entradas eran los embeddings generados por el modelo pre entrenado y su salida son 17 neuronas con la probabilidad (valor entre 0 y 1) de que el texto pertenezca a él ODS correspondiente.

En este estudio, se realizó una comparación de diferentes modelos pre entrenados para la transformación de los datos de textos a su representación numérica (embedding) para poder seleccionar el óptimo para los datos que fueron recolectados por la misma autora. En esta comparación se utilizaron distintos modelos como Distilroberta, Robertas y MPNETC. Como se puede evidenciar en la tabla 1, los mejores resultados se obtienen utilizando Distilroberta ya que se mantiene más información del proceso y codificación de los textos. [9].

Tabla 1: Exactitud modelos base sobre conjunto de datos de validación

Modelo	Accurracy
Distilroberta	0.98
Robertas	0.98
MPNET	0.95

Nota: Tabla extraída y adaptada de [9]

En este informe, se presenta también la construcción del modelo de red neuronal que toma los embeddings generados por el modelo pre entrenado, y, haciendo uso de una red neuronal, realiza una clasificación de los textos en los ODS, por lo que al final se obtiene un algoritmo con dos modelos. El primer modelo consta de un transformador pre entrenado de libre uso por Google que permite convertir los textos a una representación matemática, la cual es comprensible para una red neuronal (segundo modelo) que permite la clasificación de los textos.

Como se puede ver en la tabla 2, en el resultado final del algoritmo construido se puede evidenciar unos puntajes de precisión y recall altos (>0.9) en la mayoría de los Objetivos de Desarrollo Sostenible. Sin embargo, a la vez, se puede ver unos puntajes bajos, llegando hasta valores de 0.59 en la precisión de algunos ODS específicos. Estos se pueden evidenciar especialmente en los ODS 9, 10 y 12. Haciendo un análisis de características comunes de estas categorías, es que son los ODS cuya cantidad de datos es menor en comparación con las demás. Estos 3 ODS tienen un total de datos no mayor a 500 cada uno. No obstante, las demás categorías llegan a tener al menos más de 1000 datos cada uno.

Tabla 2: Resultados de clasificación por categoría.

	Distilroberta	
Etiqueta	precisión	recall
ODS 1	0.88	0.80
ODS 2	0.87	0.85
ODS 3	0.92	0.95
ODS 4	0.95	0.95
ODS 5	0.93	0.96
ODS 6	0.90	0.94
ODS 7	0.90	0.92
ODS 8	0.79	0.65
ODS 9	0.71	0.84
ODS 10	0.59	0.70
ODS 11	0.89	0.88
ODS 12	0.78	0.71
ODS 13	0.88	0.81
ODS 14	0.96	0.96
ODS 15	0.88	0.87
ODS 16	0.95	0.96

Nota: Tabla extraída y adaptada de [9]

Finalmente, en esta investigación se construye una herramienta web que permite utilizar el algoritmo creado para la clasificación de nuevos textos. Sin embargo, esta herramienta se ve limitada debido a que muestra únicamente el ODS con mayor relación obtenida según el algoritmo, pero es posible que un solo texto esté relacionado a 2 o más categorías, por lo que mostrar únicamente una clasificación es corto.

2.3 Propuesta de solución

Dado el análisis de las problemáticas y limitaciones del estado del arte actual, se propone rehacer el algoritmo de clasificación y la página web con algunas modificaciones para poder mejorar las estadísticas obtenidas y una página web con más información, detallada y con una nueva funcionalidad.

Para lograr esto, se propone antes del entrenamiento de los modelos, realizar un proceso de aumentación de textos en las categorías cuya cantidad total de datos sea inferior a 500 cada uno. Esto con el fin de poder ver cómo se comportan los algoritmos y transformadores con datos artificiales creados a partir de los datos originales.

Además de esto, se propone utilizar el mismo modelo de la generación de embeddings ya que este fue el que mejores resultados se obtuvieron en el estudio previo, pero realizar una nueva búsqueda de hiper parámetros y selección de capas para la red neuronal que clasifica los textos.

Finalmente, se decide hacer una página web nueva que permita mostrar las dos principales categorías de los ODS a los que el algoritmo dice que hace referencia el texto ingresado, con una gráfica de barras que muestre el total la probabilidad de temática de cada uno de los ODS, con una nueva funcionalidad que permita clasificar varios textos contenidos en el mismo archivo (PDF o Excel).

2.4 Objetivos

El objetivo del proyecto es desarrollar una solución, basada en técnicas de procesamiento de lenguaje natural y deep learning, que facilite la interpretación y análisis de información textual para la identificación de relaciones semánticas con los Objetivos de Desarrollo Sostenibles.

2.4.1 Objetivos específicos

- Aplicar técnicas de aumentación de datos para enriquecer el conjunto de entrenamiento
- Refinar los modelos de clasificación ya existentes mediante hiper parámetros y técnicas de Deep Learning.
- Proponer un nuevo modelo de clasificación que pueda mejorar o complementar los existentes.
- Ampliar las capacidades de la aplicación web para incluir varias funcionalidades, como la capacidad de ingresar texto y obtener su clasificación correspondiente, así como la posibilidad de cargar archivos con textos para determinar a qué Objetivo de Desarrollo Sostenible (ODS) pertenecen.

3. Desarrollo de la solución

En este capítulo se va a describir la implementación de la herramienta, comenzando por el tratamiento de los datos suministrados, luego explicando y detallando el proceso y selección de aumentación de textos, el entrenamiento del modelo y la selección de hiper parámetros.

3.1 Validación de los datos

En primer lugar, lo que se realizó fue una exploración exhaustiva de los datos, su calidad, contenido, variables contenidas, y, en general, un preprocesamiento de los datos. Al comenzar se puede observar que los datos suministrados contienen distintas columnas que son las siguientes: text, Textos_espanol, sdg, labels_negative, labels_positive, agreement, large. De estas variables se decide seleccionar únicamente las 3 primeras, es decir, los textos en los dos idiomas (español e inglés) y su respectiva categoría, dado que las demás columnas no tienen información representativa para los objetivos del proyecto.

En segundo lugar, dado que los datos fueron extraídos y traducidos previamente, existe la posibilidad de que estos no se encuentren totalmente limpios para un procesamiento y entrenamiento de los modelos, por lo que se realizó una validación de los respectivos idiomas de los textos, así como una revisión de caracteres especiales. Para esto, se hizo uso de dos librerías distintas. La primera, llamada *langdetect* proporcionada por pypi permitió filtrar los textos en los idiomas en los que se encontraban escritos y con esto poder verificar y en dado caso eliminarlos, únicamente se dejaron los textos que se encontraran en inglés y español en sus respectivas columnas. La segunda librería, llamada *ftfy* permitió corregir todos los textos en español que no se encontraban con un encoding apropiado, es decir, existían textos que probablemente debido a la exportación de los datos, las vocales con tildes no se leían correctamente. Por ejemplo, en vez de aparecer un ‘ó ‘, aparecía una ‘Ã³’. Todos estos problemas de semántica fueron solucionados.

3.2 Aumentación de textos

Como se menciona previamente, una de las mayores dificultades del algoritmo ya existente fue la clasificación en las categorías con pocos datos, por lo que el primer desafío fue encontrar una forma de generar datos artificiales de algunas categorías de los ODS que fueran los suficientemente parecidos en contenido a los textos originales, sin ser una copia idéntica o casi idéntica de los mismos. Para esto, se decide hacer el proceso de aumentación de textos en todas las categorías cuyas cantidades totales de datos fueran inferiores a 500. Es decir, los ODS 9, 10, 12 y 15.

La primera opción que se intentó ejecutar fue el uso de librerías Open Source como *nlpaug* [17]. Esta librería, tiene diversas variaciones dependiendo del tipo de algoritmo que se quiera utilizar y el idioma. Esta librería tiene 3 tipos de aumentaciones, cada una con diferentes algoritmos; la aumentación de caracteres, de palabras y de oraciones. En este proyecto se realiza las últimas dos.

En la aumentación de palabras hay diferentes algoritmos. Sin embargo, debido a la naturaleza de la librería, la mayoría de estos funcionan únicamente con textos en inglés, por lo que dependiendo del algoritmo escogido se utilizan los textos en español o en inglés. El primer algoritmo que se utilizó fue la aumentación de palabras con sinónimos, este obtiene palabras del texto original y las sustituye por sinónimos de WordNet, que es una base de datos léxica en inglés y otros idiomas. Para este algoritmo, gracias a WordNet, era posible seleccionar el idioma de preferencia.

Al usar este algoritmo, en un principio con textos cortos y de ejemplos, se pudo evidenciar que los textos generados tenían una variación significativa al original. Por ejemplo, un texto de ejemplo “The quick brown fox jumps over the lazy dog.” Era sustituido por el siguiente texto aumentado: “The speedy brown fox jumps over the inattentive dog.”. Sin embargo, al empezar a utilizar los textos proporcionados sobre los ODS, se podía notar que, al ser textos más largos, la diferencia entre los mismos no solía ser muy grande. Por lo general,

únicamente eran reemplazados y sustituidos por un sinónimo el verbo principal de la oración, por ejemplo:

Tabla 3: Comparación de textos aumentados con NLPAUG

Original	Synonym Augmenter
Es muy poco probable que la meta 9.c de los ODS se alcance en el plazo de 2020. En la práctica, es prácticamente imposible disfrutar de Internet de forma eficaz a través de una conexión 2G. Sólo el 76% de la población mundial tiene acceso a una señal 3G, y sólo el 43% tiene acceso a una conexión 4G. Así pues, la mayor parte del mundo conectado sigue sin estarlo, la mayoría en países en desarrollo.	Es muy poco probable que la meta 9. c de los ODS se alcance en el plazo de 2020. En la práctica, es prácticamente imposible disfrutar de Internet de forma eficiente a través de una conexión 2G. Sólo el 76% de la localidad global tiene acceso a una señal 3G, y sólo el 43% tiene acceso a una unión 4G. Así pues, la mayor parte del mundo conectado sigue sin estarlo, la mayoría en países en desarrollo.

Debido a la alta similitud de los textos generados con los textos aumentados, no es considerable utilizar esta técnica para generar los nuevos datos, ya que al tener muchos fragmentos parecidos los modelos pre entrenados van a generar embeddings similares, generando datos iguales para el entrenamiento de la red neuronal, lo que surge en sesgos en las representaciones y sobreajuste en el modelo. En otras palabras, es más perjudicial que favorable realizar una aumentación de datos tan similar.

El conjunto de los algoritmos se puede agrupar en 3 acciones: insertar, cambiar o borrar palabras, esto mediante diferentes formas, puede ser mediante una inserción por similitud TF-IDF (fórmula que permite calcular el peso de una palabra en un documento [18]), cambiar palabras por sus antónimos, eliminar palabras al azar o insertar palabras por su similitud contextual en embeddings utilizando BERT o DistilBERT, etc. A pesar de haber implementado todos los algoritmos mencionados anteriormente, se obtuvieron resultados similares al previamente presentado, por lo que no se consideraron lo suficientemente óptimos para los cumplir con los objetivos propuestos, por lo que se propone hacer uso del API de OpenAI.

La API ChatGPT es una interfaz de programación de aplicaciones que permite a los usuarios experimentar un chatbot basado en Inteligencia Artificial [19]. En base a estas features del API, se decide hacer una funcionalidad que permita obtener los textos originales de las categorías seleccionadas para su aumentación, y mediante un prompt (conjunto de palabras que desencadenan la generación de contenidos a través de un software de Inteligencia Artificial [20]) poder generar estos nuevos textos artificiales, con un contenido parecido sin ser exactamente el mismo.

Como se menciona antes, el prompt se podría definir como una “orden” que interpreta la IA y genera contenido (frases, palabras, párrafos) en base a esta prescripción. Para la generación de estos textos artificiales se le pasa a la API el siguiente prompt en español con el motor de *davinci-002* [21]:

“Dado el siguiente texto en español: 'original_text', genera un nuevo texto similares en tema y contenido. No debes generar el mismo texto”

Donde ‘original_text’ es un dato/texto original proporcionado para este proyecto. Esto se hace de manera iterativa, en la cual, se van seleccionando de manera ordenada los textos originales hasta que la suma de la cantidad de datos total (originales y aumentados) sea mayor a 500. Con esto garantizamos que el origen de cada uno de los datos aumentados sea un texto diferente. Gracias a esta herramienta, se pudo obtener datos nuevos cuyo contenido y temática sea similar a los originales, sin ser una copia de este. Un ejemplo es el siguiente:

Tabla 4: Comparación de textos aumentados con OpenAI

Original	OpenAI API	ODS
Las disparidades en la productividad industrial entre las naciones ricas y pobres siguen siendo marcadas. Por ejemplo, el VAM per cápita fue de solo 114 dólares en los PMA, frente a los 4.938 dólares de Europa y América del Norte, en 2018. Con una cantidad relativamente pequeña de inversión de capital y una base de recursos predominantemente local, las industrias a pequeña escala generan una cantidad sustancial de empleo y autoempleo. Sin embargo, uno de los mayores desafíos a los que se enfrentan esas industrias es el acceso a préstamos o líneas de crédito para las actividades empresariales cotidianas. Una financiación adecuada es crucial para que esas industrias crezcan, ya que les permite innovar, mejorar la eficiencia, expandirse a nuevos mercados y crear nuevas oportunidades de empleo. Aunque el 31,5% de las pequeñas industrias (manufactureras y de servicios) de todo el mundo se benefician de préstamos o líneas de crédito, destacan las diferencias regionales.	Las desigualdades económicas entre las diferentes naciones también está presente en el ámbito industrial. El valor agregado de la producción por habitante fue de 114 dólares en los Pases Menos Desarrollados en 2018, mientras que en Europa y 26Norteamérica fue de 4.938 dólares. Dado que gran parte de estas industrias son de pequeña escala, producen mucha cantidad de empleo y auto empleo. No obstante, uno de los grandes retos al que se enfrentan es el acceso a préstamos y financiamiento. Estas soluciones les permiten innovar, mejorar la eficiencia, extenderse a otros mercados y generar empleo.	9

Como se puede evidenciar en la tabla 4, los textos generados por el API de OpenAI tratan de decir lo mismo que su predecesor. No obstante, podemos ver que las diferencias entre ambos textos son muy significativas, donde por lo general solamente mantiene los datos y nombres propios. Sin embargo, podemos ver una limitación de estos; dado que en el momento de usar el API es necesario indicar el número de tokens máximo del texto generado (parámetro *max_tokens*). A pesar de indicarle que este valor sea igual a la cantidad de palabras que tiene el texto original, muchas veces los textos se ven recortados como es el caso del ejemplo de la tabla 4, en el cual el párrafo final no logra parafrasearlo. Pese a esta limitante, esto no es un error que impida que los modelos puedan ser

entrenados o generen algún tipo de sobre especialización, por lo que se decide trabajar con esta API para realizar la aumentación de datos. Al finalizar con la aumentado los datos, se obtuvieron los siguientes datos:

Tabla 5: Total de datos

ODS	Original	Aumentados
1	787	787
2	567	567
3	1425	1425
4	1635	1635
5	1693	1693
6	1071	1071
7	1233	1233
8	683	683
9	439	550
10	369	550
11	940	940
12	247	490
13	726	726
14	579	579
15	406	500
16	1669	1669
Total	14469	15098

Nota: Debido a que los datos aumentados tienen una relación 1 a 1 con los originales, en la categoría 12 no era posible llegar a 500.

3.3 Generación de embeddings con el modelo pre entrenado

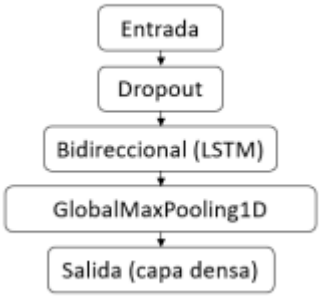
Ya teniendo un nuevo conjunto de datos con algunas categorías aumentadas, el siguiente paso es generar los embeddings de cada uno de los datos/textos. Como se menciona en el marco teórico, un embedding es básicamente una representación numérica de un texto para el fácil entendimiento de una máquina. Para esto, se utiliza el modelo seleccionado en el trabajo de grado en el que se basa este proyecto. En este trabajo se realizó la debida comparación de los modelos pre entrenados con sus respectivos resultados, medición de métricas y demás para poder elegir el mejor modelo para estos datos.

La decisión final fue utilizar el modelo pre entrenaod de Google DistilRoBERTa con la biblioteca de Hugging Face (Transformers), haciendo uso de los métodos AutoTokenizer y Automodel, y finalmente realizar de forma manual el pooling sobre la última capa oculta del modelo, la cual da como resultado un vector de 768 dimensiones [9]. Luego de hacer la debida transformación de todos los textos a su correspondiente embedding, se procedió a hacer la debida separación de los datos en entrenamiento, validación y test. Esto mediante la siguiente distribución: 64% entrenamiento, 16% validación y 20% Test.

3.4 Entrenamiento y validación

Una vez obtenidos los embeddings y realizada la separación del dataset para obtener los datos de entrenamiento, se procede a hacer el entrenamiento de la red neuronal que se encarga de realizar la clasificación de los textos. En el trabajo de grado anterior se utilizó una red neuronal de 5 capas: Entrada, Dropout, LSTM, GlobalMaxPooling1D y Salida (Capa densa), por lo que se utilizará esa arquitectura para empezar a probar y entrenar el modelo.

Figura 2: Red neuronal inicial



Nota: Imagen extraída de [9].

En primer lugar, hay que tener en cuenta que la entrada es una capa de 768 neuronas, esto debido a que los embeddings tienen esta misma dimensión, es decir, cada dimensión del embedding es una neurona en la capa de entrada. En segundo lugar, dado que estamos haciendo una clasificación en 16 categorías correspondientes a cada ODS, la capa de salida tiene 16 neuronas, las cuales cada una representa la probabilidad (valor entre 0 y 1) de que el embedding de entrada corresponda a ese ODS. Para el entrenamiento, el conjunto 'X' (características de entrada) de datos son los embeddings generados y con conjunto 'Y' (etiqueta de salida) son un vector en el cual todas las dimensiones son 0 exceptuado la que corresponde a la categoría del texto, cuyo valor es 1.

Teniendo esto claro, se empieza el entrenamiento utilizando una red neuronal parecida a la que se presentó en el proyecto de grado anterior, con algunas variaciones en las funciones callback, delta mínimo y paciencia. Sin embargo, se decide poner una segunda capa LSTM para que la disminución de dimensiones no sea tan grande entre capas. Para entender esto hay que tener claro que partimos de una capa de 768 neuronas y terminamos en una capa de 16 neuronas, por lo que las capas intermedias deben reducir la dimensionalidad de los datos.

Luego de agregar la segunda capa LSTM a la red neuronal luego de la primera, se realiza una búsqueda de hiperparámetros de las unidades que deben tener cada una de estas capas. Esta búsqueda se hace mediante el método de *Grid Search,* que permite buscar la mejor combinación de resultados entre todas las composiciones posibles. A esta búsqueda también se agregó el porcentaje de la capa DropOut, que es la capa que “apaga” aleatoriamente algunas de las neuronas (dependiendo del porcentaje dado) para que el modelo no se sobre especialice. La búsqueda arrojó que la primera capa LSTM debe tener 512 neuronas, la segunda debe tener 128 y la capa de DropOut debe tener un porcentaje de 0.3.

Luego de un segundo entrenamiento, con los valores obtenidos en la búsqueda de hiper parámetros, realizando la validación del accuracy y los del modelo en las épocas de entrenamiento, se pudo evidencias que, a pesar de tener una capa de DropOut cuyo objetivo es que no se sobre especialice el modelo, se estaba incurriendo en este error. Esto se podía evidencias debido a que los valores de estas métricas, con el pasar de las épocas del entrenamiento, el puntaje de los valores de entrenamiento y de validación se separaban. Por lo que se decide implementar una segunda capa de DropOut, además de agregar un regularizador kernel y un regularizador recurrente en las capas en las capas bidireccionales (LSTM). Además de agregar el optimizador Adam con una tasa de aprendizaje que también se encontró con la búsqueda de *Grid Search.* Luego de realizar estos ajustes, se logró obtener un modelo que no esté sobre especializado y mantuviera, en general, los valores de precisión obtenidos en el trabajo anterior.

Figura 3: Valores de la métrica los para los datos de entrenamiento y validación a través de las épocas.

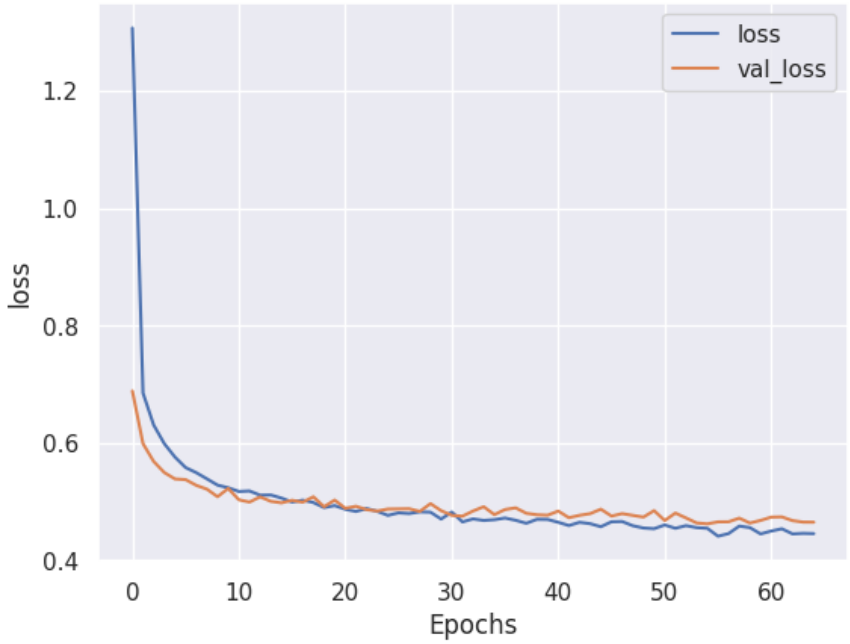
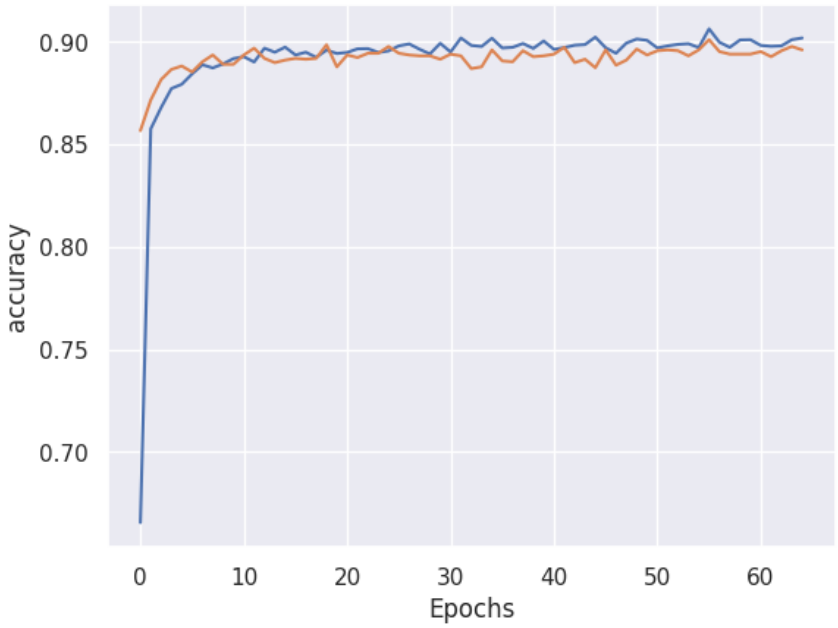
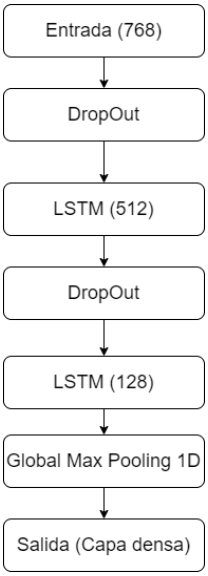


Figura 4: Valores de la métrica accuracy para los datos de entrenamiento y validación a través de las épocas.



Luego de haber obtenido este modelo con estas métricas de validación, se procedió a guardar la arquitectura y modelo final.

Figura 5: Arquitectura de la red neuronal final.



3.5 Construcción herramienta web

Finalmente, se decide construir una nueva página web que permita a los usuarios hacer uso del modelo final realizado, para que puedan clasificar los textos. Para esto se hizo uso de la herramienta *Streamlit.* Que posee unos componentes para la interfaz ya predefinidos y facilita la implementación de esta.

La primera funcionalidad permite ingresar un texto y el modelo va a realizar la clasificación del mimo. En la interfaz, al lado del texto se podrán ver las dos categorías con mayor probabilidad a la que pertenezcan este texto junto a su respectiva imagen. Además, en la parte derecha se ve un diagrama de barras con la probabilidad de pertenencia del texto en las dos categorías obtenidas. En la figura 6 se realiza la clasificación del siguiente texto:

Los intereses del clan a menudo tienen prioridad sobre los intereses de las víctimas individuales y las familias eligen reconciliarse a través del sistema consuetudinario en

lugar de buscar reparación para las víctimas. Esto lleva a que las mujeres víctimas de violación se vean obligadas a casarse con el violador, siguiendo el dictamen de los ancianos de la aldea aplicando prácticas consuetudinarias (A/HRC/20/16/Add.3). En Ghana, las autoridades tradicionales, como los jefes tribales en muchas áreas rurales, gobiernan sobre cuestiones y disputas relacionadas con la tierra y los derechos de propiedad, así como asuntos que involucran "interferencias sobrenaturales", incluidas las denuncias de brujería. En Afganistán, la sharia, el derecho consuetudinario, el sistema legal formal y secular y el derecho internacional existen en paralelo.

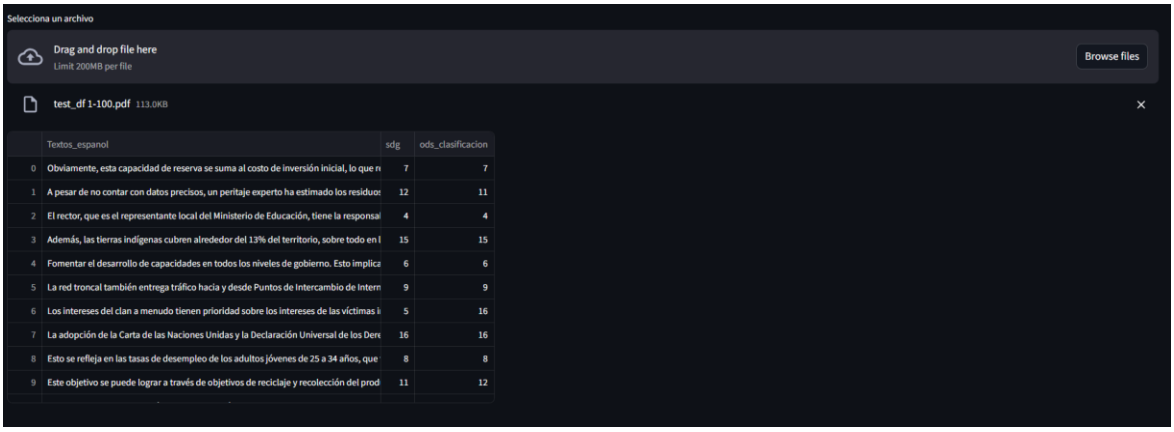
Figura 6: Clasificación de un texto en la página web

Como se puede ver en el texto, este puede tratar sobre diversos Objetivos de Desarrollo Sostenible, dado que toca varios temas. A pesar de que este texto tiene la etiqueta ODS 5, el modelo obtiene como clasificación principal el ODS 16, y, a la vez, la segunda clasificación posible es ODS 5. Si nos detenemos a leer el texto, podemos ver que este habla tanto de la igualdad de género como de justicia y paz, por lo que resulta entendible que el modelo tenga cerca de un 50% de probabilidad de pertenencia a estas dos clasificaciones.

La segunda funcionalidad de la página web comprende en poder subir un archivo (PDF o Excel) y poder generar las clasificaciones de este mismo. Esta funcionalidad es útil al momento de tener muchos textos que clasificar y no tener que hacer la inserción de cada uno de ellos manualmente. Como se puede ver en la figura 7, se importó un archivo PDF el cual contiene una tabla cuya primera columna tiene los textos a clasificar. Debajo del

botón *Browser* se va a mostrar una la misma tabla del PDF con una columna extra que es la clasificación dada por el modelo. Esta tabla se puede descargar como CSV.

Figura 7: Clasificación de textos n un archivo PDF



4. Pruebas y resultados

Luego de obtener la red neuronal final, se usaron los datos de Test (20%) separados previo al entrenamiento para poder revisar la precisión del modelo obtenido, en la cual se obtuvieron los siguientes resultados:

Tabla 6: Resultados del modelo generado

ODS	precision	recall	f1-score
1	0,87	0,78	0,82
2	0,84	0,9	0,87
3	0,94	0,95	0,94
4	0,95	0,98	0,96
5	0,95	0,96	0,96
6	0,93	0,93	0,93
7	0,93	0,94	0,93
8	0,76	0,74	0,75
9	0,82	0,89	0,85
10	0,78	0,67	0,72
11	0,89	0,89	0,89
12	0,91	0,89	0,9
13	0,9	0,85	0,88
14	0,95	0,97	0,96
15	0,91	0,93	0,92
16	0,96	0,98	0,97
accuracy	0,9115	0,912	0,9115
macro avg	0,89	0,89	0,89
weighted avg	0,91	0,91	0,91

En esta tabla se puede ver que la mayoría de las categorías, tienen unos valores de precisión y recall encima de 0.9. Lo que significa que el modelo es bastante acertado al momento de clasificar estas categorías. Además, la exactitud general fue de 91.15%.

También se puede ver que el recall general del modelo es de 89.21%. Es decir, de las clasificaciones originales de los ODS, el modelo clasifica bien este porcentaje. Por último, también se ve que la precisión general fue de 89.52%. Es decir, de las clasificaciones dadas por el modelo, este porcentaje es correcto. Gracias a esto podemos decir que aproximadamente el modelo tiene un valor de acierto del 90%.

Sin embargo, uno de los objetivos de este proyecto es ver cómo la aumentación de textos puede hacer variar las métricas de las categorías cuyas cantidades de datos totales son muy pocas en comparación a las demás. Para esto utilizaré las métricas obtenidas en el trabajo de Wilches [9] para poder comparar los resultados obtenidos:

Tabla 7: Comparación de resultados obtenidos sin aumentación de datos

ODS	precisión sin aug	precision	recall sin aug	recall
1	0.88	0,87	0.80	0,78
2	0.87	0,84	0.85	0,9
3	0.92	0,94	0.95	0,95
4	0.95	0,95	0.95	0,98
5	0.93	0,95	0.96	0,96
6	0.90	0,93	0.94	0,93
7	0.90	0,93	0.92	0,94
8	0.79	0,76	0.65	0,74
9	0.71	0,82	0.84	0,89
10	0.59	0,78	0.70	0,67
11	0.89	0,89	0.88	0,89
12	0.78	0,91	0.71	0,89
13	0.88	0,9	0.81	0,85
14	0.96	0,95	0.96	0,97
15	0.88	0,91	0.87	0,93
16	0.95	0,96	0.96	0,98
accuracy	No Data	0,9115	0.89	0,912

Nota: Tabla extraída y modificada de [9]

En esta última tabla se puede observar que en la mayoría de las categorías se logró una mejoría con respecto a la anterior. Además de esto, los ODS que se encuentran subrayadas fueron a los que se les realizó la aumentación de datos. En estos ODS específicamente, podemos ver una gran mejoría en los valores de precisión y recall, especialmente en la precisión del ODS 10 que pasó de 0.59 a 0.78. Es decir, aumento en un casi 20%. A pesar de esto, también se puede ver que hay algunas categorías que bajaron su rendimiento. Sin embargo, en ninguno de los casos hubo un empeoramiento superior a 3 puntos porcentuales, es decir, no es muy significativo a cambio de obtener mejores métricas en todos los demás, especialmente en aquellos que fueron aumentados.

5. Conclusiones y trabajo futuro

5.1 Conclusiones

Se evidencia una completa mejoría en las distintas métricas (precisión y recall) de las 4 categorías en las cuales se realizó una aumentación de datos (9, 10, 12 y 15). Superando los modelos anteriores que no implementaron estrategias de aumentación de textos para balancear las clases.

El modelo logró una precisión global del 91.15%. Lo que demuestre la alta eficacia en la combinación de las diferentes tecnologías utilizadas (Aumentación de textos, modelos pre entrenados y redes neuronales).

Se concluye que, para textos de muchas palabras, las librerías de Open Source no son las mejores opciones para la aumentación de datos, dado que los textos generados no varían en más de 4 palabras al texto original, por lo que para estos casos es mejor el uso de herramientas y APIs más sofisticadas como OpenAI.

5.2 Trabajo futuro

Teniendo en cuenta la limitación vista en la herramienta de OpenAI para poder utilizar todo el texto original para generar los nuevos textos, sin cortar la última frase, se recomienda trabajar en la aumentación de datos con otros modelos más avanzados de la misma herramienta como *Davinci-003* o *GPT 3.5,* los cuales generan textos más robustos a un mayor precio, posiblemente sin la limitación del máximo de tokens.

Debido a que fue necesario incluir distintas formas y tecnologías para evitar la sobre especialización en la red neuronal final, se recomienda también hacer un análisis más profundo en la arquitectura de esta, con el fin de obtener mejores resultados con datos que el modelo desconozca (no utilizados en el entrenamiento.

Bibliografía

[1] "¿Qué son las Naciones Unidas? Información completa sobre la ONU". Consultado: el 12 de enero de 2024. [En línea]. Disponible en: https://www.ferrovial.com/es/recursos/naciones-unidas/

[2] Envera, "Agenda 2030: así contribuye Envera a once Objetivos de Desarrollo Sostenible", https://grupoenvera.org/agenda-2030-asi-contribuye-envera-once-los-objetivos-desarrollo-sostenible/?gclid=Cj0KCQiAhomtBhDgARIsABcaYynI1x9evifEfT33cibr13AyfHouzCJKY69pWhCqOm6xPkR_2Gam32saAtNcEALw_wcB#anchor.

[3] Programa de las Naciones Unidas para el Desarrollo, "¿Qué son los Objetivos de Desarrollo Sostenible?", https://www.undp.org/es/sustainable-development-goals.

[4] Infraestructura Colombiana de Datos Espaciales, "ODS - Objetivos Desarrollo Sostenible, Agenda 2030 Colombia", https://www.icde.gov.co/datos-y-recursos/articulacion-icde/ods-objetivos-desarrollo-sostenible-agenda-2030-colombia.

[5] Iberdrola, "¿Qué es la Inteligencia Artificial?", https://www.iberdrola.com/innovacion/que-es-inteligencia-artificial.

[6] Iberdrola, "Qué es el 'machine learning'", https://www.iberdrola.com/innovacion/machine-learning-aprendizaje-automatico.

[7] IBM, "¿Qué es el deep learning?", https://www.ibm.com/mx-es/topics/deep-learning.

[8] Universidad EAFIT, "Inteligencia artificial: de cara al logro de los ODS", https://www.eafit.edu.co/noticias/revistauniversidadeafit/173/inteligencia-artificial-logro-ods.

[9] Paula Andrea Wilches Castellanos, "Uso de la inteligencia artificial para apoyar el análisis de opiniones en procesos de planeación participativa y su traducción al lenguaje de los objetivos de desarrollo sostenible.", Universidad de los Andes, Bogotá, 2023.

[10] "Procesamiento del Lenguaje Natural". Consultado: el 14 de enero de 2024. [En línea]. Disponible en: https://www.iic.uam.es/inteligencia-artificial/procesamiento-del-lenguaje-natural/

[11] J. D. Polo, "CÓMO FUNCIONA BERT, EL MODELO DE PROCESAMIENTO DEL LENGUAJE NATURAL DE GOOGLE", https://wwwhatsnew.com/2023/03/21/como-funciona-bert-el-modelo-de-procesamiento-del-lenguaje-natural-de-google/.

[12] G. Espíndola, “¿Qué son los embeddings y cómo se utilizan en la inteligencia artificial con python?”, https://gustavo-espindola.medium.com/qu%C3%A9-son-los-embeddings-y-c%C3%B3mo-se-utilizan-en-la-inteligencia-artificial-con-python-45b751ed86a5.

[13] Microsoft, “Ajuste de hiperparámetros de un modelo (v2)”, https://learn.microsoft.com/es-es/azure/machine-learning/how-to-tune-hyperparameters?view=azureml-api-2.

[14] Amazon, “¿Qué es una red neuronal?”, https://aws.amazon.com/es/what-is/neural-network/.

[15] IBM, “¿Qué son las redes neuronales?”, https://www.ibm.com/es-es/topics/neural-networks.

[16] J. Barrios, “La matriz de confusión y sus métricas”, https://www.juanbarrios.com/la-matriz-de-confusion-y-sus-metricas/.

[17] “GitHub - makcedward/nlpaug: Data augmentation for NLP”. Consultado: el 14 de enero de 2024. [En línea]. Disponible en: https://github.com/makcedward/nlpaug?tab=readme-ov-file

[18] “¿Qué es TF-IDF y cómo se calcula? - Seobility Wiki”. Consultado: el 18 de enero de 2024. [En línea]. Disponible en: https://www.seobility.net/es/wiki/TF_IDF#.C2.BFQu.C3.A9_es_TF-IDF.3F

[19] M. Terol, “API de OpenAI: revolucionando la industria de chatbots”, https://blogthinkbig.com/api-de-openai#:~:text=Qu%C3%A9%20posibilidades%20ofrece%20la%20API%20de%20OpenAI&text=El%20modelo%20tiene%20caracter%C3%ADsticas%20que,de%20voz%20como%20de%20texto.

[20] Núñez Vilma, “¿Qué es un prompt? Descripción de su importancia en la inteligencia artificial”, https://vilmanunez.com/que-es-un-prompt-inteligencia-artificial/.

[21] “Models - OpenAI API”. Consultado: el 21 de enero de 2024. [En línea]. Disponible en: https://platform.openai.com/docs/models/moderation