Diseño e implementación de un bot de charla basado en GPT para la acreditación internacional de los programas de pregrado de la E3T

Mateo Rueda Rodriguez, Cristian Alfonso Hernández Prince

Trabajo de Grado para Optar al Título de Ingeniero Electrónico

Director

Homero Ortega Boada

Doctor en ciencias de la ingeniería, radiocomunicaciones

Codirector

Oscar Arnulfo Quiroga Quiroga

Doctor en Tecnología Industrial

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones

Ingeniería Electrónica

Bucaramanga

2024

Dedicatoria

Este proyecto está dedicado a mi familia, cuya inspiración y apoyo han sido la fuerza que me impulsa y están ahí detrás de cada logro alcanzado. Su influencia ha dejado una marca indeleble en este trabajo, y esta dedicación refleja mi profundo agradecimiento por su impacto en mi trayectoria académica.

A mis padres Divertel Rueda y Sandra Patricia Rodriguez por su apoyo incondicional incluso en momentos muy difíciles en estos últimos años.

A mi hermano, Lucas Rueda por su compañía y apoyo en estos años, y también al resto de mi familia, abuelos, tíos y primos por el apoyo incondicional que me han dado.

Mateo Rueda Rodriguez

Dedico este proyecto primeramente a mis padres, Fernando Alfonso Hernández Rojas y Maria Estella Prince Angarita, cuyo apoyo incondicional ha sido pilar clave en mi travesía universitaria. Su guía y amor han sido fundamental para alcanzar este hito académico.

A mis hermanos y familiares cercanos, quienes con sus palabras alentadoras me motivaron a seguir avanzando y perseguir mis metas.

Asimismo, dedico este trabajo a la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T) y a sus docentes por brindarme la oportunidad de crecer académicamente y por contribuir significativamente a mi formación profesional.

Cristian Alfonso Hernández Prince

Agradecimientos

Quisiéramos expresar nuestro sincero agradecimiento a todas las personas que contribuyeron de manera significativa a la realización de este proyecto. Agradecemos a nuestro director de proyecto Homero Ortega Boada por su orientación valiosa y su compromiso constante.

Nuestra gratitud se extiende a nuestro codirector Oscar Arnulfo Quiroga , cuya colaboración fue fundamental para superar los desafíos.

También queremos reconocer el apoyo de amigos y familiares, cuyo aliento incondicional fue un pilar esencial durante todo el proceso.

Tabla de Contenido

Pág.

Lista de Tablas

Pág.

Lista de Figuras

Pág.

Lista de Apéndices

pág.

Los apéndices están adjuntos y puede visualizarlos en la base de datos de la biblioteca UIS

Glosario

ABET: la acreditación ABET es una prueba de que un programa universitario ha cumplido con los estándares esenciales para producir graduados listos para entrar en los campos críticos de la ciencia aplicada, la informática, la ingeniería, la tecnología y la ingeniería.(ACOFI,2023)

API: las API son mecanismos que permiten a dos componentes de software comunicarse entre sí mediante un conjunto de definiciones y protocolos.(Amazon)

AWS: acrónimo de Amazon Web Services

Colab: Colaboratory, o "Colab" para abreviar, es un producto de Google Research. Permite a cualquier usuario escribir y ejecutar código arbitrario de Python en el navegador. (Google)

E3T: acrónimo de Escuela de Ingenierías Eléctrica, Electrónica y de Telecomunicaciones.

GPT: los transformadores generativos preentrenados, comúnmente conocidos como GPT, son una familia de modelos de redes neuronales que utilizan la arquitectura de transformadores y representan un avance clave en la inteligencia artificial (IA) que impulsa las aplicaciones de IA generativa, como ChatGPT. (Amazon)

HTML: Lenguaje de Marcas de Hipertexto, del inglés HyperText Markup Language, es el componente más básico de la Web. Define el significado y la estructura del contenido web.(Mozilla, 2023a)

Javascript: es un lenguaje de programación basada en prototipos, multiparadigma, de un solo hilo, dinámico, con soporte para programación orientada a objetos, imperativa y declarativa.(Mozilla, 2023b)

JSON: se describe el formato JSON (JavaScript Object Notation).Es un formato ligero de intercambio de datos. JSON es de fácil lectura y escritura para los usuarios. (IBM,2022)

NLP: el procesamiento del lenguaje natural (NLP, por sus siglas en inglés) es una rama de la inteligencia artificial que ayuda a las computadoras a entender, interpretar y manipular el lenguaje humano.(SAS)

Prompt: es una instrucción o texto inicial que se le proporciona a una herramienta de IA generativa para guiar su generación de respuestas o resultados, según los formatos en los que se especialice la herramienta.(Qué es un prompt en ia Y Para Qué sirve (+ ejemplos))

Python: es un lenguaje de programación ampliamente utilizado en las aplicaciones web, el desarrollo de software, la ciencia de datos y el machine learning.(Amazon)

Resumen

Título: Diseño e implementación de un bot de charla basado en GPT para la acreditación internacional de los programas de pregrado de la E3T.[1*]

Autor: Mateo Rueda Rodriguez, Cristian Alfonso Hernández Prince[2*]

Palabras Clave: GPT, Chatbot, Inteligencia Artificial

Descripción: La Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T) requiere elevar el nivel de madurez del modelo de calidad de sus programas de formación de pregrado en un plazo de 5 años y demostrar que los programas viven un proceso de mejoramiento continuo, para mantener su acreditación internacional ABET. Sin embargo, los actores del proceso de mejoramiento continuo tienen dificultades para acceder a información actualizada sobre el proceso que vive la E3T, lo que obstaculiza su contribución al proceso de mejora. La causa del problema radica en que la información disponible es cambiante, lo que dificulta la actualización de la información en la página web de la E3T.

El proyecto se centró en el desarrollo de un bot de charla basado en GPT para la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T), con énfasis en la integración de información actualizada, mecanismos de retroalimentación y una interfaz de usuario intuitiva. La innovación radicó en la utilización de tecnología de procesamiento del lenguaje natural para proporcionar respuestas precisas a preguntas en constante cambio sobre el proceso de mejora continua de la E3T. Este enfoque no solo abordó desafíos de acceso a información actualizada, sino que también sentó las bases para futuros avances en la integración de inteligencia artificial en el ámbito educativo y organizacional.

[1*] Trabajo de Grado de Investigación

[2**] Facultad de ingeniería Fisicomecánicas. Escuela de ingenierías eléctrica, electrónica y de telecomunicaciones. Ingeniería electrónica. Director: Homero Ortega Boada. Doctor en ciencias de la ingeniería, radiocomunicaciones. Codirector: Oscar Arnulfo Quiroga Quiroga. Doctor en tecnología

Abstract

Title: Design and implementation of a GPT-based chatbot for the international accreditation of the undergraduate programs at E3T.[3*]

Author(s): Mateo Rueda Rodriguez, Cristian Alfonso Hernández Prince[4]

Key Words: GPT, Chatbot, Artificial intelligence

Description: The School of Electrical, Electronics, and Telecommunications Engineering (E3T) needs to raise the maturity level of the quality model of its undergraduate programs within a 5-year period and demonstrate that the programs undergo a continuous improvement process to maintain their ABET international accreditation. However, stakeholders in the continuous improvement process struggle to access updated information on the ongoing process at E3T, hindering their contribution to the improvement process. The root cause of the problem lies in the changing nature of available information, making it difficult to update the information on E3T's website.

The project focused on developing a GPT-based chatbot for the School of Electrical, Electronics, and Telecommunications Engineering (E3T), emphasizing the integration of updated information, feedback mechanisms, and an intuitive user interface. The innovation lay in using natural language processing technology to provide accurate answers to constantly changing questions about E3T's continuous improvement process. This approach not only addressed challenges in accessing updated information but also laid the groundwork for future advancements in integrating artificial intelligence in the educational and organizational domains.

[3*] Research degree work

[4] Faculty of Physicomechanical Engineering. School of Electrical, Electronic, and Telecommunication Engineering. Electronic Engineering. Director: Homero Ortega Boada. PhD in Engineering Sciences, Radiocommunications. Co-Director: Oscar Arnulfo Quiroga Quiroga. PhD on Technology

Introducción

Actualmente, la inteligencia artificial y las tecnologías de procesamiento del lenguaje natural (NLP) revolucionan el ámbito empresarial y académico. Estas tecnologías tienen el potencial de mejorar significativamente los procesos de acreditación y calidad en instituciones educativas, como la Escuela de Ingeniería Eléctrica, Electrónica y de Telecomunicaciones (E3T). Este proyecto explora cómo la API GPT de OpenAI puede ayudar a la E3T a enfrentar desafíos relacionados con la acreditación y la gestión de la calidad.

El proyecto se enfoca en mejorar el acceso de diferentes actores a la información actualizada de la E3T, especialmente en relación con el proceso de acreditación ABET. Se reconoce que las dificultades de acceso a esta información pueden tener diversas causas, como oportunidades de mejora en la planificación, resistencia al cambio, falta de capacitación y limitaciones en la emisión de normas y procesos.

El objetivo del proyecto es diseñar e implementar un Bot de charla con inteligencia artificial y basado en GPT capaz de leer la documentación que la E3T tiene disponible sobre el proceso de acreditación ABET y a partir de esa información responder a preguntas.para esto será necesario recopilar la información y documentación que posee la E3T sobre la acreditación ABET.

Este bot de charla tendrá que entrenar el modelo a partir de la información procesada y realizar un proceso de verificación para garantizar la calidad y confiabilidad del bot. Esto implica asegurarse de que el bot proporcione respuestas precisas y actualizadas, evitando información contradictoria o falsa y establecer un mecanismo de actualización y mantenimiento de la información para garantizar que los administradores puedan reentrenar el modelo a partir de documentación actualizada y recibir indicaciones de respuestas insatisfactorias.

También se evaluaran los costos y beneficios de utilizar la tecnología GPT de OpenAI en comparación con otras opciones disponibles en el mercado.Con esto determinar si la robustez y capacidades de GPT justifican su costo en comparación con alternativas como los modelos de procesamiento de lenguaje natural

Finalmente se implementará el bot de charla por medio de un servicio web accesible en la página de ABE3T.

La solución que proponemos se fundamenta en la API de GPT de OpenAI, una herramienta poderosa que nos permite entrenar al sistema con la documentación disponible y responder a diversas solicitudes de información. Aunque nuestro enfoque inicial se dirige hacia la información de acreditación ABET, comprendemos que este proyecto es solo el primer paso hacia aplicaciones más amplias y transformadoras.

La tecnología de NLP y los chatbots están emergiendo como herramientas disruptivas en el ámbito de la inteligencia artificial. Las API de OpenAI, particularmente GPT, son líderes en este campo y ofrecen posibilidades infinitas para la creación de soluciones innovadoras.

A medida que avanzamos, no solo buscamos proporcionar un acceso más fácil y actualizado a la información sobre el proceso de mejora continua de la E3T, sino también abrir nuevas oportunidades en diversos campos, desde la optimización de los programas educativos hasta la redefinición de los estándares de calidad en la educación superior.

Nuestro proyecto representa un paso audaz hacia un futuro donde la tecnología no solo mejora la eficiencia de los procesos educativos, sino que también impulsa la excelencia y la innovación en la gestión de la calidad. Estamos comprometidos a desatar el potencial completo de la inteligencia artificial para transformar la educación y construir un futuro más prometedor para las generaciones venideras.La versatilidad y el rendimiento demostrados por GPT en

aplicaciones similares nos brindaron la confianza necesaria para avanzar con esta elección. Además, la API GPT proporcionó una interfaz amigable y robusta que facilitó el proceso de desarrollo.

Después de un riguroso proceso de validación, los resultados obtenidos reflejan la eficacia y la promesa de nuestra solución. Durante las pruebas, nuestro sistema demostró una capacidad excepcional para comprender y responder . La precisión y relevancia de las respuestas generadas confirmaron la coherencia y la adecuación de las soluciones proporcionadas.Los resultados del proceso de validación respaldan la efectividad de nuestra solución y subrayan su capacidad para abordar las necesidades cambiantes de la E3T en su búsqueda de la excelencia académica y la mejora continua.

1.Marco Teórico

Este capítulo busca preparar al lector para comprender fácilmente todo este informe. Se emprenderá una exploración exhaustiva del fascinante mundo del Procesamiento de Lenguaje Natural (NLP), desentrañando sus fundamentos, capacidades, y su aplicación concreta en el ámbito de este proyecto. Se ahondará en la comprensión de cómo el NLP posibilita la interacción fluida entre las máquinas y el lenguaje humano, destacando sus potencialidades y los contextos específicos en los que se despliega con mayor eficacia.

Además, se abordará detalladamente la elección y aplicación de las herramientas esenciales para la implementación exitosa de este proyecto. Entre estas herramientas, destaca la poderosa API de GPT, cuyo uso revoluciona la generación de texto y el entendimiento contextual. Asimismo, se explorará en profundidad la librería LangChain, una pieza fundamental que simplifica la interoperabilidad entre diversas bibliotecas de procesamiento de lenguaje natural, facilitando la integración de múltiples tecnologías. Para completar este tríptico tecnológico, se explorarán las bases de datos vectoriales, una innovación que redefine la gestión de información al almacenar datos de manera vectorial, proporcionando una estructura eficiente y dinámica.

1.1 Procesamiento del Lenguaje Natural (NLP)

El Procesamiento del Lenguaje Natural (NLP, por sus siglas en inglés, Natural Language Processing) es una rama de la inteligencia artificial (IA) que se enfoca en la interacción entre las computadoras y el lenguaje humano. Su objetivo principal es permitir que las máquinas comprendan, interpreten y generen lenguaje humano de manera similar a como lo hacen los seres

humanos. El NLP ha avanzado significativamente en las últimas décadas gracias a los avances en el aprendizaje profundo y el procesamiento de lenguaje natural basado en modelos como GPT (Generative Pre-trained Transformer). Estos modelos han demostrado ser capaces de realizar tareas de procesamiento de lenguaje humano con un alto grado de precisión, lo que ha impulsado su aplicación en una amplia variedad de campos, desde la atención médica y la educación hasta la investigación y el comercio electrónico.

1.1.1 Comprensión del Lenguaje

En el corazón del NLP se encuentra la capacidad de una máquina para comprender el lenguaje humano. Esto incluye la comprensión de la gramática, la semántica (el significado de las palabras y frases) y la pragmática (el contexto en el que se utiliza el lenguaje).

1.1.2 Procesamiento de Texto

El NLP implica la capacidad de procesar y analizar texto escrito o hablado. Esto puede incluir tareas como dividir el texto en palabras o tokens, identificar partes del discurso (como sustantivos o verbos) y analizar la estructura gramatical de las oraciones.

1.1.3 Generación de Lenguaje

Además de la comprensión, el NLP también se ocupa de la generación de lenguaje humano. Esto implica la creación de texto coherente y contextual a partir de datos o instrucciones dadas. Por ejemplo, generar respuestas automáticas de chatbots o traducir texto de un idioma a otro.

1.2 Fundamentos de NLP

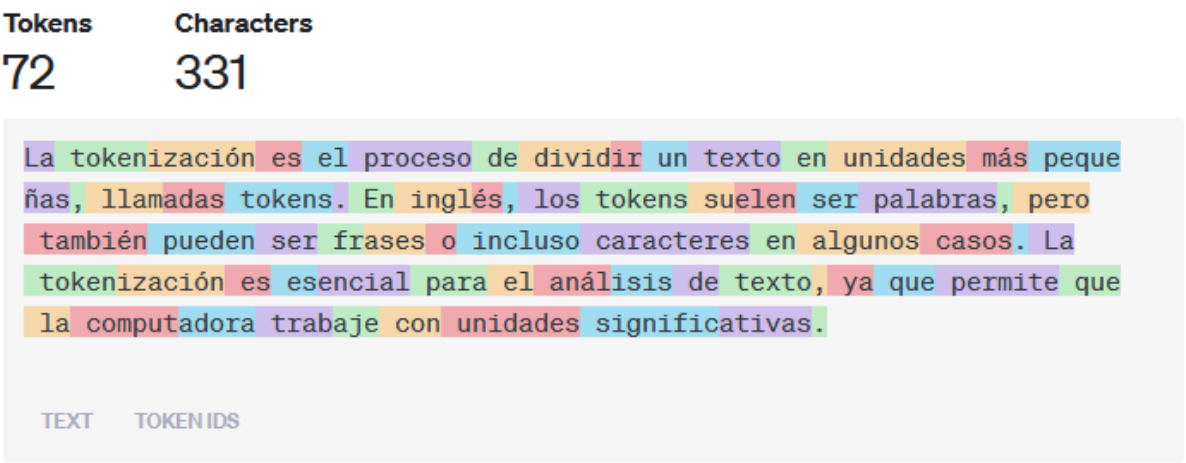
Los fundamentos del Procesamiento del Lenguaje Natural (NLP) se basan en los principios y conceptos clave que permiten a las computadoras entender y trabajar con el lenguaje humano de manera efectiva.

1.2.1 Tokenización

La tokenización es el proceso de dividir un texto en unidades más pequeñas, llamadas tokens. En inglés, los tokens suelen ser palabras, pero también pueden ser frases o incluso caracteres en algunos casos. La tokenización es esencial para el análisis de texto, ya que permite que la computadora trabaje con unidades significativas.

Figura 1

Ejemplo de Tokenización



Nota. Tomado de https://platform.openai.com/tokenizer

1.2.2 Procesamiento Morfológico

El procesamiento morfológico se refiere al análisis de las palabras en términos de su estructura y forma. Esto incluye identificar las raíces de las palabras (lema) y sus formas flexionadas (por ejemplo, "correr" y "corriendo"). El análisis morfológico es útil para tareas como la búsqueda y recuperación de información.

1.2.3 Análisis Sintáctico

El análisis sintáctico implica comprender la estructura gramatical de las oraciones. Esto incluye identificar partes del discurso (sustantivos, verbos, adjetivos, etc.) y cómo se relacionan en una oración. El árbol de análisis sintáctico es una representación gráfica de la estructura gramatical de una oración.

1.2.4 Análisis Semántico

El análisis semántico se enfoca en el significado de las palabras y cómo se combinan para formar significados más complejos en oraciones y documentos. Esto implica comprender las relaciones entre palabras y cómo se utilizan en diferentes contextos.

1.2.5 Modelado de Idioma

Los modelos de lenguaje son representaciones matemáticas que capturan la probabilidad de que una secuencia de palabras aparezca en un idioma. Estos modelos son esenciales para tareas como la predicción de palabras (autocompletar), traducción automática y generación de texto.

1.2.6 Análisis de Sentimiento

El análisis de sentimiento es una aplicación común del NLP que implica determinar si un texto expresa una opinión positiva, negativa o neutral. Esto es útil para comprender la actitud de las personas hacia un tema en las redes sociales, reseñas de productos, etc.

1.2.7 Modelos de Lenguaje Preentrenados

Los modelos de lenguaje preentrenados, como GPT (Generative Pre-trained Transformer), son modelos de aprendizaje automático que se entrenan en grandes cantidades de texto antes de ser afinados para tareas específicas. Estos modelos han impulsado avances significativos en el NLP.

1.2.8 Aprendizaje Profundo

El aprendizaje profundo (deep learning) es una técnica de aprendizaje automático que se utiliza ampliamente en el NLP. Las redes neuronales profundas permiten a las máquinas aprender representaciones más complejas y abstractas del lenguaje humano.

1.2.9 Evaluación de Modelos

La evaluación de modelos NLP es fundamental para medir su rendimiento. Se utilizan métricas como la precisión, la recuperación y el F1-score para evaluar la capacidad de un modelo para realizar tareas específicas.

1.2.10 Diversidad de Idiomas

El NLP se aplica a una amplia variedad de idiomas, y los desafíos pueden variar según el idioma. Algunos idiomas tienen recursos limitados, lo que puede requerir enfoques especiales en NLP.

1.3 Aplicaciones de NLP en la Actualidad

El Procesamiento del Lenguaje Natural (NLP) ha experimentado un crecimiento significativo debido a los avances tecnológicos, siendo fundamental en diversas aplicaciones. Desde la utilización de asistentes virtuales y chatbots para interactuar en lenguaje natural, hasta la traducción eficiente de textos con plataformas como Google Translate, el NLP ha transformado la forma en que nos comunicamos globalmente. Además, en el ámbito empresarial, las empresas emplean el NLP para analizar el sentimiento de los clientes en redes sociales,

realizar búsquedas semánticas en motores de búsqueda, resumir automáticamente documentos extensos y clasificar información de manera eficiente.

En campos especializados, como la atención médica, el NLP facilita la transcripción y análisis de registros médicos electrónicos, permitiendo a los profesionales de la salud tomar decisiones más informadas. En educación, el NLP se utiliza para desarrollar aplicaciones de tutoría en línea, corregir ensayos y ofrecer retroalimentación personalizada. Finalmente, en el ámbito legal, la tecnología NLP es esencial para la automatización de servicios legales, analizando documentos y contratos, así como asistiendo en investigaciones legales. Estas aplicaciones demuestran la versatilidad del NLP y su impacto transformador en diversas industrias.

1.4 API GPT de OpenAI

Esta herramienta permite acceso al modelo de lenguaje basado en aprendizaje profundo que ha sido entrenado en grandes cantidades de texto en línea, lo que le permite comprender y generar texto coherente y contextual en lenguaje humano.

Esta API se utiliza para crear aplicaciones que generen respuestas a preguntas, escriban contenido, generen diálogos de chatbots, etc. Además, la API permite afinar el modelo GPT para tareas específicas mediante el entrenamiento con datos personalizados, lo que la hace altamente adaptable a diferentes aplicaciones.

La API GPT también es conocida por su capacidad de mantener conversaciones contextuales con el modelo, esto significa que se puede enviar una serie de mensajes en lugar de una sola solicitud, lo que facilita la creación de chatbots y asistentes virtuales más interactivos, también admite varios idiomas y dialectos, lo que la hace versátil para aplicaciones multilingües.

El acceso a la API GPT generalmente se realiza a través de una suscripción que incluye un límite de tokens (unidades de texto procesadas) por mes, y OpenAI proporciona documentación detallada y recursos de soporte para ayudar a los desarrolladores a utilizar la API de manera efectiva. La API GPT de OpenAI sigue mejorando con actualizaciones continuas y nuevos modelos, lo que la convierte en una herramienta versátil y en constante evolución para aplicaciones basadas en lenguaje.

1.4.1 Capacidades de GPT

1.4.1.1 Representación de Palabras como Vectores

GPT convierte cada palabra en una representación matemática llamada "embedding" o incrustación, que es un vector numérico. Estos vectores representan el significado de las palabras y su contexto en relación con otras palabras en la oración o el texto.

1.4.1.2 Generación de Texto

Una vez que GPT está pre-entrenado, se puede utilizar para generar texto. Dado un contexto inicial o una frase parcial (como embedding), GPT genera texto coherente y contextualmente relevante como continuación. Puede completar oraciones, escribir párrafos enteros o incluso responder preguntas.

1.5 Langchain

Langchain es un marco de trabajo (framework) para desarrollar aplicaciones basadas en modelos de lenguaje. Permite crear aplicaciones que son conscientes del contexto y que pueden razonar utilizando modelos de lenguaje.Las bibliotecas de Langchain son bibliotecas en Python y JavaScript que contienen interfaces y integraciones para diferentes componentes, así como implementaciones predefinidas de cadenas y agentes.(Langchain)

1.5.1 Cadenas

Una cadena (chain) en el contexto de Langchain es una secuencia de acciones que se ejecutan en orden para lograr un objetivo específico. En una cadena, las acciones están codificadas directamente en el código y se ejecutan de manera secuencial.(Langchain)

1.5.2 Agentes

Un agente (agent) en Langchain es un modelo de lenguaje que se utiliza como motor de razonamiento para determinar qué acciones tomar y en qué orden. A diferencia de las cadenas, los agentes utilizan un modelo de lenguaje para tomar decisiones y determinar las acciones a seguir. Los agentes pueden utilizar diferentes tipos de modelos de lenguaje.(Langchain)

1.6 Base de datos vectorial

Una base de datos vectorial es un tipo de base de datos diseñada específicamente para almacenar y manipular datos en forma de vectores. A diferencia de las bases de datos tradicionales que almacenan datos en forma de filas y columnas, una base de datos vectorial almacena y opera con datos en forma de vectores multidimensionales.

Una base de datos vectorial es importante en la creación de un bot de charla porque permite una búsqueda eficiente, una recuperación rápida de información, un almacenamiento compacto y una escalabilidad adecuada para manejar grandes volúmenes de datos y consultas simultáneas. Esto ayuda a mejorar la experiencia del usuario y la eficiencia del bot de charla.

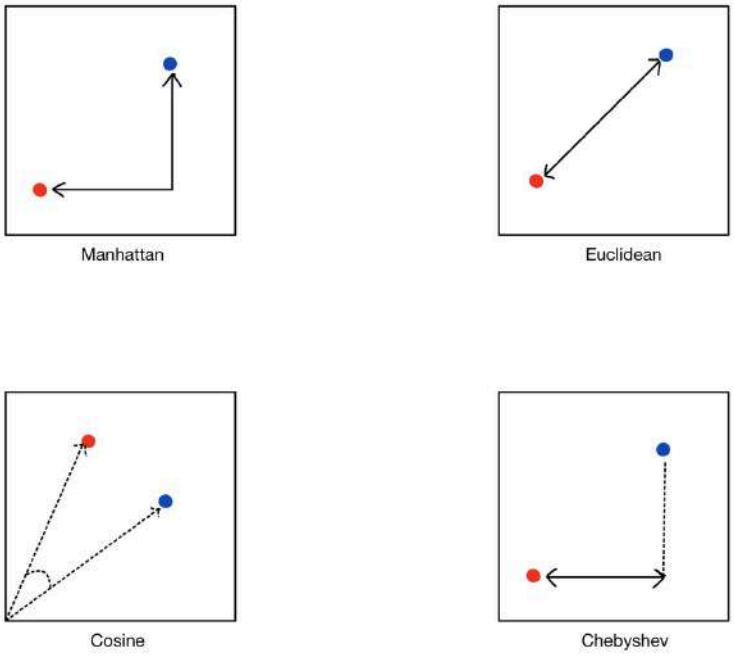
1.6.1 Consulta de Similitud

Cuando se realiza una pregunta, la pregunta se convierte en un embedding utilizando el mismo modelo de lenguaje.Luego, este embedding de consulta se compara con los embeddings almacenados en la base de datos utilizando la métrica de similitud adecuada, como la distancia

euclidiana o la similitud coseno.En el caso de embeddings de OpenAI para procesamiento de lenguaje natural, la similitud coseno es comúnmente preferida. Esto se debe a que la similitud coseno tiende a ser más robusta en entornos donde las palabras o documentos pueden tener longitudes variables, y la dirección en el espacio vectorial es más significativa que la magnitud absoluta.(Tripathi)

Figura 2

Diferentes métricas de similitud.



Nota. Tomado de: https://www.pinecone.io/learn/what-is-similarity-search/

1.6.2 Recuperación de Resultados

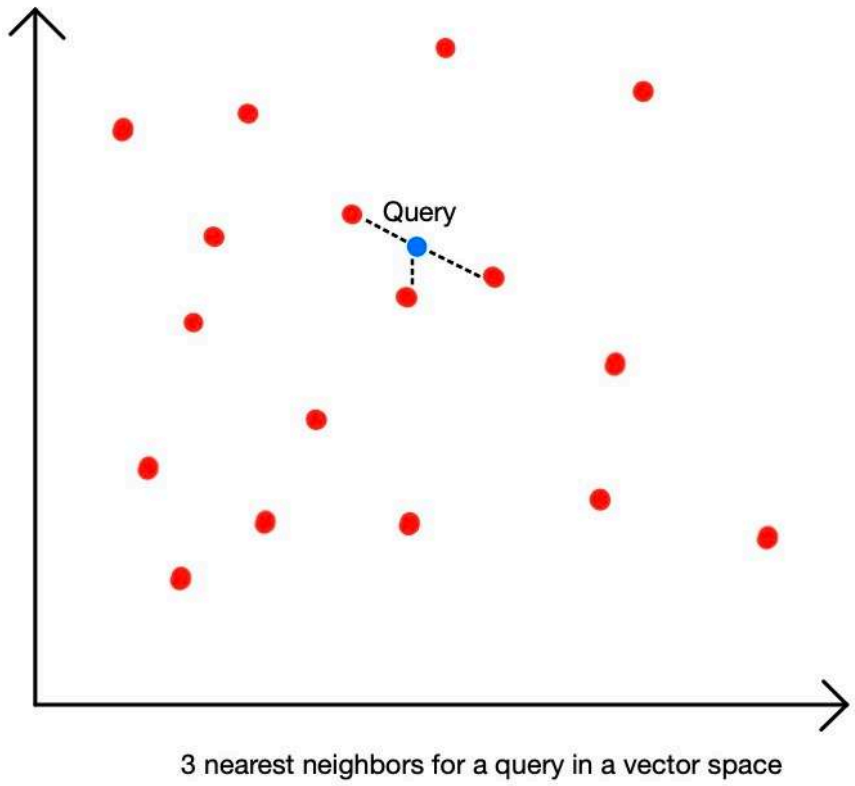
Los elementos de la base de datos se clasifican según su similitud con la consulta.Se devuelven los resultados más cercanos o similares a la pregunta original.

K vecinos más cercanos, o k-NN, es un algoritmo muy popular para encontrar vectores cercanos en un espacio dado un vector de consulta; Se puede realizar realizar k-NN en los vectores que tenemos para nuestros datos y recuperar los vecinos más cercanos para nuestro vector de consulta según la distancia entre los vectores.Una desventaja importante de k-NN es que, para encontrar los vectores más cercanos a nuestra consulta, se debe calcular su distancia con cada vector que tenemos en la base de datos.(Tripathi)

Para reducir la complejidad computacional agregada por una búsqueda exhaustiva como k-NN, se hace uso de la búsqueda aproximada de vecinos.En lugar de verificar distancias entre cada vector en la base de datos, se recupera una estimación del vecino más cercano.En la búsqueda aproximada de vecinos (ANN), se construyen estructuras de índices que reducen el espacio de búsqueda y mejoran los tiempos de búsqueda.Es suficiente entender que los algoritmos de ANN hacen uso de técnicas como el indexado, el agrupamiento, el hashing y la cuantización para mejorar significativamente la computación y el almacenamiento a costa de alguna pérdida de precisión.(Tripathi)

Figura 3

Muestra los 3 vectores vecinos más cercanos.



Nota. Tomado de: https://www.pinecone.io/learn/what-is-similarity-search/

1.6.3 Generación de la respuesta

GPT (Generative Pre-trained Transformer) genera respuestas a través de un proceso técnico complejo basado en arquitecturas de transformers. En esta sección se dará una explicación técnica paso a paso de cómo GPT genera respuestas a partir de un prompt. GPT no tiene una comprensión real del significado en el sentido humano; simplemente predice la siguiente palabra basándose en patrones aprendidos durante el preentrenamiento. El prompt inicial sirve como guía

para la generación de respuestas, y el modelo produce una continuación coherente en función de la información que ha asimilado durante su entrenamiento.

1.6.3.1 Tokenización

GPT se basa en la arquitectura Transformer, que utiliza capas de atención para procesar secuencias de tokens (palabras o partes de palabras). Las capas de atención permiten que el modelo se enfoque en diferentes partes de la entrada, capturando así relaciones a larga distancia..

1.6.3.2 Embedding de tokens

Cada token se representa mediante un vector de embeddings. Estos embeddings son valores numéricos que capturan la semántica y la relación entre los tokens.

1.6.3.3 Atención múltiple

La arquitectura Transformer utiliza capas de atención múltiple para procesar los embeddings de tokens. En estas capas, el modelo asigna ponderaciones a diferentes partes de la entrada, permitiéndole centrarse en las relaciones importantes.

1.6.3.4 Generación autoregresiva

GPT genera respuestas de manera autoregresiva, lo que significa que produce una palabra o token a la vez en función de las palabras anteriores. Comienza con el prompt proporcionado y genera continuaciones secuenciales.

1.6.3.5 Sampling o Greedy Decoding

Durante la generación de texto, el modelo puede seleccionar la siguiente palabra de varias maneras. Puede realizar un muestreo estocástico, que introduce aleatoriedad en la selección, o puede usar "decodificación codiciosa" (greedy decoding), donde simplemente elige la palabra más probable en cada paso.

1.6.3.6 Iteración

Este proceso se repite hasta que se ha generado la longitud deseada de la respuesta o se ha alcanzado algún otro criterio de finalización.

2. Desarrollo de la solución

En esta sección, nos sumergimos en la esencia misma del desarrollo de la solución, partiendo desde la identificación y comprensión de las necesidades de los usuarios hasta la articulación de una solución integral que aborde eficientemente cada uno de estos requerimientos.

A continuación, se presenta una visión panorámica de la solución general diseñada, destacando los elementos clave que la conforman y cómo estos se entrelazan para lograr una respuesta integral a los desafíos planteados. Posteriormente, se busca documentar todo el proceso seguido para el diseño de la solución.

2.1 Necesidades de Usuario

Los potenciales usuarios de este bot de charla orientado a la acreditación ABET en la Escuela de Ingeniería y Tecnología (E3T) tienen necesidades específicas y variadas que reflejan su involucramiento en el proceso de acreditación. Entre estos usuarios se encuentran estudiantes, profesores, personal administrativo y evaluadores de ABET, cada uno con inquietudes particulares.

Siguiendo la metodología de los Métodos Ágiles, las necesidades de los usuarios de la solución pueden ser identificadas a partir de historias de usuario.

Lista de historias de usuario:

Como estudiante,

- Quiero poder realizar búsquedas específicas en documentos académicos para encontrar información relevante de manera rápida y eficiente,
- Para mejorar mi comprensión de los requisitos de acreditación ABET y optimizar mi participación en el programa educativo.

Como profesor,

- Quiero acceder fácilmente a la documentación necesaria para entender los estándares de acreditación ABET,
- Para facilitar la preparación de mis cursos y garantizar que cumplan con los criterios establecidos por ABET.

Como evaluador de ABET,

- Quiero acceder a información detallada sobre los programas educativos de la E3T,
- Para realizar evaluaciones precisas y efectivas durante el proceso de acreditación, asegurando el cumplimiento de los estándares de calidad.

Como miembro del personal administrativo,

- El personal administrativo necesita información detallada sobre los procesos y requisitos administrativos asociados con la acreditación ABET. Preguntas sobre la documentación necesaria, el seguimiento de datos y la coordinación de actividades relacionadas con la acreditación son esenciales.

Como visitante interesado en la E3T,

- Quiero tener acceso fácil a información relevante sobre los programas acreditados por ABET, para tomar decisiones informadas sobre la posibilidad de matricularme o colaborar con la institución.

Como administrador del sistema,

- Quiero supervisar y optimizar el rendimiento de la solución implementada,
- Para garantizar la eficiencia operativa y la continuidad del servicio a medida que evolucionan las necesidades del departamento E3T.

- Quiero poder realizar actualizaciones y modificaciones en los documentos almacenados, para garantizar la consistencia y la precisión de la información disponible para todos los usuarios.

Dentro de los límites del presente proyecto, se ha realizado una cuidadosa delimitación de las historias de usuario para enfocar los esfuerzos en áreas específicas y garantizar una implementación efectiva. Las historias de usuario han sido acotadas a las siguientes necesidades clave:

Estudiantes: Los estudiantes buscan obtener información clara y accesible sobre el proceso de acreditación ABET, comprendiendo cómo este afecta su educación y la calidad de los programas académicos ofrecidos. Pueden tener preguntas sobre los estándares de calidad, la mejora continua y cómo la acreditación impacta su formación.

Profesores: Los profesores buscan orientación sobre cómo contribuir eficazmente al proceso de acreditación. Pueden tener preguntas sobre cómo medir, en donde están las rúbricas, cuales son, sobre los criterios de evaluación, la recopilación de evidencia y las mejores prácticas para cumplir con los estándares establecidos por ABET.

Personal Administrativo de la E3T: El personal administrativo necesita información detallada sobre los procesos y requisitos administrativos asociados con la acreditación ABET. Preguntas sobre la documentación necesaria, el seguimiento de datos y la coordinación de actividades relacionadas con la acreditación son esenciales.

Evaluadores de ABET: Los evaluadores de ABET buscan claridad sobre los estándares y expectativas específicas que guían su evaluación. Pueden tener preguntas sobre el proceso de revisión, la interpretación de criterios y la interacción con la institución educativa.

Administradores del sistema: Los administradores del sistema buscan que el sistema sea sencillo de mantener y actualizar.Necesitan documentación de cómo funciona el sistema, como se actualiza.

2.2 Solución General

En un futuro, la gestión documental educativa se transformará en un ecosistema fluido e inteligente, potenciado por tecnologías innovadoras que van más allá de la implementación actual. La visión se centra en ofrecer una experiencia integral y colaborativa, maximizando la eficiencia y el acceso a información crucial para estudiantes, profesores, evaluadores y cualquier interesado en el ámbito educativo.

Los motores de búsqueda basados en inteligencia artificial se perfeccionarán para proporcionar resultados más precisos y contextualmente relevantes. La capacidad de entender el lenguaje natural y el contexto específico permitirá a los usuarios obtener información específica de manera más rápida y eficiente.

La evolución de los asistentes virtuales, basados en modelos mejorados, proporcionará respuestas más precisas y personalizadas. Estos asistentes no solo responderán preguntas, sino que también ofrecerán sugerencias contextuales y guiarán a los usuarios a través de procesos específicos relacionados con la acreditación ABET.

2.3 Solución Particular

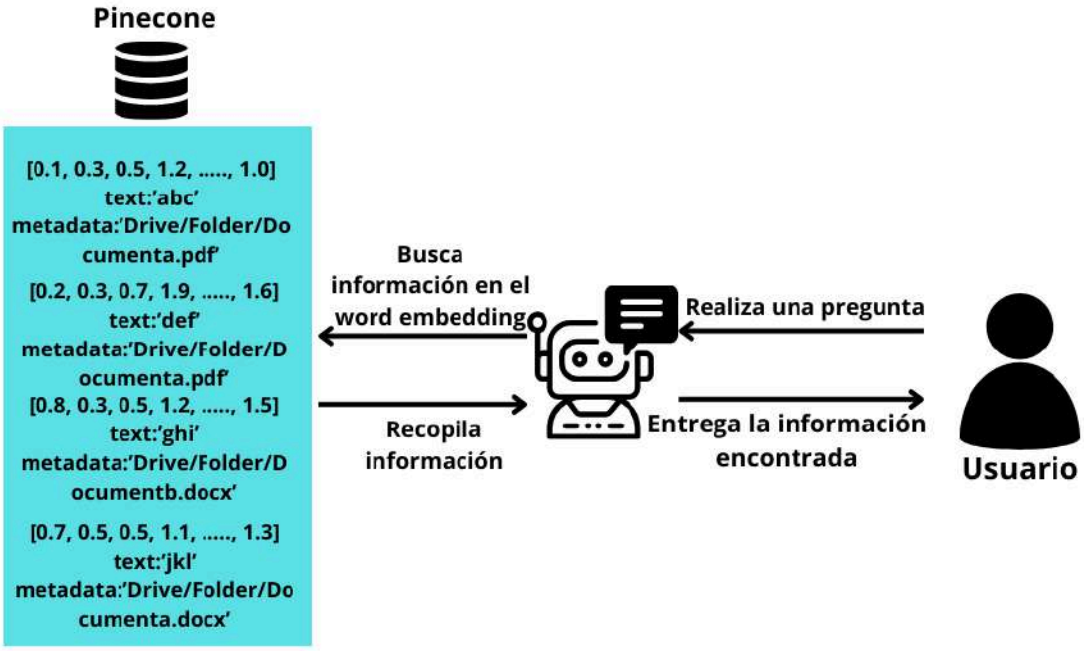
En el dinámico escenario donde la documentación ABET y los modelos de GPT están en constante evolución, reconocemos que la clave para el éxito de nuestro proyecto de bot de charla reside en la capacidad de adaptación continua. Esta solución se enfoca en la creación de un marco sólido que no solo responda a las necesidades actuales, sino que también esté hábilmente equipado para enfrentar los cambios futuros de manera ágil y eficiente.

La desarrollaremos implementando el código fuente de manera intuitiva para el equipo administrativo, otorgándoles la capacidad de realizar actualizaciones manuales en la base de datos según sea necesario. Este enfoque se complementa con guías de uso amigables, diseñadas para permitir ajustes específicos y mantener la agilidad del sistema incluso frente a cambios inesperados.

Además, nuestros esfuerzos se centran en la implementación de procedimientos de validación continua que garanticen la coherencia y precisión de las respuestas del bot. Este proceso integral incluirá casos de prueba específicos destinados a abordar cambios en las pautas de ABET y evaluaciones regulares del rendimiento del modelo. Este enfoque meticuloso no solo asegura que nuestro bot responda con precisión, sino que también se adapte con eficacia a los cambios en el entorno educativo y normativo.

Figura 4

Diagrama del la solución particular



2.4 Selección de plataformas

Una solución basada en GPT, básicamente, consiste en una aplicación que hace uso de una API, pero va más allá, ya que esa API proporciona solo el servicio de interpretación, pero no proporciona una base de datos para el almacenamiento de la respectiva data vectorial, tampoco, proporciona una interfaz gráfica, ni tal cosa, ni tal otra. Por lo tanto es necesario tomar decisiones de cuáles son las plataformas que se van usar para cada una de esas necesidades.

La implementación del diseño está determinada en tres componentes: El modelo de procesamiento de lenguaje natural, la base de datos vectorial y el proveedor de servicios de computación en la nube.

Este proyecto se pensó como el uso de los modelos de Open AI desde un principio, por tanto estos fueron seleccionados de manera arbitraria por su notoriedad, con el fin de llamar la atención de los posibles usuarios.

2.4.1 Selección del modelo de Open AI

Para el diseño del bot se requiere dos modelos de open AI uno de embeddings, encargado de generar vectores a partir de la documentación a utilizar y otro para generar las respuestas que se le entregará al usuario.

El modelo de embeddings a utilizar será “text-embedding-ada-002” puesto que es el último modelo de embeddings de OpenAI y es recomendado debido a que es más poderoso y más económico que modelos anteriores.

El modelo de generación de respuestas será “gpt-4-1106-preview” pues hace parte de la familia GPT-4 turbo que es una versión actualizada de GPT-4 a menor precio.No se selecciona GPT-3.5 debido a que tiende a confundirse y alucinar cuando en las preguntas del usuario se utilizan caracteres especiales.

2.4.2 Selección de base de datos vectorial

Existen varias bases de datos que se pueden utilizar para este proyecto entre estas están Pinecone, MongoDB y ChromaDB.

Todas las bases de datos anteriormente mencionadas están especializadas en el manejo de vectores, no solo para manejo de incrustaciones para modelos de procesamiento de lenguaje natural, sino también otras aplicaciones de inteligencia artificial, IoT y análisis de datos.

Entre estas se escogió Pinecone debido a que presenta cierta facilidad en su uso y su interfaz de manejo es más sencilla que en MongoDB a pesar de que MongoDB es ligeramente más económico. ChromaDB es una opción Open-Source sin embargo puede presentar problemas de rendimiento en aplicaciones en tiempo real.

2.4.3 Selección de Proveedor de servicios de computación en la nube.

En la elección de la plataforma en la nube para alojar nuestro bot de charla, nos enfrentamos a una decisión estratégica crucial. La selección no solo determinará la infraestructura que respalda la aplicación, sino que también influirá en su rendimiento, escalabilidad y costos asociados. En este proceso, evaluaremos cuidadosamente las opciones disponibles, sopesando las fortalezas y limitaciones de cada plataforma para asegurar una implementación óptima.

La elección de la nube no solo es un paso técnico; es una decisión estratégica que impactará directamente en la eficiencia y adaptabilidad de nuestro bot. En este análisis, exploramos los factores clave que guían nuestra elección, desde la integración fluida con herramientas existentes hasta la capacidad de escalar según las demandas del usuario. La plataforma en la nube no solo será el hogar del bot de charla, sino también un cimiento crucial para su éxito continuo.

En la elección del proveedor de nube para alojar nuestro bot de charla basado en Flask, la evaluación de opciones como AWS, Azure, GCC y PythonAnywhere es esencial para garantizar una implementación eficiente y ajustada a nuestras necesidades específicas.

Ventajas de PythonAnywhere:

- **Especialización en Python y Flask:** PythonAnywhere se distingue por su enfoque específico en el ecosistema Python y su sólido soporte para el framework Flask. Esta especialización facilita la configuración y el despliegue de nuestra aplicación, asegurando una integración suave.
- **Entorno Integrado con Anaconda:** La conexión directa con Anaconda simplifica el manejo de dependencias y la compatibilidad con bibliotecas específicas, facilitando la gestión del entorno de desarrollo de nuestra aplicación.
- **Modelo de Precios Accesible:** Para proyectos con requisitos de escala moderada, el modelo de precios de PythonAnywhere puede resultar más accesible en comparación con proveedores de nube más grandes, contribuyendo a una gestión eficiente de costos.

Desventajas de PythonAnywhere:

- **Limitaciones en Escalabilidad:** Aunque es adecuado para proyectos de escala moderada, PythonAnywhere puede presentar limitaciones en términos de escalabilidad comparado con proveedores de nube más grandes como AWS y Azure.

Ventajas de AWS, Azure y GCC:

- **Escalabilidad Ilimitada:** Plataformas como AWS, Azure y GCC ofrecen una escalabilidad prácticamente ilimitada, lo que las convierte en opciones robustas para proyectos que requieren un crecimiento considerable en el futuro.
- **Diversidad de Servicios y Recursos:** Estos proveedores ofrecen una amplia gama de servicios y recursos, desde servicios de cómputo hasta herramientas avanzadas de inteligencia artificial. Esto proporciona una flexibilidad excepcional para adaptarse a diversas necesidades y requisitos.

Desventajas de AWS, Azure y GCC:

- **Curva de Aprendizaje Pronunciada:** La complejidad y la diversidad de servicios de estas plataformas pueden resultar abrumadoras para usuarios menos experimentados, aumentando la curva de aprendizaje y el tiempo necesario para la configuración inicial.
- **Costos Potencialmente Más Altos:** Aunque ofrecen una amplia gama de servicios, AWS, Azure y GCC pueden tener costos asociados más altos, especialmente para proyectos de menor escala. La gestión cuidadosa de recursos es crucial para evitar gastos innecesarios.

En conclusión, la elección entre PythonAnywhere y proveedores de nube más grandes implica un equilibrio cuidadoso entre la especialización, la escalabilidad y los costos. Nuestra decisión de optar por PythonAnywhere se basa en la alineación específica con las necesidades tecnológicas de nuestro proyecto, buscando una integración fluida y un despliegue eficiente. Sin embargo con el código fuente del servidor web es posible migrar el proyecto a otro proveedor en dado caso de que el alcance de la implementación sobrepase el servicio.

2.5 Procuracion de la Documentación

En el corazón de la eficacia del bot de charla de la E3T se encuentra un desafío clave: la necesidad de comprender y aplicar un conjunto definido de reglas para la redacción de documentos. Este proceso es crucial para la generación óptima de embeddings por parte del modelo GPT.La calidad de la información con la que se alimenta el modelo de embeddings tiene un impacto significativo en su rendimiento. Es esencial que la E3T identifique y establezca pautas claras y cohesivas para la redacción de documentos, garantizando que la información procesada por el modelo sea consistente y contextualmente relevante.

La elaboración de documentación destinada a la generación de embeddings por parte del modelo GPT no solo implica una tarea de redacción convencional, sino que se convierte en un proceso estratégico para asegurar la coherencia y relevancia contextual. Ante este desafío, la E3T puede implementar diversas pautas y estrategias clave para maximizar la efectividad de este proceso.

Claridad en la Estructura:

- Definir una estructura clara y lógica para los documentos, facilitando la comprensión tanto para lectores humanos como para el modelo GPT.
- Utilizar títulos, subtítulos y secciones para organizar la información de manera jerárquica, mejorando la accesibilidad y la interpretación por parte del modelo.

Uso de Lenguaje Coherente:

- Establecer directrices sobre el uso consistente de terminología y vocabulario, evitando ambigüedades y asegurando una interpretación uniforme por parte del modelo GPT.

Contextualización de la Información:

- Proporcionar contextos claros y relevantes para cada sección del documento, permitiendo al modelo GPT comprender la relación entre diferentes fragmentos de información.

Evitar Ambigüedades:

- Eliminar ambigüedades y ambivalencias en la redacción, garantizando una interpretación precisa por parte del modelo y evitando posibles confusiones.
- Considerar el manejo adecuado de sinónimos y polisemia. Un ejemplo de polisemia sería la palabra banco con dos significados; una entidad financiera y un tipo de silla.

Validación Humana:

- Implementar procesos de revisión humana para verificar la calidad y coherencia de la documentación. La validación manual contribuye a detectar matices que podrían escapar a la interpretación del modelo.

Aclaración de Elementos Visuales:

- Debido a la naturaleza basada en texto de los embeddings de OpenAI, es crucial proporcionar explicaciones detalladas de cualquier imagen o tabla incluida en la documentación. Esto garantiza una comprensión completa por parte del modelo, evitando omisiones de información.

2.6 Generación de Base de datos vectorial

La base de datos vectorial sería para el bot lo que en términos humanos sería la memoria, las respuestas de este estarán determinadas por la información que se le suministre, si esta información es incorrecta o incompleta, también lo será la respuesta del bot.

En esta sección, se explicará el proceso de creación de la base de datos y qué decisiones se tomaron para facilitar la experiencia del administrador.

Este parte del proyecto se realiza por medio de Google Colab debido a que la documentación a utilizar se encuentra guardada en Google Drive, si se quiere acceder a dichos documentos a través de código se requerirá una verificación de acceso a drive de manera periódica, utilizando colab se limita esa restricción en caso de que el dueño de la carpeta sea el que utilice el notebook de Colab.

El primer paso consiste en determinar el directorio donde están guardados los documentos y guardar esa dirección en una variable de tipo string, con este dirección se genera una variable tipo document, que consiste en una cadena de texto, que corresponde al texto del documento y a unos metadatos asociados al texto, que sera la direccion de donde se extrajo el texto y el nombre del documento como tal.Esto se hace a partir de un DocumentLoader de LangChain; LangChain es un framework para desarrollar aplicaciones impulsadas por modelos de lenguaje. Permite crear aplicaciones que son conscientes del contexto, conectan el modelo de lenguaje a fuentes de contexto (instrucciones de prompts, ejemplos de pocos disparos, contenido para fundamentar su respuesta, etc.) y razonan, confían en un modelo de lenguaje para razonar (sobre cómo responder en función del contexto proporcionado, qué acciones tomar, etc.)(Langchain)

Con este proceso finalizado debe existir una variable de tipo directorio que contenga el texto de los documentos y sus metadatos, este debe tener un tamaño igual al número de documentos dentro del directorio inicial, esta variable debe ser seccionada en cúmulos de información más pequeños ;hay varias razones por las que puede ser útil seccionar documentos:

- **Para que los embeddings puedan reflejar mejor el significado.** Si los documentos son demasiado largos, los embeddings pueden perder precisión.

- **Para mantener el contexto de cada trozo.** Si los documentos son muy largos, puede ser difícil para el modelo entender y retener todo el contexto.

- **Para ajustarse a la ventana de contexto del modelo.** Muchos modelos de lenguaje tienen una ventana de contexto limitada que no puede procesar documentos muy largos.

Este último ítem es importante debido a que las API de Open AI tienen un límite de procesamiento de 4096 tokens cuyo uso se divide en la entrada y la salida se hace necesario dividir documentos que superen este umbral.Los tokens se pueden entender como piezas de palabras, antes de que la API procese las peticiones la entrada se divide en tokens.En general un token corresponde a 4 caracteres de texto en inglés. (What Are Tokens and How to Count Them?, 2023)

Por ejemplo, el párrafo anterior corresponde a 382 caracteres que en procesamiento significaron 117 tokens, lo que en términos de la API GPT 3.5 costaría en procesamiento 0.234 USD. (Tokenizer)

Con los documentos ya seccionados se procede a generar los embeddings, se inicia la comunicación con el servicio de la base de datos a utilizar. Se crea el índice donde se guardaran los embeddings.

Se envían a la base de datos los documentos seccionados, la variable de embeddings creada y el nombre del índice; este se encarga de hacer la petición de generar los vectores a partir del texto;

y luego organiza los vectores con su respectivo texto y metadatos.Para realizar búsquedas eficientes, se crea un índice de similitud que organiza los embeddings en función de su proximidad en el espacio vectorial.

Con esto se ha creado la base de datos que servirá como base del bot de charla.

2.7 Interpretación de Preguntas y Generación de Respuestas

Este capítulo explora cómo la elección de la base de datos vectorial impacta la interpretación de preguntas a través de consultas de similitud. También aborda la configuración del modelo de lenguaje mediante prompts y ajustes de temperatura para responder preguntas de manera coherente. Revelaremos cómo estas decisiones fundamentales dan forma a la capacidad del bot para entender y generar respuestas en el contexto de la conversación.

Con la base de datos generada, se habilita la capacidad de responder a las interrogantes formuladas por el usuario. No obstante, para llevar a cabo este proceso, resulta esencial enviar el contexto adecuado a OpenAI, facilitando así el procesamiento y la generación de una respuesta coherente.

La transformación de la pregunta del usuario en un vector comparativo con los embeddings se vuelve imperativa para enviar el contexto pertinente. Empleando el mismo modelo de embeddings utilizado en la creación de la base de datos, convertimos la pregunta del usuario en un vector. Luego, a través de las APIs de Pinecone y Langchain, comparamos este vector de la pregunta con los vectores almacenados en la base de datos. Pinecone agiliza este procedimiento al integrar un sistema interno de consulta de similitud, el cual proporciona a Langchain el texto asociado al vector de mayor similitud.

Si no se utiliza una base de datos vectorial como Pinecone, el proceso de consulta de similitud podría realizarse de manera local, directamente en el entorno del sistema, utilizando bibliotecas de procesamiento de texto y algoritmos de similitud.

La implementación local de algoritmos de similitud puede ser eficiente para bases de datos pequeñas, pero puede volverse menos eficiente a medida que la base de datos crece. Los servicios especializados, como Pinecone, están optimizados para realizar búsquedas de similitud de manera rápida y eficiente, incluso en grandes conjuntos de datos, prescindir de servicios como Pinecone y realizar la consulta de similitud de manera local podría aumentar el grado de dificultad en el manejo de la base de datos. Aquí hay algunas consideraciones:

Eficiencia en Búsquedas:

- La implementación local de algoritmos de similitud puede ser eficiente para bases de datos pequeñas, pero puede volverse menos eficiente a medida que la base de datos crece. Los servicios especializados, como Pinecone, están optimizados para realizar búsquedas de similitud de manera rápida y eficiente, incluso en grandes conjuntos de datos.

Complejidad del Código:

- La implementación local de algoritmos de similitud y la gestión de la base de datos añaden complejidad al código. Se deben manejar aspectos como la indexación de vectores, la optimización de búsquedas y la gestión de recursos. Los servicios como Pinecone proporcionan una interfaz sencilla para realizar estas operaciones.

Escalabilidad:

- A medida que el proyecto escala y la base de datos crece, la escalabilidad puede convertirse en un desafío al realizar consultas de similitud de manera local. Los servicios externos, como Pinecone, están diseñados para escalar fácilmente y gestionar grandes volúmenes de datos.

Mantenimiento:

- El mantenimiento de algoritmos de similitud locales y la gestión de la base de datos pueden requerir un esfuerzo adicional en términos de actualizaciones, correcciones de errores y mejoras de rendimiento. Los servicios externos suelen encargarse de estas tareas, permitiendo que el equipo se enfoque en aspectos más específicos del proyecto.

Mientras que implementar la consulta de similitud localmente puede ser factible, especialmente para proyectos más pequeños, utilizar servicios especializados como Pinecone facilita la gestión y mejora la eficiencia, escalabilidad y mantenimiento del sistema en el contexto de bases de datos más grandes y proyectos más complejos.

Una vez obtenido este contexto, procedemos a solicitar a OpenAI la generación de la respuesta. Con Langchain, también tenemos la capacidad de agregar un contexto general para orientar al bot sobre la tarea que está desempeñando. En nuestro caso, este contexto específico indica al bot que está operando como un chatbot basado en GPT, encargado de responder preguntas relacionadas con el sistema de acreditación ABET.

En el transcurso de esta sección, hemos navegado por las fases del desarrollo de nuestra solución, desde la identificación primordial de las necesidades de los usuarios hasta la implementación de un sistema integral que aborda con eficacia cada uno de estos requerimientos. Nos hemos sumergido en la selección estratégica de plataformas, examinando con minuciosidad aquellas que mejor se alinean con los objetivos del proyecto y permiten una implementación fluida y eficaz. Este proceso ha actuado como el cimiento sobre el cual construir la infraestructura necesaria para llevar a cabo nuestra visión.

A medida que avanzamos, hemos explorado las directrices fundamentales para la redacción y estructuración de documentos, reconociendo su papel esencial como la memoria del bot. Las pautas específicas delineadas aseguran la calidad y coherencia de la información almacenada, respaldando así la toma de decisiones y la generación de respuestas contextualmente precisas.

Finalmente, nos sumergimos en los intrincados procedimientos para el procesamiento de preguntas de los usuarios. Desde la recepción inicial de la consulta hasta la generación de respuestas, cada paso ha sido meticulosamente explorado. Hemos destacado cómo nuestro sistema interpreta y maneja la información para proporcionar respuestas coherentes y pertinentes.Al llegar a este punto, celebramos los logros alcanzados y reflexionamos sobre el camino recorrido. Sin embargo, este cierre no marca el final, sino el inicio de nuevas perspectivas y desafíos. La evolución tecnológica continúa, y con ella, se presentan oportunidades para mejorar y perfeccionar nuestra solución.

3. Validación de la solución

Mientras la documentación ABET y los modelos GPT evolucionan constantemente, nuestra estrategia se centra en la creación de un marco robusto y ágil que no solo responda al presente, sino que también esté listo para abrazar los cambios futuros de manera eficaz.

Adentrándonos en la validación de resultados, explicaremos los procedimientos meticulosos que hemos implementado para asegurar la coherencia y precisión de las respuestas del bot. En un entorno donde las pautas de ABET y las expectativas de los usuarios pueden cambiar, nuestra estrategia de validación continua se erige como un pilar fundamental. Detallaremos casos de prueba específicos creados para abordar cambios en las pautas de ABET y evaluaciones regulares del rendimiento del modelo, asegurando que nuestro bot se mantenga a la vanguardia en cada interacción.

Con el objetivo de validar la eficacia del bot de charla en responder preguntas relacionadas con la documentación en constante evolución, se ideó un proceso sistemático mediante un script en Colab. Este script se integra con una Hoja de cálculo en Google Drive, donde se introducen una serie de preguntas específicas en una columna designada. Este enfoque permite evaluar la capacidad del bot para proporcionar respuestas precisas y contextualmente relevantes.

El proceso se inicia con la inserción de preguntas relacionadas con la documentación en la hoja de cálculo generada. Posteriormente, se ejecuta una función especializada que interactúa con el bot para obtener respuestas a cada una de las preguntas previamente registradas en la hoja de cálculo. Una vez completada esta fase, el script actualiza automáticamente la hoja de cálculo con las respuestas correspondientes, creando así un registro detallado de las interacciones y resultados.

Esta metodología ofrece la ventaja de generar múltiples respuestas de manera eficiente, agilizando el proceso de validación. Es importante destacar que, aunque el bot de charla puede responder a preguntas de manera automatizada, la revisión y validación de las respuestas siguen siendo procesos manuales. Los chatbots, por su naturaleza, carecen de la capacidad para validar su propia existencia de manera autónoma o reflexiva. Su funcionamiento se rige por las instrucciones y datos con los que han sido programados, y su capacidad se limita al procesamiento de información y la generación de respuestas basadas en ese conocimiento.

La validación de la existencia o la autorreflexión son conceptos más complejos que involucran aspectos de conciencia y autoconciencia, capacidades que actualmente están más allá del alcance de los chatbots y sistemas de inteligencia artificial actuales. A través de este proceso de validación, se busca evidenciar la capacidad del bot de charla para ofrecer respuestas coherentes y precisas, revelando así la limitación intrínseca de sus respuestas a la información contenida en la documentación proporcionada durante la generación de la base de datos vectorial.

Tabla 1

Extracto de la Hoja de Cálculo de Validación

Pregunta	Respuesta del Bot
Que es SO1?	SO1 es un Student Outcome (Resultado de Aprendizaje) definido por ABET para programas de ingeniería en la E3T, que se relaciona con competencias específicas que los estudiantes deben desarrollar. El Manual del SO1 proporciona instrucciones detalladas para que los profesores creen instrumentos de medición que cumplan con los requisitos de ABET, asegurando que las mediciones sean confiables y coherentes, y que los procesos de formación estén comprometidos con el mejoramiento continuo.

Que es SO2?	SO2 es un Student Outcome (Resultado de Aprendizaje) definido por ABET que se relaciona con la capacidad de los estudiantes para resolver problemas complejos de diseño en ingeniería. Estos problemas son abiertos y su solución está representada en un producto que satisface una necesidad del mundo real. A diferencia de los problemas que se resuelven en poco tiempo, los problemas de diseño suelen requerir un plazo más largo (meses) para su resolución y tienen asociado un método específico para abordarlos.
Dime los pasos que se usan en el método E3T para solucionar problemas complejos	Los pasos utilizados en el método E3T para solucionar problemas complejos son: 1. Definición del Problema: Comprender claramente cuál es el problema que se está enfrentando, identificar objetivos, preguntas de investigación, variables, restricciones y criterios. 2. Diseño Experimental: Planificar cómo se llevará a cabo el experimento, incluyendo la formulación de hipótesis, definición de cómo variarán las variables independientes, cómo se recopilarán los datos y qué controles se implementarán para asegurar la validez del experimento. 3. Recopilación de Datos: Ejecutar el experimento siguiendo el diseño planificado y tomar medidas precisas y confiables de las variables dependientes en diferentes condiciones. 4. Planeación: Se aplica el ID1 de la rúbrica SO1, que implica definir el problema de ingeniería complejo e identificar las características propias de este (variables, criterios y restricciones) necesarias para plantear una solución apropiada y argumentar el planteamiento.
Qué debo tener en cuenta para el paso 1 del método E3T?	Para el Paso 1 del método E3T, debes tener en cuenta la definición clara y precisa del problema. Esto incluye identificar los objetivos, formular preguntas de investigación relevantes, determinar las variables y restricciones, y establecer los criterios de éxito. Este paso es fundamental para guiar todo el proceso de resolución de problemas y debe alinearse con los estándares de acreditación de ABET.

Nota. Esta tabla muestra algunas respuestas generadas por el bot en el proceso de validación, todas las respuestas de este proceso se encuentran anexadas. Todas las preguntas fueron escritas manualmente y las respuestas se obtuvieron al aplicar un script de Google Colab sobre una hoja de cálculo de Google Drive.

La evaluación de las respuestas generadas por el bot mediante el script revela un desempeño sólido y coherente en la respuesta a preguntas específicas relacionadas con la documentación de la E3T. La tabla proporcionada destaca ejemplos representativos de respuestas, subrayando la capacidad del bot para contextualizar y ofrecer información pertinente en función de las consultas planteadas.

Este proceso de validación no solo confirma la capacidad del bot en el contexto presente, sino que también establece un protocolo para futuras actualizaciones y mejoras. La retroalimentación derivada de la revisión constante permite ajustar y mejorar el rendimiento del sistema, asegurando su adaptabilidad y relevancia en el cambiante escenario de la acreditación educativa y las tecnologías asociadas.

Existen diversas alternativas a los modelos de la API de OpenAI para la implementación de chatbots, cada una con sus propias características y modelos de precios. Por ejemplo, Kommunicate ofrece planes como Lite, con un costo de $100 al mes, incluyendo 2 compañeros de equipo y el rastreo de 500 usuarios mensuales (MTU); y Avanzado, por $200 al mes, con 5 compañeros de equipo y 5000 MTU. Por otro lado, Intercom Fin mide el uso en Resoluciones, con un precio actual de $0.99 por resolución para aquellos suscritos a un plan activo de Intercom.

Otras opciones incluyen Chatbot API, que presenta planes como: Starter por $52 al mes (suscripción anual), Team por $142 al mes (suscripción anual), y Business por $424 al mes (suscripción anual). Además, Wit.AI se destaca por ser de uso gratuito, incluso con fines comerciales, proporcionando una opción accesible para aquellos que buscan soluciones de procesamiento del lenguaje natural sin costos iniciales.

En resumen existen muchas alternativas en el mercado actualmente, normalmente estas alternativas no solo cuentan con su modelo de NLP, si no que a su vez pueden tener una interfaz definida y personalizable, por lo tanto no hay necesidad de un desarrollo desde 0 si se quiere implementar un chatbot en un proyecto.

La utilización de las API de GPT, presenta notables ventajas al eliminar la necesidad de alquilar software o hardware para el entrenamiento del modelo. A diferencia de los enfoques tradicionales que requieren infraestructuras costosas y considerable capacidad computacional para el proceso de entrenamiento, las API de GPT ofrecen una solución eficiente y accesible. Al externalizar el trabajo intensivo en recursos a los servidores de OpenAI, los usuarios pueden aprovechar la potencia de modelos avanzados sin la carga financiera y logística asociada con la gestión de un entorno de entrenamiento local. Esta característica democratiza el acceso a la vanguardia en procesamiento del lenguaje natural, permitiendo a una gama más amplia de desarrolladores y empresas beneficiarse de las capacidades de modelos de alta calidad sin comprometer recursos significativos en la fase de entrenamiento.

4. Conclusiones

En el transcurso de este proyecto, se lograron avances significativos hacia la consecución de los objetivos planteados. El entrenamiento del modelo a partir de la información procesada se llevó a cabo con éxito, destacando la importancia de un proceso de verificación meticuloso que garantizara la calidad y confiabilidad del bot. La atención particular se centró en asegurar respuestas precisas y actualizadas, evitando la presentación de información contradictoria o falsa.

La implementación de un mecanismo de actualización y mantenimiento de la información constituyó un componente clave. La capacidad para reentrenar el modelo a partir de documentación actualizada y recibir indicaciones de respuestas insatisfactorias asegura la adaptabilidad continua del bot ante cambios en las directrices ABET y otras actualizaciones relevantes.

La evaluación de costos y beneficios en el uso de la tecnología GPT de OpenAI fue un proceso integral. Comparando sus capacidades con alternativas disponibles en el mercado, se buscó determinar si la robustez y eficacia de GPT justificaban su costo. Este análisis proporcionó una visión crítica sobre la elección de la tecnología subyacente en el proyecto, fundamentando las decisiones en función de la relación costo-rendimiento.El uso de una API disminuye costos en el inicio de un proyecto ya que se terceriza el procesamiento de la información y elimina la necesidad de adquirir o alquilar hardware con la potencia necesaria para entrenar un modelo de inteligencia artificial, en este caso un modelo de procesamiento de lenguaje natural

Finalmente, la implementación exitosa del bot de charla como un servicio web accesible a través de la página de la E3T marca la culminación de los objetivos. La accesibilidad y disponibilidad del sistema ofrecen una vía eficaz para que los usuarios interactúen con el bot y obtengan respuestas contextualmente precisas en el contexto de la acreditación educativa.

5. Recomendaciones

Con miras a mejoras continuas, se sugiere la implementación de un sistema de identificación con roles específicos, diferenciando claramente entre administradores y usuarios. Esta medida contribuirá a una gestión más eficiente de funciones y a la seguridad del sistema.

Para optimizar costos, se sugiere el uso de servicios de API, como OpenAI, que permite externalizar el procesamiento de información y evita la necesidad de adquirir hardware costoso.También se aconseja explorar modelos open source, como los disponibles en HuggingFace, esto permite evaluar alternativas y adaptarse a las últimas innovaciones en procesamiento de lenguaje natural.

La documentación debe ser considerada un activo crítico. En el futuro, se aconseja dedicar esfuerzos continuos para mantenerla actualizada y precisa. Esta práctica no solo asegurará respuestas confiables del bot, sino que también facilitará procesos de actualización y mejora.

Explorar herramientas avanzadas como LangSmith para el desarrollo de aplicaciones a gran escala se presenta como una recomendación futura. Este enfoque permitirá aprovechar características específicas y avanzadas para optimizar el rendimiento del sistema y mantenerse a la vanguardia de las capacidades tecnológicas.

Referencias Bibliográficas

ACOFI. (2023). Modelo de Acreditación ABET. https://www.acofi.edu.co/modelo-de-acreditacion-abet/

Amazon. (n.d.). ¿Qué es Python?. What is Python? https://aws.amazon.com/es/what-is/python/

Amazon. (n.d.). ¿Qué es GPT?. What is GPT? https://aws.amazon.com/es/what-is/gpt/

Amazon. (n.d.). ¿Qué es una interfaz de programación de aplicaciones (API)?. What is api?. https://aws.amazon.com/es/what-is/api/

Chase, H. (n.d.). API References — LangChain 0.0.188. LangChain. Retrieved June 1, 2023, from https://python.langchain.com/en/latest/reference.html

Dagster. Retrieved June 1, 2023, from https://dagster.io/blog/chatgpt-langchain

Embeddings - OpenAI API. (2023). Platform OpenAI. Retrieved June 1, 2023, from https://platform.openai.com/docs/guides/embeddings

Firat, M. (2023, January). How Chat GPT Can Transform Autodidactic Experiences and Open Education?. Research Gate. https://www.researchgate.net/publication/367613715_How_Chat_GPT_Can_Transform_Autodidactic_Experiences_and_Open_Education

Google. (n.d.). Colaboratory. Google colab. https://research.google.com/colaboratory/intl/es/faq.html

Hunt, P. (2023, January 9). Build a GitHub Support Bot with GPT3, LangChain, and Python.

IBM. (2022, June 7). Formato JSON (JavaScript object notation). json. https://www.ibm.com/docs/es/baw/20.x?topic=formats-javascript-object-notation-json-format

Langchain. (n.d.). https://python.langchain.com/docs

Liu, J. (n.d.). Welcome to llamaindex 🦙 !. LlamaIndex 🦙 0.6.8. https://gpt-index.readthedocs.io/en/latest/

Mora, J. N., & Leal, D. D. (2023). Diseño e implementación de un Bot para atender consultas sobre Capstone design y trabajos de grado en los programas de pregrado de la E3T. E3T.

Mozilla. (2023a, July 24). HTML: Lenguaje de etiquetas de hipertexto: MDN. MDN Web Docs. https://developer.mozilla.org/es/docs/Web/HTML

Mozilla. (2023b, July 24). JavaScript: MDN. MDN Web Docs. https://developer.mozilla.org/es/docs/Web/JavaScript

MINTIC Colombia. (2022, March 9). Colombia Adopta de Forma Temprana recomendaciones de ética en inteligencia artificial de la UNESCO para la Región - Colombia Adopta de Forma Temprana recomendaciones de ética en inteligencia artificial de la UNESCO para la región. MINTIC Colombia. https://mintic.gov.co/portal/inicio/Sala-de-prensa/Noticias/208109:Colombia-adopta-de-forma-temprana-recomendaciones-de-etica-en-Inteligencia-Artificial-de-la-Unesco-para-la-region

NLP (natural language processing): ¿qué es y para qué sirve? (2021, August 6). UNIR. Retrieved June 1, 2023, from https://www.unir.net/marketing-comunicacion/revista/nlp-procesamiento-lenguage-natural/

Nithuna, S., & Laseena, C. A. (2020, September 1). Review on Implementation Techniques of Chatbot. IEEEXplore. https://ieeexplore.ieee.org/abstract/document/9182168

Pricing. (2023). OpenAI. Retrieved June 1, 2023, from https://openai.com/pricing

Python In Office. (2023, April 14). LangChain - Train ChatGPT with Your Own Data.

PythonAnywere. (2015, May 13). 403 forbidden error. PythonAnywhere help. https://help.pythonanywhere.com/pages/403ForbiddenError/

Qué es un prompt en ia Y Para Qué sirve (+ ejemplos). Entel Comunidad Empresas. (n.d.). https://ce.entel.cl/articulos/que-es-un-prompt/

Sáiz-Manzanares, M. C., Marticorena-Sánchez, R., Martín-Antón, L. J., González Díez, I., & Almeida, L. (2023, January 1). Perceived satisfaction of university students with the use of chatbots as a tool for self-regulated learning. ScienceDirect. https://www.sciencedirect.com/science/article/pii/S2405844023000506

SAS. (n.d.). Qué Es el procesamiento de lenguaje natural - natural language processing?. Procesamiento del lenguaje natural Qué es y por qué es importante. https://www.sas.com/es_co/insights/analytics/what-is-natural-language-processing-nlp.html

Servicios web. (2022, 12 13). IBM. Retrieved June 2, 2023, from https://www.ibm.com/docs/es/was-nd/9.0.5?topic=services-web

Soriano, P. (2022, March 26). HTML, CSS y JavaScript. Lenguajes para el Desarrollo de Páginas Web. Geoinnova. https://geoinnova.org/blog-territorio/html-css-y-javascript-lenguajes-para-el-desarrollo-de-paginas-web/

Tokenizer. (n.d.). OpenAI API. Retrieved June 1, 2023, from https://platform.openai.com/tokenizer

Tripathi, R. (n.d.). What is similarity search?. Pinecone. https://www.pinecone.io/learn/what-is-similarity-search/

Unesco. (1970, January 1). Ética de la inteligencia artificial. UNESCO. https://www.unesco.org/es/artificial-intelligence/recommendation-ethics

What are tokens and how to count them? (2023, 05). OpenAI Help Center. Retrieved June 1, 2023, from https://help.openai.com/en/articles/4936856-what-are-tokens-and-how-to-count-them

YouTube. Retrieved May 31, 2023, from http://www.youtube.com/watch?v=tWCaANq7xKg

Apéndices

Apéndice A. Carpeta con el código Fuente del servicio Web

Apéndice B. Cuaderno de Google Colab con el sistema para crear y actualizar la base de datos vectorial

Apéndice C. Cuaderno de Google Colab con el sistema para validar las respuestas del bot

Apéndice D. Hoja de Cálculo de Google Drive con las respuestas generadas en el proceso de Validación.

Apéndice E. Manual de Administrador para el sistema de administración de la base de datos y del servicio web