TRABAJO DE GRADO

Opción Seminario-Diplomado.

ASISTENTE VIRTUAL CAPAZ DE RESPONDER PREGUNTAS CON BASE A DOCUMENTOS SUMINISTRADOS HACIENDO USO DE TÉCNICAS Y ALGORITMOS DE MACHINE LEARNING E INTELIGENCIA ARTIFICIAL

Corporación Universitaria Remington.

Nombre de la facultad: Ingeniería

Nombre del programa académico: Ingeniería de Sistemas

Estudiante:

Jorge Armando Narváez Naranjo
Victor Ovidio Castillo Rodriguez

Tutor: Danny López Segura

Opción de Trabajo de grado Seminario-Diplomado.

2025

Contenido

Índice de tablas

Índice de Ilustraciones

Resumen

En el presente proyecto se propone la creación de un diseño e implementación de un asistente virtual inteligente para el fondo de empleados Fondecom, con el objetivo de facilitar la información que contiene los documentos de la entidad. Este asistente virtual será capaz de responder las preguntas realizadas por los usuarios y/o empleados, utilizando como suministro la información dada en los documentos corporativos.

La solución está basada en técnicas de inteligencia artificial, especialmente en modelos de lenguaje natural y algoritmos de machine learning, lo cual permitirá ofrecer respuestas coherentes al usuario. Este asistente virtual busca disminuir los tiempos de búsqueda de información, y con esto mejorar la toma de decisiones en la entidad en los diferentes procesos.

Durante el desarrollo del proyecto se realizó un análisis del proceso para la recopilación de datos, pruebas de los diferentes modelos de lenguaje, y la evaluación de su desempeño de manera manual en escenarios reales de consulta. Finalmente, se realiza un análisis de los resultados obtenidos.

Palabras clave

Inteligencia artificial, machine learning, procesamiento, lenguaje natural, asistente virtual, sistema de recomendación, recuperación de información, gestión documental.

1. Marco conceptual y contextual

1.1. Contexto:

1.1.1 Sistemas de recomendación.

En este proyecto, si bien no se implementó un sistema de recomendación en el sentido clásico (como en plataformas de comercio electrónico o entretenimiento), se adoptaron herramientas

similares para personalizar las respuestas del asistente virtual a partir del análisis semántico de las consultas realizadas por los usuarios.

El asistente virtual implementa métodos de procesamiento de lenguaje natural (PLN) junto a una base vectorial de los documentos de modo que identifique fragmentos de texto que se adecuen mejor a la pregunta introducida. El funcionamiento de este asistente es similar al de un sistema de recomendaciones basado en contenido, que ya la respuesta se genera a partir de la comparación entre la pregunta del usuario y el contenido del asistente que ha sido indexado anteriormente.

Aunque en este proceso no se recopila información del historial de cada pregunta realizada por los usuarios (como lo haría un sistema de filtrado colaborativo, el cual predice las respuestas basándose en el historial), el motor de búsqueda semántica implementado es basado en embeddings y FAISS el cual actúa como un sistema de recomendación que selecciona y prioriza la información más relevante para cada pregunta.

Por lo tanto, se puede afirmar que el proyecto se basa en un enfoque de recomendación **basado en contenido semántico**, buscando facilitar el acceso a la información corporativa según las necesidades de los empleados y/o usuarios. Esto ayuda a mejorar la eficiencia en la gestión de información interna por parte de los usuarios en la entidad de Fondecom.

1.1.2 Algoritmos de Machine learning en sistemas de recomendación.

En el desarrollo del asistente virtual, se utilizaron algoritmos de machine learning guiados a mejorar la comprensión del lenguaje natural y la búsqueda semántica de información relevante dentro de los documentos corporativos.

Se emplearon modelos **preentrenados de embeddings**, como los derivados de BERT (por ejemplo, all-MiniLM-L6-v2), los cuales convierten tanto las consultas de los usuarios como

los fragmentos de documentos en vectores (Reimers & Gurevych, 2019). Esta representación vectorial permite comparar la semántica de las preguntas y respuestas, facilitando la búsqueda del contenido indexado. Este tipo de modelo pertenece a la familia de **redes neuronales profundas**.

Además, se implementó **FAISS** como motor de búsqueda vectorial, el cual permite realizar comparaciones eficientes entre vectores de gran tamaño, identificando de manera rápida cuál es el fragmento de documento más cercano a la pregunta realizada por el usuario (Johnson et al., 2019).

Teniendo en cuenta que no se realizó un entrenamiento desde cero, se utilizó el concepto de *transfer learning* al aplicar modelos ya entrenados.

En resumen, el proyecto implementó **algoritmos basados en embeddings y búsqueda semántica**, apalancados en redes neuronales tipo Transformer, lo que permitió construir un asistente virtual inteligente, capaz de comprender preguntas en lenguaje natural y ofrecer respuestas coherentes de acuerdo con la información corporativa.

1.2 Descripción de caso de estudio.

El caso de estudio para el presente proyecto de grado se centra en la empresa **Fondecom**, una entidad que presta servicios a sus asociados, gestionando información financiera, administrativa y documental de manera constante. Dada la cantidad de documentos que maneja; incluyendo reglamentos internos, resoluciones, actas, informes, manuales de procedimientos, y comunicaciones oficiales, se ha identificado una necesidad de mejorar el acceso y la consulta de dicha información por parte de los empleados y/o usuarios.

Los procesos de búsqueda y consulta de documentos en Fondecom se realizan actualmente de forma tradicional; a mano y alojados en sitios de Sharepoint en Office 365, lo cual se evidencia un mayor consumo de tiempo y recursos, estos inconvenientes recurrentes

generan retrasos en la toma de decisiones o en la atención de consultas internas e incluso a la hora de atender un asociado. Todo esto demuestra la necesidad de una solución, en este caso: tecnológica, una solución que permita a los trabajadores acceder de manera ágil, sencilla y efectiva al contenido relevante de la documentación Corporativa.

En este contexto, se propone el desarrollo de un **asistente virtual inteligente**, que mediante el uso de técnicas de procesamiento de lenguaje natural (NLP) y algoritmos de machine learning, sea capaz de comprender preguntas formuladas en lenguaje natural y responder con base a la información contenida en los documentos internos de la empresa. Este asistente actuará como un intermediario inteligente entre los empleados y la base documental, reduciendo el tiempo de búsqueda operativa y facilitando el acceso al conocimiento corporativo.

1.3 Pregunta problema:

¿Cómo desarrollar un asistente virtual, basado en técnicas de machine Learning e inteligencia artificial, que permita a los empleados de Fondecom consultar y obtener respuestas precisas a partir de la información contenida en los documentos internos de la entidad, optimizando así las búsquedas de información para los empleados y/o usuarios en los diferentes procesos?

1.4 Hipótesis:

La implementación de un asistente virtual basado en técnicas de inteligencia artificial y machine learning permitirá mejorar significativamente el acceso a la información contenida en los documentos internos de la empresa Fondecom, optimizando el tiempo de búsqueda y aumentando la eficiencia operativa del personal.

2. Objetivos

2.1 Objetivo general.

Se busca desarrollar un asistente virtual empleando técnicas de inteligencia artificial y machine learning, permitiendo a los empleados de Fondecom consultar información contenida en documentos corporativos mediante el uso de lenguaje natural, con el fin de optimizar los procesos de búsqueda y acceso a la información interna.

2.2 Objetivos específicos.

- Analizar el entorno corporativo de Fondecom y sus necesidades en cuanto al acceso rápido de la información en los documentos internos.
- Recolectar y procesar los documentos internos de la entidad, organizándolos adecuadamente para su uso en modelos de procesamiento de lenguaje natural.
- Diseñar e implementar un asistente virtual que utilice técnicas de machine learning y procesamiento de lenguaje natural para interpretar preguntas y suministrar respuestas lo más precisas posibles.
- Validar la efectividad del asistente virtual a través de pruebas con el área de TIC, identificando posibles mejoras.

3. Desarrollo e implementación del aprendizaje

3.1 Preparación y análisis de los datos

La preparación y análisis de los datos componen una de las fases más importantes en el desarrollo de este asistente virtual, ya que la calidad del desarrollo depende mucho la

información que se suministre, en esta etapa incluye la recolección, limpieza, organización y transformación de los documentos internos que servirán como base de conocimiento para el sistema.

3.1.1. Recolección de documentos

Se recopila distintos tipos de documentos proporcionados por Fondecom, entre los cuales se incluyen manuales, reglamentos, políticas internas, y comunicaciones oficiales, para efectos de la validación y pruebas se recopiló el documento estatuto de Fondecom 2024 el cual alberga mucha información de la empresa.

3.1.2. Procesamiento y limpieza de texto

Los documentos fueron sometidos a un proceso de limpieza y normalización, que incluyó las siguientes tareas:

- Eliminación de caracteres especiales.
- Conversión del texto a formato plano y estructurado.
- Eliminación de espacios, como se evidencia en la ilustración 1.

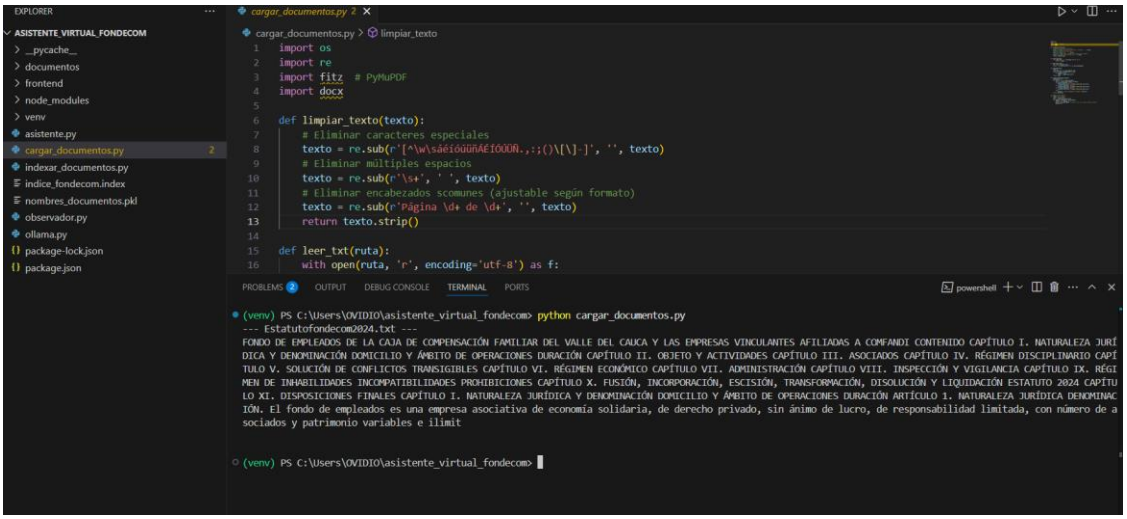

Ilustración 1. Fragmento de código cargar.documentos.py para el proceso de limpieza y cargue de los documentos.

3.1.3. Análisis exploratorio de los datos

Se realiza un análisis de los datos ingresados al asistente virtual en aras de aplicar la técnica de tokenización ya que al usar SentenceTransformer para generar los vectores de los textos, el modelo internamente realiza tokenización, este punto es esencial para el proceso de embeddings en NLP.

3.1.4. Representación vectorial

Se explora técnicas de representación vectorial basadas en transformers, utilizando embeddings preentrenados del modelo all-MiniLM-L6-v2, una versión optimizada derivada de BERT. Esta técnica permite capturar información de forma efectiva la semántica de los documentos y las preguntas formuladas por los usuarios, facilitando una búsqueda más precisa en la base documental, así como lo muestra la Ilustración 2 en la cual se evidencia la ejecución del archivo indexar_documentos.py y parte del código del archivo asistente.py donde captura la pregunta del usuario y la transforma semánticamente para realizar el proceso.

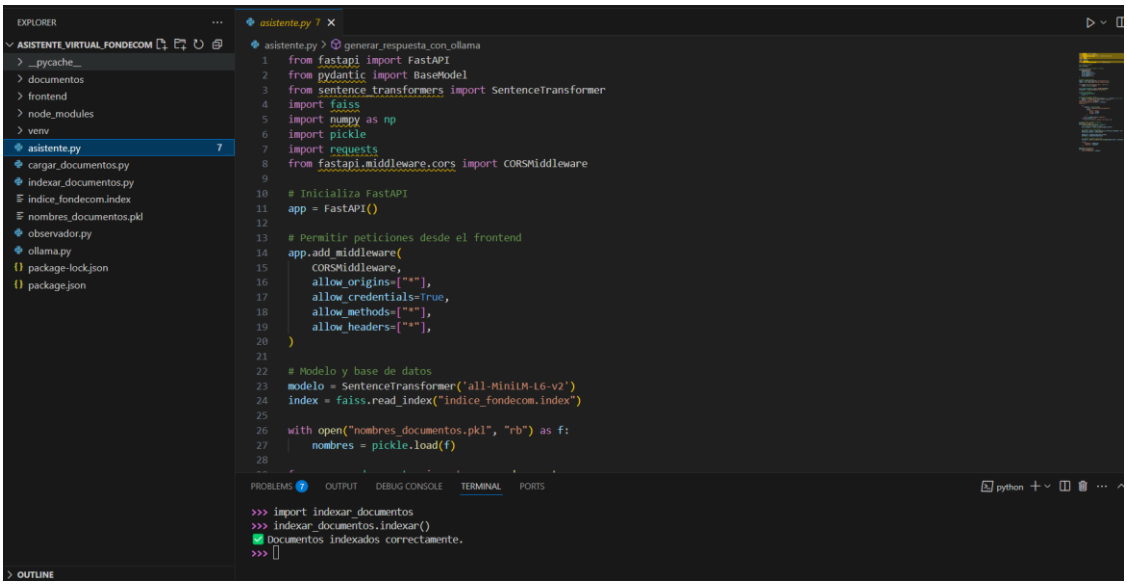

Ilustración 2. Ejecución del script indexar_documentos.py donde se procesan e indexan los archivos corporativos y fragmento del código asistente.py.

3.1.5. Selección del modelo de lenguaje

Para la comprensión del lenguaje natural y la representación semántica de los textos, se seleccionó un modelo basado en BERT (all-MiniLM-L6-v2), utilizado mediante la librería sentence-transformers. La búsqueda de documentos relevantes se realizó con FAISS, un motor de búsqueda vectorial eficiente, así como se muestra en la Ilustración 3. Finalmente, se integró tres modelos generativos (Mistral, llama3, Gemma) a través de Ollama para generar respuestas coherentes (Ollama, 2024). Esta arquitectura permite la recuperación semántica con generación de lenguaje natural de forma efectiva, así como se muestra en la Ilustración 4-6 donde se muestra la integración de los diferentes modelos de lengua de Ollama.

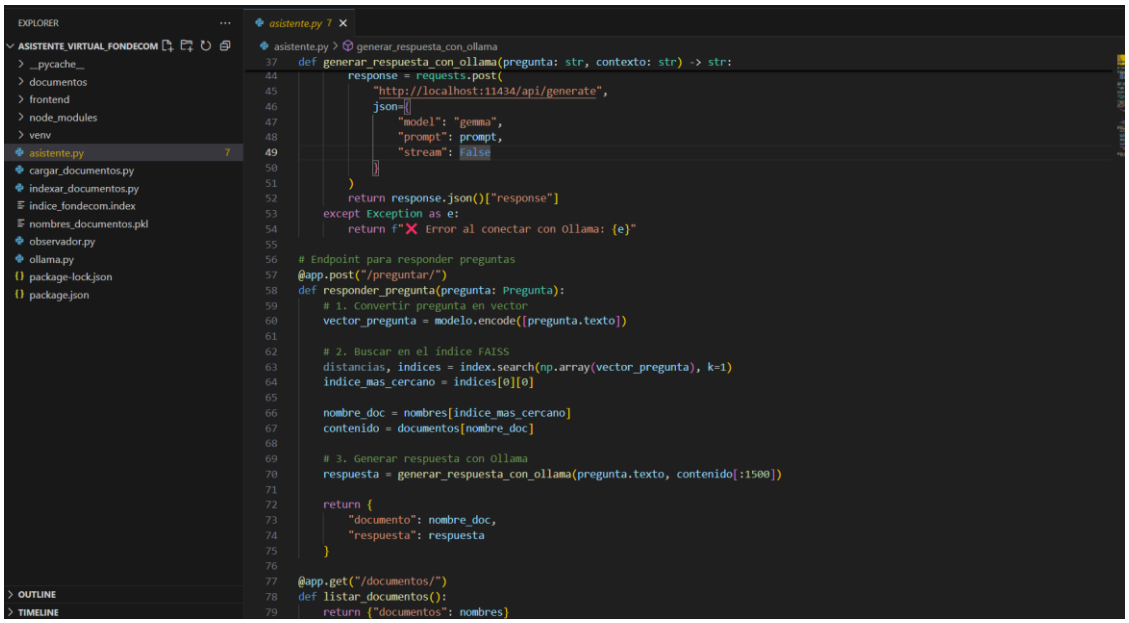

Ilustración 3. Fragmento del código Python donde se utiliza sentence-transformers y FAISS para buscar la información.

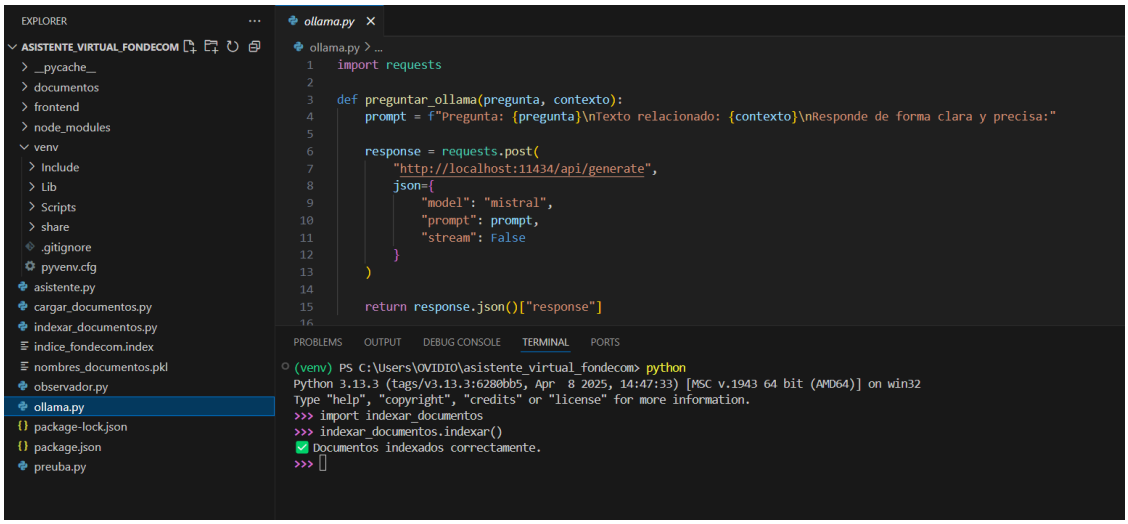

Ilustración 4. Integración de Ollama en el asistente virtual para generación de respuestas con modelo Mistral.

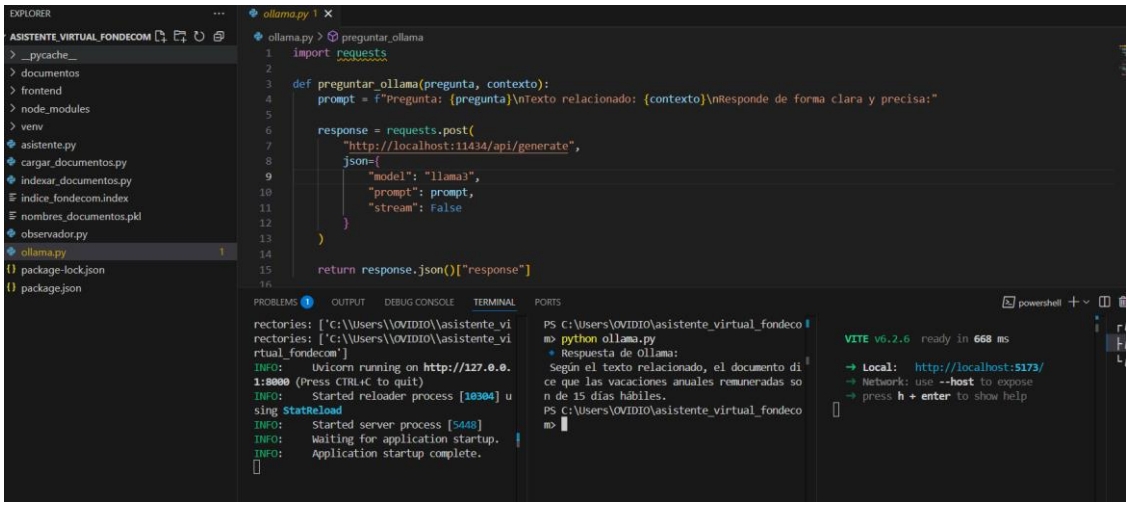

Ilustración 5. Integración de Ollama en el asistente virtual para generación de respuestas con modelo Llama3

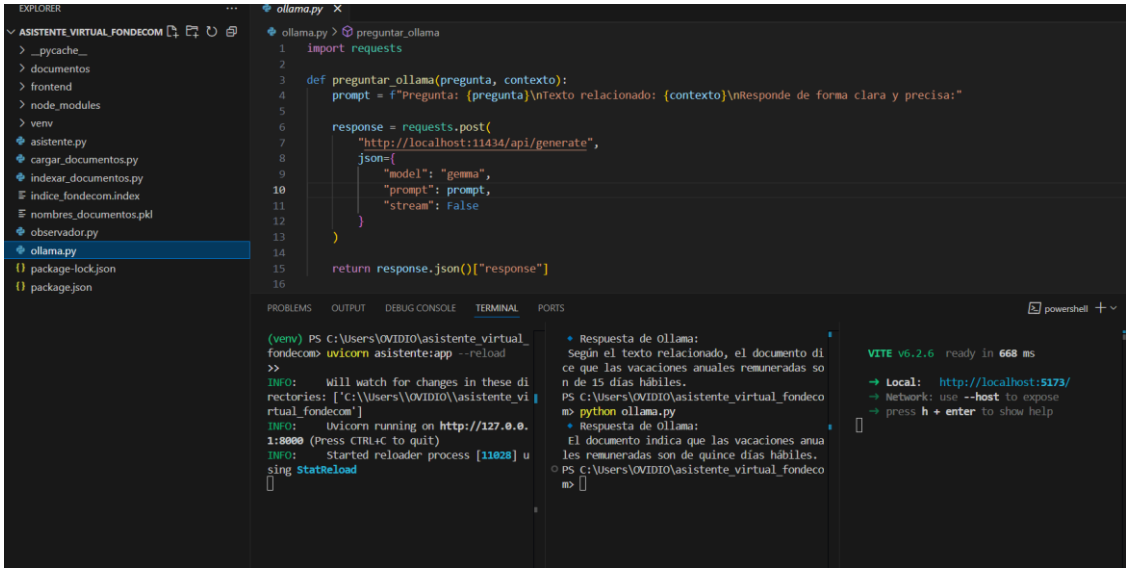

Ilustración 6 Integración de Ollama en el asistente virtual para generación de respuestas con modelo Gemma

3.1.6. Entrenamiento y pruebas

Para este proyecto no se realiza un entrenamiento desde cero, por el contrario, se utiliza modelos preentrenados con buenos resultados en tareas de comprensión semántica. Se integra el modelo all-MiniLM-L6-v2 a través de la librería sentence-transformers, empleando su capacidad para generar representaciones vectoriales contextuales sin requerir un entrenamiento exhaustivo. El sistema actualmente no incluye un clasificador adicional que valide la relevancia de las respuestas, ni se han aplicado métricas formales como precisión, recall o F1-score. Sin embargo,

las respuestas obtenidas fueron analizadas con base a la información indexada por parte de los documentos corporativos de la entidad.

3.1.7. Diseño de la interfaz del asistente

Se desarrolla una interfaz web intuitiva para que los empleados y/o usuarios interactúen con el asistente virtual. La interfaz permite lo siguiente:

- Escribir preguntas en lenguaje natural.
- Recibir respuestas textuales extraídas directamente de los documentos.
- Visualizar una respuesta de acuerdo con los datos de los documentos.
- La interfaz se construye usando tecnologías web modernas:

Ilustración 7. Interfaz del prototipo del asistente virtual.

- **Frontend**: (React)

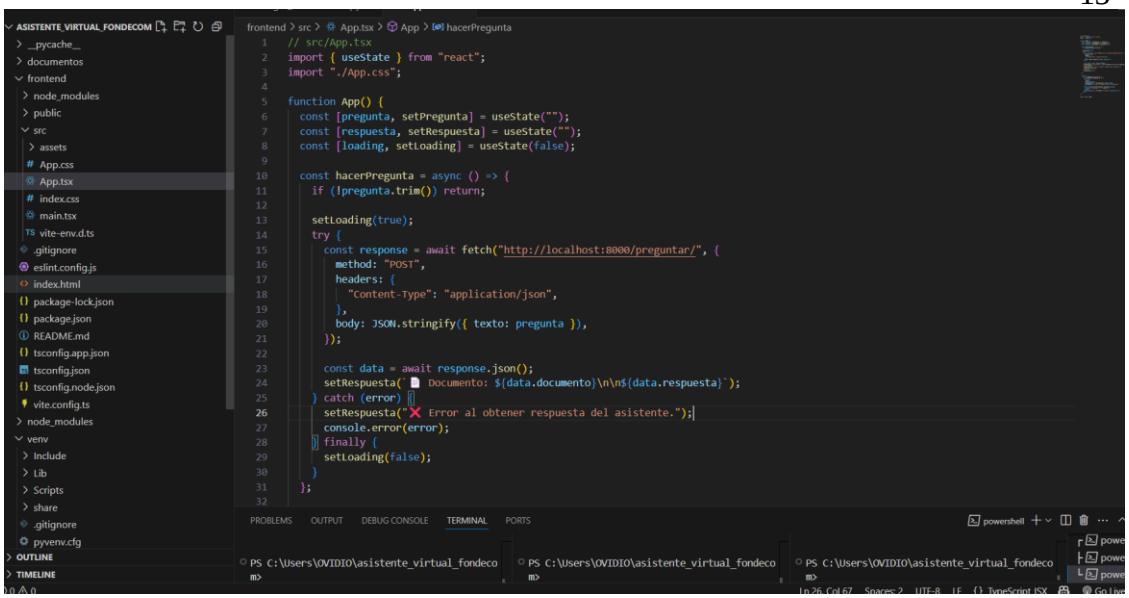

Ilustración 8. Desarrollo del Frontend en React.

- **Backend**: Se crea una API en FastAPI (Tiangolo, 2023) para permitir la conexión entre el modelo y la interfaz desarrollada en React (React, 2023).

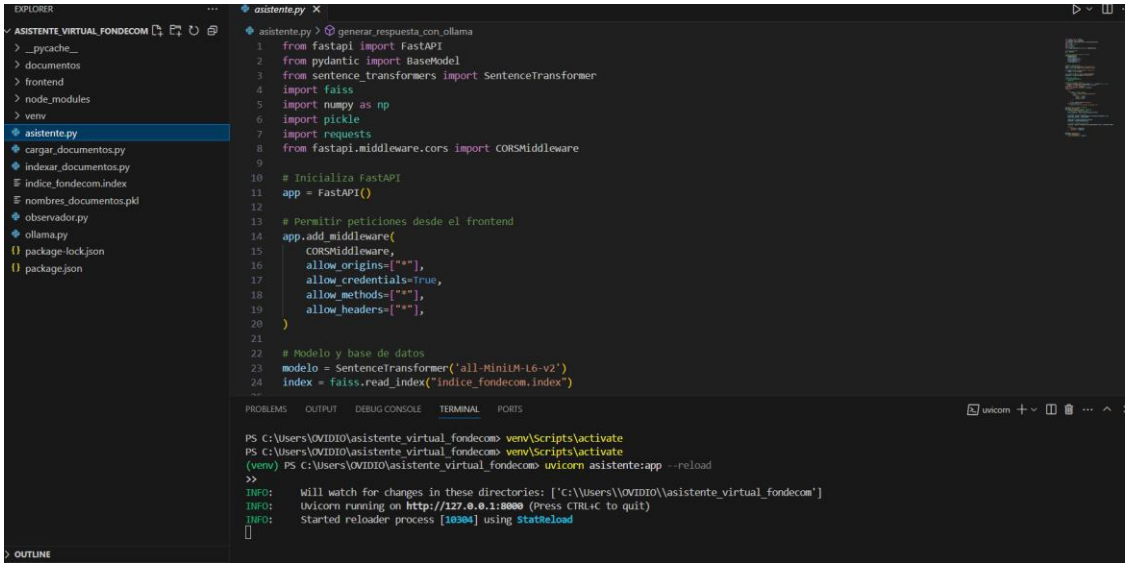

Ilustración 9. Servidor FastAPI ejecutándose mediante Uvicorn.

3.2 Modelo de toma de decisiones y análisis de desempeño

3.2.1. Modelo de toma de decisiones

El asistente virtual propuesto está diseñado para tomar decisiones automáticas en función de

la interpretación semántica de la pregunta del usuario y el contenido de los documentos corporativos. El modelo de toma de decisiones sigue un enfoque basado en la recuperación o búsqueda de información relevante mediante técnicas de procesamiento de lenguaje natural (PLN).

El proceso se divide en las siguientes etapas:

1. **Recepción de la pregunta**: el usuario ingresa una pregunta en lenguaje natural.

2. **Preprocesamiento**: se tokeniza el proceso mediante modelo embeddings preentrenados (Sentence Transformers)

3. **Vectorización**: la pregunta se convierte en un vector semántico utilizando modelos preentrenados como BERT.

4. **Búsqueda semántica**: se compara la pregunta vectorizada con los vectores de los documentos almacenados mediante FAISS.

5. **Selección de respuesta**: se extrae el fragmento más relevante del documento con la puntuación de similitud más alta.

6. **Generación de respuesta**: con la integración de Ollama (Mistral, llama3, Gemma), el modelo reformula la respuesta usando lenguaje claro y natural.

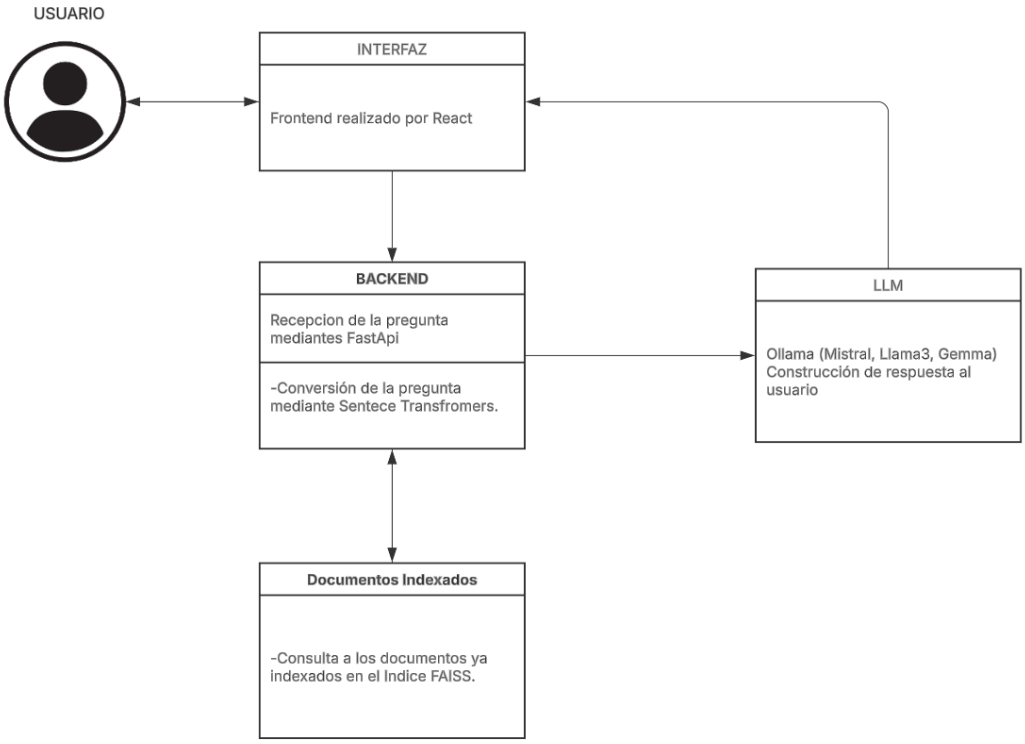

Ilustración 10. Arquitectura del modelo del asistente virtual.

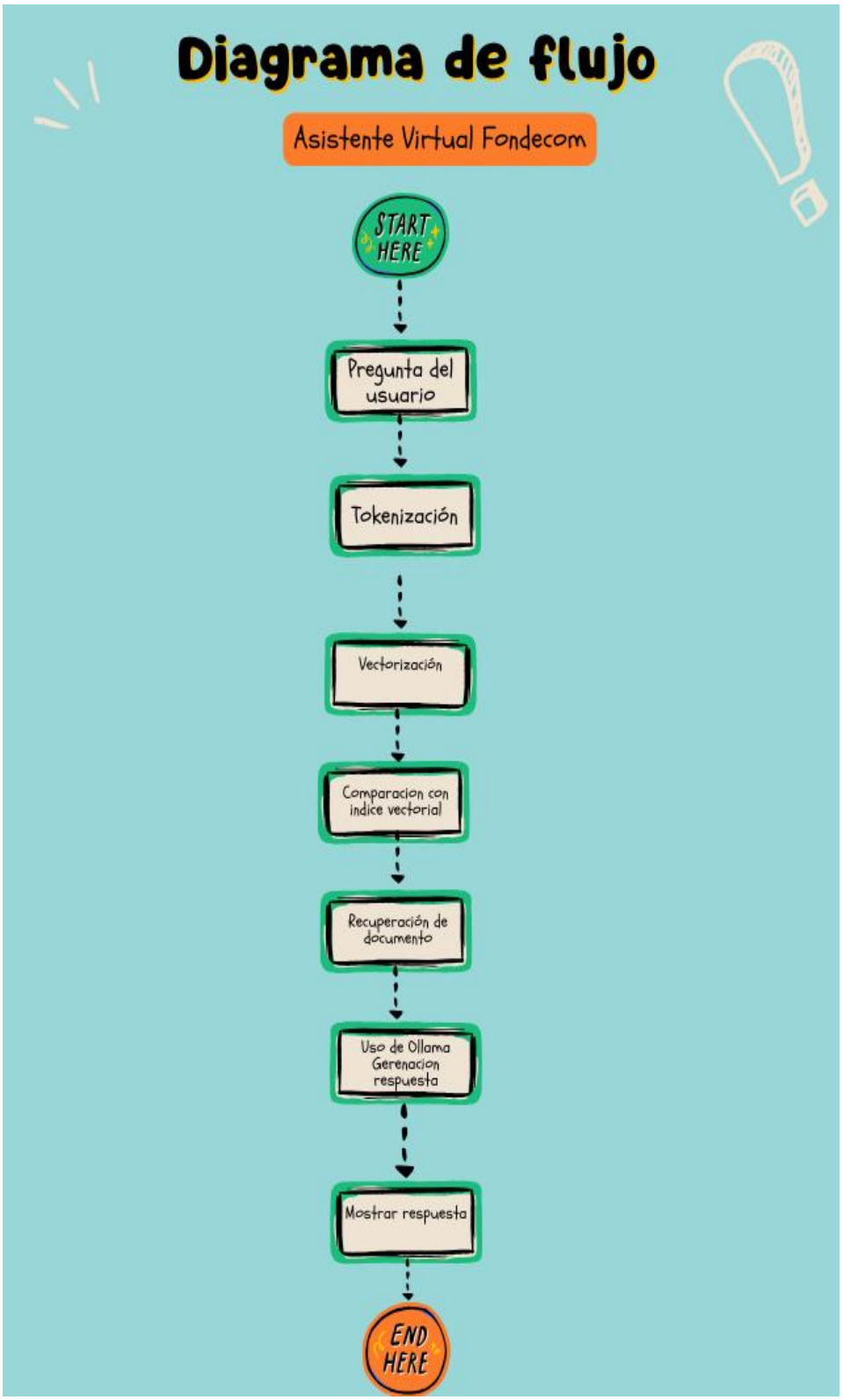

Ilustración 11. Diagrama de flujo del asistente virtual.

3.2.2. Análisis de desempeño

Se realiza 6 pruebas piloto con colaboradores del área TIC de Fondecom, quienes realizan consultas reales al asistente virtual. Las respuestas son calificadas manualmente como relevantes en el 90% de los casos, lo que evidencia la usabilidad del sistema en un entorno real, haciendo uso de tres modelos de lenguaje (Mistral, Llama3, y Gemma).

Tabla 1. Análisis de desempeño.

MODELO	TIEMPO ESTIMADO DE RESPUESTA	RESPUETA
MISTRAL	40 Segundos	Se preguntó lo siguiente: ¿Que es Fondecom? El asistente menciona que Fondecom es un fondo de empleados, siendo en este contexto, una respuesta precisa. Imagen respuesta 1
MISTRAL	44 segundos	Se preguntó lo siguiente: ¿Dónde está ubicado Fondecom? El resultado es nuevamente coherente con respecto a la información contenida en el documento del

		estatuto de Fondecom Imagen respuesta 2
LLAMA3	1 minuto con 20 Segundos	Se pregunta lo siguiente: ¿Qué es Fondecom? La respuesta es coherente al documentos del estatuto. Imagen respuesta 3
LLAMA3	45 segundos	Se preguntó lo siguiente: ¿ Dónde está ubicado Fondecom? la respuesta es coherente al documento suministrado. Imagen respuesta 4
GEMMA	1 minuto con 18 segundos	Se preguntó lo siguiente: ¿Que es Fondecom? El asistente menciona que Fondecom es un fondo de empleados, siendo en este contexto, una respuesta precisa. Imagen respuesta 5
GEMMA	28 segundos	Se preguntó lo siguiente: ¿ Dónde está ubicado Fondecom? la respuesta es coherente al documento indexado en el asistente virtual. Imagen respuesta 6

Ilustración 12. Prueba 1.

Ilustración 13. Prueba 2.

Ilustración 14. Prueba 3.

Ilustración 15. Prueba 4.

Ilustración 16. Prueba 5.

Ilustración 17. Prueba 6.

Conclusiones

La experiencia de la creación del asistente virtual para Fondecom permitió reflexionar sobre el uso e implementación de la inteligencia artificial y el machine learning ya que esto puede facilitar el acceso a la información dentro de una organización a partir del análisis de documentos internos. La implementación de un sistema capaz de entender preguntas en lenguaje natural se logró con el objetivo de crear una herramienta útil para responder dudas frecuentes de los empleados y/o usuarios.

Uno de los puntos más destacables del proyecto es que no fue necesario entrenar modelos desde cero debido al uso de modelos preentrenados como BERT, junto con motores de búsqueda semántica como FAISS, con estos se obtuvieron resultados muy acertados, lo cual permitió optimizar tiempo y recursos. Además, se evidenció que el trabajo de limpieza y estructuración de los documentos es fundamental para garantizar respuestas de calidad.

También se diseñó una interfaz web sencilla y amigable, lo cual ayuda a que cualquier usuario pueda interactuar fácilmente con el asistente, haciendo que el proceso de consulta sea mucho más rápido y práctico.

Finalmente, las pruebas realizadas mostraron que esta solución puede ser una gran aliada para mejorar la eficiencia operativa en Fondecom, pero se evidencia que es posible mejorar los tiempos de respuesta haciendo uso de recursos en la nube, por otro lado, demuestra que tiene mucho potencial para seguir creciendo y posiblemente implementarse en todas las áreas de la organización.

Bibliografía

Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805. (s.f.). Obtenido de https://arxiv.org/abs/1810.04805

FastAPI. (n.d.). FastAPI: Modern, fast (high-performance) web framework for building APIs with Python 3.7+. (s.f.). Obtenido de https://fastapi.tiangolo.com/

Harris, C. R., Millman, K. J., van der Walt, S. J., Gommers, R., Virtanen, P., Cournapeau, D., ... & Oliphant, T. E. (2020). Array programming with NumPy. Nature, 585(7825), 357–362. (s.f.). Obtenido de https://doi.org/10.1038/s41586-020-2649-2

Hugging Face. (n.d.). sentence-transformers documentation. (s.f.). Obtenido de https://www.sbert.net/

LangChain. (n.d.). LangChain documentation. (s.f.). Obtenido de https://docs.langchain.com/

Mozilla Developer Network (MDN). (2023). JavaScript documentation. (s.f.). Obtenido de https://developer.mozilla.org/en-US/docs/Web/JavaScript

Ollama. (2024). Run large language models locally. (s.f.). Obtenido de https://ollama.com

React. (n.d.). React – A JavaScript library for building user interfaces. (s.f.). Obtenido de https://react.dev/

Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. arXiv preprint arXiv:1908.10084. (s.f.). Obtenido de https://arxiv.org/abs/1908.10084

Uvicorn. (n.d.). Uvicorn documentation. (s.f.). Obtenido de https://www.uvicorn.org/

Vercel. (n.d.). Vite documentation. (s.f.). Obtenido de https://vitejs.dev/