ARQUITECTURA E IMPLEMENTACIÓN DE UN ASISTENTE VIRTUAL BASADO EN TECNOLOGÍAS OPENAI PARA EL ACCESO EFICIENTE A INFORMACIÓN ACADÉMICA

ARCHITECTURE AND IMPLEMENTATION OF A VIRTUAL ASSISTANT BASED ON OPENAI TECHNOLOGIES FOR EFFICIENT ACCESS TO ACADEMIC INFORMATION

Cristhian José Córdova Minalla [1*]

[1] Universidad Técnica de Machala. Estudiante de la Facultad de Ingeniería Civil. Ecuador. ORCID: https://orcid.org/0009-0002-9193-7001. Correo: ccordova5@utmachala.edu.ec

Jeiner Jair Mendieta Castro [2]

[2] Universidad Técnica de Machala. Estudiante de la Facultad de Ingeniería Civil. Ecuador. ORCID: https://orcid.org/0009-0008-0261-0263. Correo: jmendieta7@utmachala.edu.ec

Wilmer Braulio Rivas Asanza [3]

[3] Universidad Técnica de Machala. Docente de la Facultad de Ingeniería Civil. Ecuador. ORCID: https://orcid.org/0000-0002-2239-3664. Correo: wrivas@utmachala.edu.ec

Bertha Eugenia Mazón Olivo [4]

[4] Universidad Técnica de Machala. Docente de la Facultad de Ingeniería Civil. Ecuador. ORCID: https://orcid.org/0000-0002-2749-8561. Correo: bmazon@utmachala.edu.ec

*** Autor para correspondencia: ccordova5@utmachala.edu.ec**

Resumen

La presente investigación se centró en la propuesta de una arquitectura basada en tecnologías de OpenAI utilizando modelos avanzados de procesamiento de lenguaje natural (NLP) y técnicas de *fine-tuning* para desarrollar un asistente virtual orientado a la gestión y acceso eficiente a la información académica en instituciones de educación superior, tomando como caso de estudio la Universidad Técnica de Machala (UTMACH). Se empleó una metodología estructurada en dos actividades: la primera radicó en el diseño de una arquitectura modular y multicapa, que integró modelos de lenguaje, bases de datos vectoriales y

herramientas de conversión de voz a texto y viceversa. La segunda etapa se enfocó en la implementación práctica del asistente virtual, desarrollando una interfaz interactiva y realizando pruebas piloto en entornos reales para evaluar su funcionalidad y precisión. Los resultados, obtenidos a través de una matriz de confusión, reflejan una precisión del 96.88% en respuestas estrictamente correctas y del 97.50% al incluir respuestas parcialmente correctas. La evaluación del asistente evidencia la viabilidad de la arquitectura y su capacidad para gestionar información académica con alta precisión en distintos entornos, identificando áreas de mejora en la coherencia de ciertas respuestas.

Palabras clave: Inteligencia Artificial; Asistente virtual; OpenAI; Fine-tuning; NLP

Abstract

This research focused on proposing an architecture based on OpenAI technologies, utilizing advanced natural language processing (NLP) models and fine-tuning techniques to develop a virtual assistant aimed at managing and efficiently accessing academic information in higher education institutions, taking the Technical University of Machala (UTMACH) as a case study. The methodology was structured into two main activities: the first involved designing a modular and multi-layered architecture that integrated language models, vector databases, and voice-to-text and text-to-voice conversion tools. The second stage focused on the practical implementation of the virtual assistant, developing an interactive interface and conducting pilot tests in real environments to evaluate its functionality and accuracy. The results, obtained through a confusion matrix, showed an accuracy of 96.88% in strictly correct responses and 97.50% when including partially correct responses. The evaluation of the assistant demonstrates the feasibility of the architecture and its ability to manage academic information with high accuracy in different environments, identifying areas for improvement in the coherence of certain responses.

Keywords: *Artificial Intelligence; Virtual Assistant; OpenAI; Fine-Tuning; NLP*

Fecha de recibido: 14/01/2025
Fecha de aceptado: 11/03/2025
Fecha de publicado: 01/04/2025

Introducción

En años recientes, la digitalización de los procesos educativos ha ganado importancia en el sector de la educación a escala mundial. No obstante, en Latinoamérica, numerosas universidades enfrentan retos considerables para incorporar tecnologías de vanguardia que permitan un acceso eficaz a la información académica (Álvarez & Prieto, 2023; Crespo & Benavides, 2024; Ogosi Auqui, 2021). A pesar del aumento en la utilización de asistentes virtuales en diferentes sectores, su implementación en instituciones educativas de la región sigue siendo limitada debido a factores como los altos costos de desarrollo y mantenimiento, así como la falta de infraestructura tecnológica adecuada (Jara, 2015; Crespo & Benavides, 2024). Esto conduce

a la persistencia de procedimientos convencionales y manuales, como el envío de correos electrónicos, visitas presenciales a oficinas administrativas o consultas telefónicas para la resolución de dudas académicas, los cuales generan atascos, especialmente en periodos de alta demanda, afectando la calidad del servicio proporcionado a los alumnos (Mori & Palomino, 2021; Quesada, 2021).

La Universidad Técnica de Machala (UTMACH), similar a muchas otras instituciones en Ecuador, depende de estas prácticas tradicionales para gestionar las consultas de sus estudiantes relacionadas con programas académicos, coordinadores y contactos administrativos. Este enfoque manual no solo es ineficiente en términos de tiempo, sino que también incrementa la carga operativa sobre el personal administrativo, provocando demoras significativas en las respuestas. En un contexto donde la transformación digital evoluciona constantemente con avances en inteligencia artificial (IA), existe una necesidad urgente de innovar en la manera en que se gestiona la información académica.

Estudios recientes demuestran que los asistentes virtuales basados en IA pueden optimizar la atención al usuario, proporcionando respuestas rápidas y precisas (Brown et al., 2020; Langston et al., 2025). Por ejemplo, el proyecto UBOT, desarrollado en la Universidad Bernardo O'Higgins, empleó una arquitectura modular basada en prototipos evolutivos y utilizó Dialogflow de Google para la creación de agentes conversacionales. Esta tecnología permitió diseñar flujos comunicacionales efectivos, reconociendo patrones mediante algoritmos de aprendizaje automático, lo que mejoró significativamente las interacciones entre estudiantes y entornos virtuales, al responder consultas frecuentes y facilitar el acceso a la información institucional (Rubio et al., 2022).

Asimismo, un estudio en la Escuela Interamericana de Bibliotecología de la Universidad de Antioquia utilizó la metodología de *design thinking* para proponer una arquitectura inicial de chatbot. Este diseño se centró en identificar necesidades informativas clave mediante análisis cualitativo y categorización temática, optimizando los procesos comunicativos y administrativos de los estudiantes (Múnera et al., 2022). Estas iniciativas no solo demuestran la efectividad de las tecnologías de IA, como *Dialogflow* y el procesamiento de lenguaje natural, sino que también subrayan la pertinencia de desarrollar soluciones que respondan de manera integral a las necesidades académicas y administrativas.

No obstante, la adopción de estas tecnologías conlleva diversos desafíos. Quinde et al. (2024) advierte que, aunque la IA puede personalizar el aprendizaje y mejorar la evaluación, también plantea riesgos asociados con la dependencia tecnológica, la privacidad de datos y dilemas éticos. Estos factores resaltan la "necesidad de un enfoque equilibrado que maximice los beneficios de la IA" (Quinde et al., 2024, p. 187), minimizando sus riesgos y promoviendo un uso ético y consciente.

Si bien los estudios revisados ofrecen contribuciones significativas, ninguno de ellos propone una arquitectura para el desarrollo de un asistente virtual que pueda ser implementado en diversos ámbitos. En este marco, la propuesta se enfoca en presentar una arquitectura basada en tecnologías de OpenAI que emplea modelos avanzados de procesamiento de lenguaje natural (NLP) y técnicas de *fine-tuning*. A diferencia de tecnologías como Dialogflow de Google, que están diseñadas principalmente para flujos de conversación preestablecidos y aplicaciones específicas (Google Cloud, 2024). Los modelos de OpenAI, como GPT, destacan por su capacidad para generar respuestas altamente contextualizadas, adaptarse dinámicamente a consultas

complejas y manejar una amplia gama de tareas con un entrenamiento mínimo (Brown et al., 2020; OpenAI, 2021; Langston et al., 2025).

La arquitectura será aplicada en un caso de estudio en una institución educativa como la UTMACH, con el fin de ajustar y evaluar el modelo entrenado para atender sus necesidades académicas específicas. Para ello, la metodología se subdivide en dos actividades principales: la primera consiste en diseñar una arquitectura modular y multicapa basada en tecnologías de OpenAI; y la segunda, destinada a implementar de manera práctica el asistente virtual dentro de dicha arquitectura. Con este enfoque, se pretende mostrar la flexibilidad de la solución para adaptarse a distintos escenarios, ya sea en el ámbito educativo o en otros sectores, así como evaluar su nivel de precisión en la generación de respuestas mediante el análisis de una matriz de confusión. De esta forma, se sientan las bases para futuras implementaciones en diversas instituciones de la región. Además, las características de esta propuesta basada en OpenAI la convierten en una alternativa efectiva para cubrir un amplio espectro de requerimientos, tanto por texto como por voz, en tiempo real, dentro de una estructura diseñada para satisfacer dichas necesidades.

Materiales y métodos

Para alcanzar los objetivos establecidos, la metodología de esta investigación se organiza en dos actividades principales. La primera actividad consiste en la propuesta de una arquitectura modular y de múltiples capas, basada en tecnologías de OpenAI, diseñada para implementar un asistente virtual adaptable a diferentes ámbitos y capaz de responder satisfactoriamente a las consultas realizadas por los usuarios. En esta etapa se incluye el preprocesamiento de los datos institucionales, el ajuste fino de un modelo de lenguaje y la integración de bases de datos vectoriales junto con fuentes de información externas. Aunque esta arquitectura es adaptable a diversos contextos, su caso de estudio y prueba se enfocó en la Universidad Técnica de Machala (UTMACH) con el objetivo de atender sus necesidades académicas específicas.

La segunda actividad se enfoca en la implementación práctica del asistente virtual, incluyendo el diseño de una interfaz de usuario interactiva, la integración del modelo de lenguaje ajustado y la configuración de los sistemas necesarios para garantizar su funcionamiento idóneo. Finalmente, se llevaron a cabo pruebas piloto en entornos reales para validar la capacidad del asistente en la resolución de consultas académicas y administrativas de los usuarios.

Actividad 1: Arquitectura Propuesta para el asistente Virtual

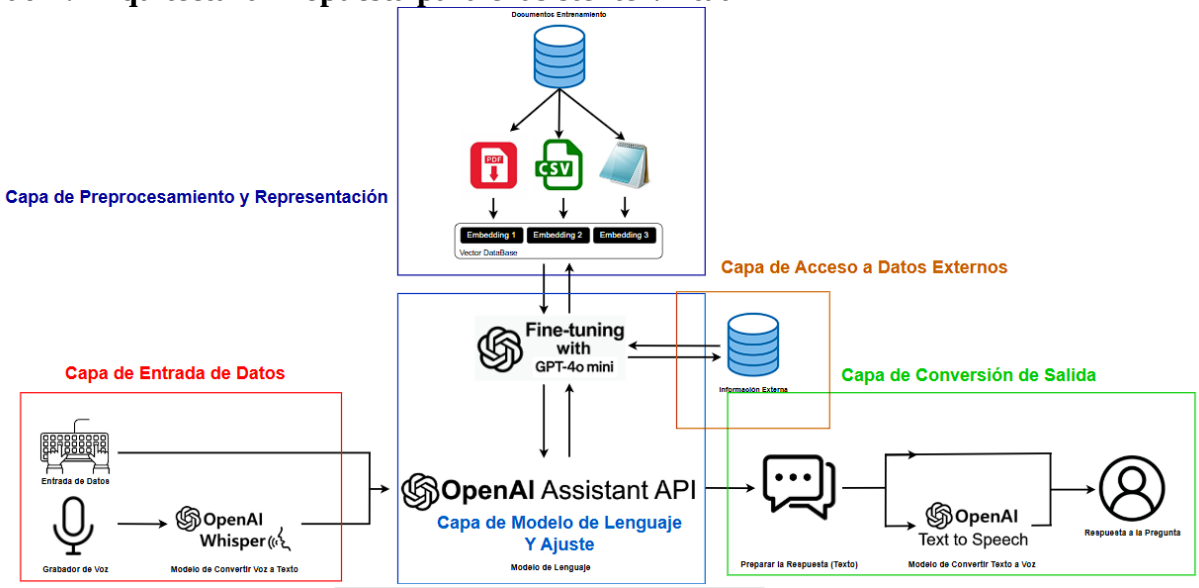

Figura 1. Arquitectura Propuesta para el asistente Virtual.
Fuente: Elaboración propia (2025).

En la Figura 1, se presenta la arquitectura modular diseñada para implementar un asistente virtual basado en tecnologías de OpenAI. Este esquema muestra cómo las distintas capas interactúan de forma integrada para permitir una comunicación eficiente entre los usuarios y el sistema. La arquitectura combina herramientas avanzadas de procesamiento de lenguaje natural (NLP), conversión de voz a texto y generación de respuestas dinámicas en tiempo real, garantizando flexibilidad y precisión en el manejo de consultas.
A continuación, se detallan las funciones y características principales de cada capa de la arquitectura:

Capa de Entrada de Datos

Esta capa se encarga de capturar las consultas realizadas por los usuarios a través de texto o voz. Para las consultas por voz, se utilizó OpenAI Whisper, una herramienta de vanguardia en la transcripción de audio a texto, garantizando alta precisión incluso en entornos con ruido (Yépez & Cruz, 2024). Las consultas en texto, en cambio, son enviadas directamente al sistema mediante una interfaz de usuario interactiva.

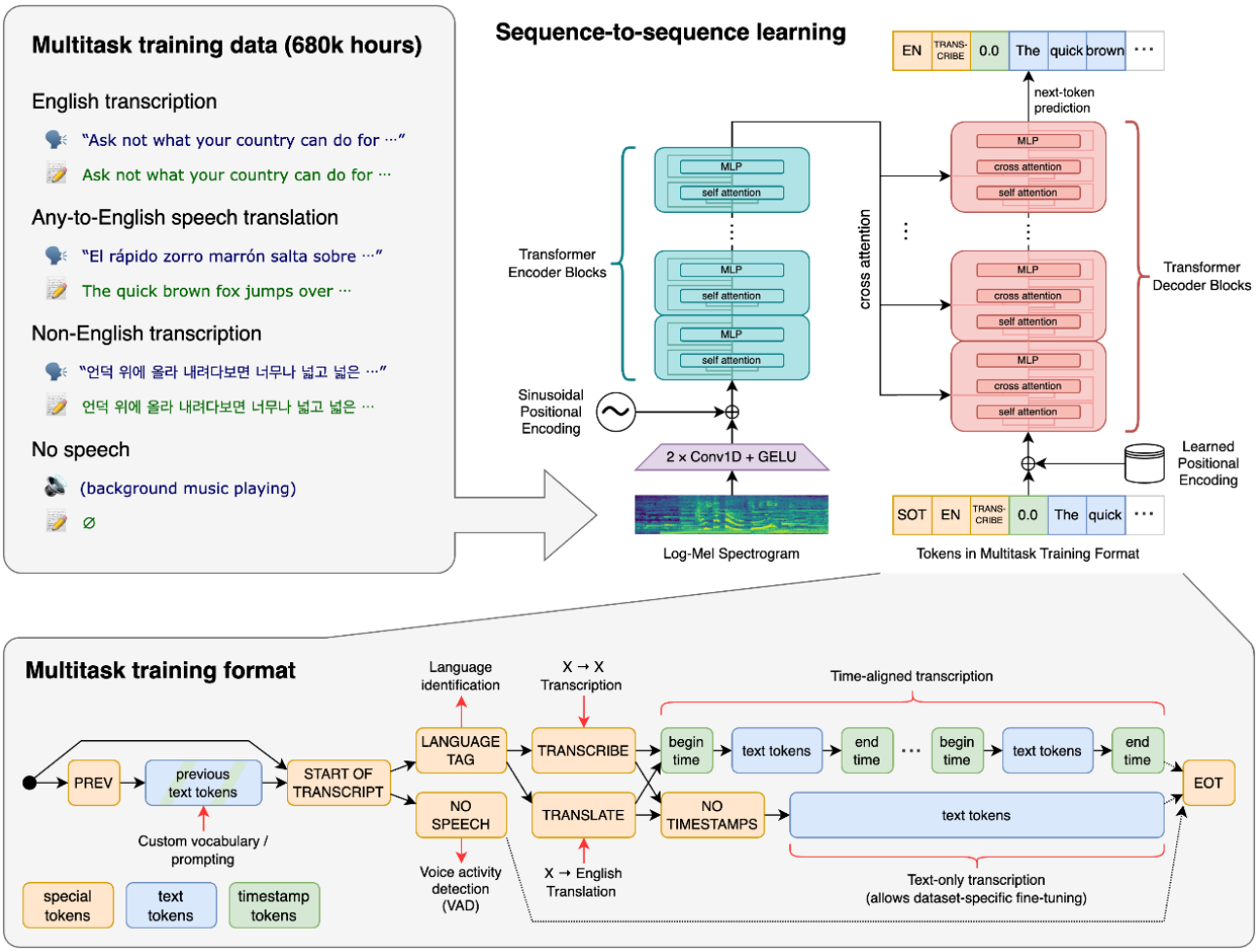

Figura 2. Proceso de Transcripción de Voz a Texto con OpenAI Whisper
Fuente: (OpenAI, 2021)

En la Figura 2, se presenta cómo OpenAI Whisper procesa datos de voz, convirtiéndolos en texto mediante aprendizaje multitarea. El modelo transcribe, traduce y detecta el idioma del audio, comenzando con la conversión del sonido a espectrogramas y finalizando con la generación de texto preciso y alineado temporalmente. Este flujo garantiza transcripciones claras y contextualmente relevantes, incluso en entornos con ruido o idiomas diversos.

Capa de preprocesamiento y representación

En esta etapa, los documentos institucionales (PDF, CSV, TXT) se procesan para generar *embeddings* mediante modelos de procesamiento de lenguaje natural. Estos *embeddings* son almacenados en una Base de Datos Vectorial, lo que permite realizar búsquedas rápidas y contextuales. Este paso asegura que la información relevante esté accesible para el modelo de lenguaje durante la generación de respuestas (Han et al., 2023).

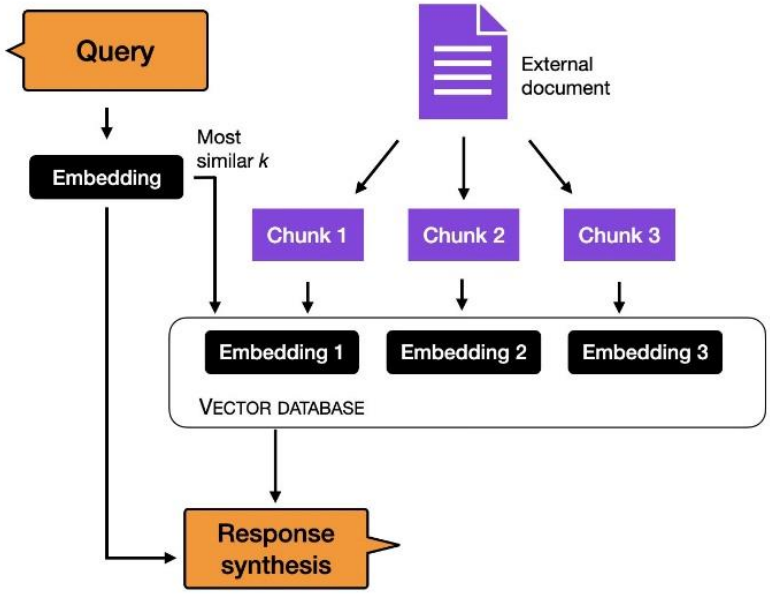

Figura 3. Proceso de Indexación.
Fuente: (Sebastian, 2023)

En la Figura 3, se presenta cómo se procesan documentos externos, dividiéndolos en fragmentos más pequeños *(chunks).* Cada uno de estos fragmentos se convierte en un vector o representación numérica *(embedding)* mediante un modelo de procesamiento de lenguaje natural. Dichos embeddings se almacenan en una base de datos vectorial, lo que permite realizar búsquedas rápidas basadas en similitud semántica (Cachay, 2024). Cuando un usuario envía una consulta *(query)*, esta también se traduce a un embedding y se compara contra los embeddings almacenados para identificar los fragmentos más pertinentes *(top-k).* Finalmente, estos fragmentos relevantes se utilizan para generar una respuesta *(response synthesis)* más contextualizada y precisa.

Capa de Acceso a Datos Externos

Esta capa permite al asistente virtual complementar su conocimiento almacenado en la Base de Datos Vectorial con información proveniente de fuentes externas. Cuando la información recuperada mediante embeddings no es suficiente para responder con precisión a una consulta, el sistema activa esta capa para realizar búsquedas en bases de datos relacionales como PostgreSQL, MySQL o cualquier otra fuente de almacenamiento estructurado. De esta manera, el asistente puede acceder a información actualizada y detallada en tiempo real, elevando la exactitud y fiabilidad de las respuestas sin necesidad de un reentrenamiento constante del modelo de lenguaje.

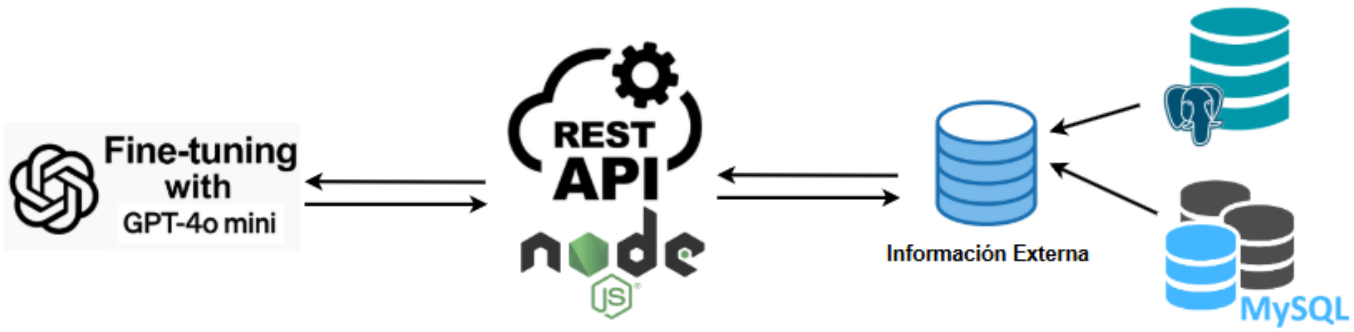

Figura 4. Integración del Asistente Virtual con Fuentes de Datos Externos
Fuente: Elaboración propia (2025)

En la Figura 4, se muestra cómo la capa de acceso a datos externos interactúa con el asistente virtual a través de una API REST construida en Node.js. Siempre que se realiza una solicitud que requiere información adicional, se utiliza una API para mediar entre el modelo de lenguaje y las bases de datos externas. A través de esta conexión, el asistente puede extraer datos estructurados desde PostgreSQL o MySQL, incorporándolos en la generación de respuestas. Este enfoque híbrido garantiza que el asistente sea más flexible y escalable, garantizando que la información utilizada para responder siempre esté actualizada.

Capa de Modelo de Lenguaje y Ajuste

El núcleo del sistema es el modelo de lenguaje GPT-4o mini, el cual fue sometido a un proceso de ajuste fino (fine-tuning) con datos específicos proporcionados por la UTMACH. Este proceso optimiza el modelo para generar respuestas contextualizadas y precisas, alineadas con las necesidades académicas y administrativas de la institución (Peña-Torres, 2024; Jain, 2022). Una vez ajustado, el modelo evalúa las consultas del usuario, selecciona la información relevante almacenada en la base de datos vectorial o en fuentes externas, y produce una respuesta coherente y específica.

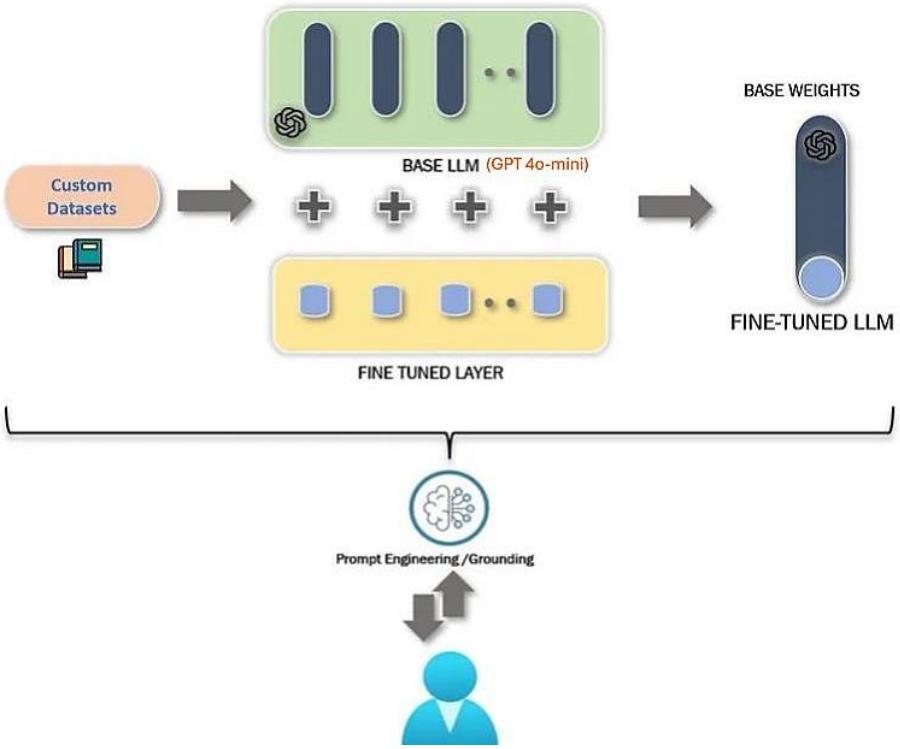

Figura 5. Ajuste fino OpenAI GPT-4o mini.
Fuente: (OpenAI, 2021)

En la Figura 5, se presenta el proceso de ajuste fino (*fine-tuning*) en los modelos de lenguaje de OpenAI tomando como ejemplo GPT-4o mini. En un primer paso, se toma el modelo base preentrenado, que posee una gran habilidad para procesar y generar texto gracias a su aprendizaje inicial con grandes bases de datos. A continuación, se incorporan los conjuntos de datos personalizados (*custom datasets*) diseñados para un dominio o tarea específica, de manera que se modifique el comportamiento del modelo base a ese contexto. Esta mezcla de modelo base y datos especializados establece las condiciones para el ajuste que en este caso consiste en añadir o modificar capas *(fine-tuned layer)* que permiten mayor conocimiento, es decir, permiten preservar la comprensión general del modelo, así como la precisión en la tarea objetivo.

El resultado es un modelo ajustado (*fine-tuned LLM*) capaz de responder con mayor precisión y coherencia a interrogantes que están relacionadas dentro del dominio para el que fue entrenado. Dicho modelo combina los pesos originales (*base weights*) con las nuevas capas de ajuste; es decir, se están combinando tanto el conocimiento preexistente como el refinamiento específico logrado a partir de los conjuntos de datos personalizados. Finalmente, la ingeniería de *prompts* (*prompt engineering*) juega un papel esencial, ya que garantiza que la consulta se estructure adecuadamente y el modelo aproveche todo su potencial, respondiendo de manera óptima dentro del contexto definido.

Procesamiento de Datos

El proceso de *fine-tuning* del modelo GPT-4o mini requiere que los datos estén estructurados adecuadamente para el aprendizaje supervisado. En este caso, el ajuste fino se realiza utilizando el formato JSONL (JSON Lines), que facilita y flexibiliza el entrenamiento de datos. Este formato ayuda a organizar los datos en registros estructurados donde cada línea es un objeto JSON separado que contiene una pregunta-respuesta o un diálogo contextualizado.

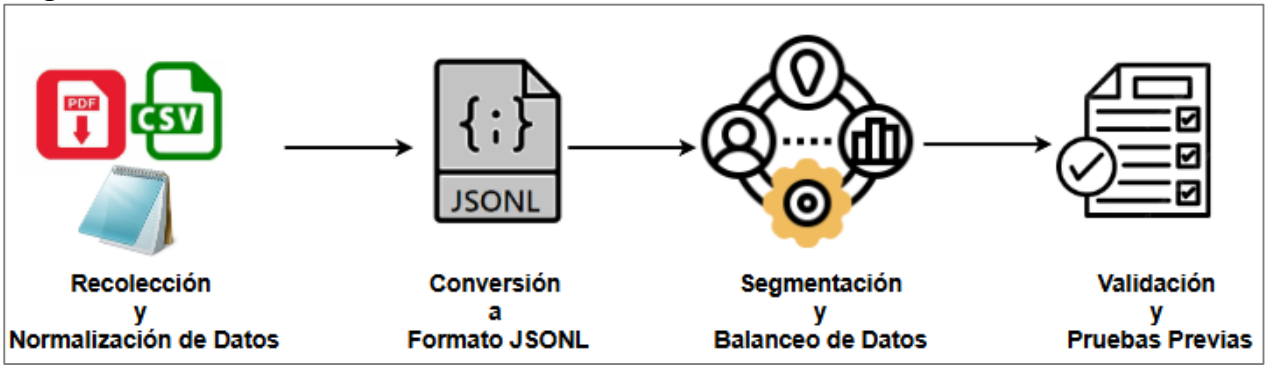

Figura 6. Procesamiento de Datos
Fuente: Elaboración propia (2025)

En la Figura 6, se presenta el proceso de preparación de datos necesarios para el ajuste fino (*fine-tuning*) del modelo GPT-4o mini centrado en la estructura y calidad de la información que se utilizará durante el entrenamiento. El primer paso es la recolección y normalización, donde se recopilan documentos en PDF, CSV o TXT, provenientes de fuentes institucionales. Simultáneamente, esos datos son filtrados y estructurados para que sean relevantes, útiles al modelo y no estén sobreescritos.

Luego, los datos normalizados se convierten al formato JSONL. Durante cada interacción de entrenamiento, se asegura que se mantenga el contexto conversacional. Este formato permite estructurar los datos en pares de entrada y salida, haciendo que el modelo pueda procesarlos fácilmente. Después, se realizan la segmentación y balanceo de datos, organizando las consultas en distintas categorías para mejorar la variedad del entrenamiento. Finalmente, se procede a la validación y pruebas previas utilizando un subconjunto de datos y revisando la coherencia y la relevancia de las respuestas para optimizar el modelo antes del entrenamiento final.

Capa de Conversión de Salida

Una vez que el modelo genera la respuesta, esta capa se encarga de entregarla al usuario en el formato solicitado, ya sea como texto mostrado directamente en la interfaz de usuario o como audio utilizando la tecnología *Text-to-Speech*. Este enfoque permite atender a una amplia variedad de necesidades, incluyendo el acceso a usuarios con limitaciones visuales o preferencias por respuestas auditivas, garantizando así una interacción inclusiva y eficiente.

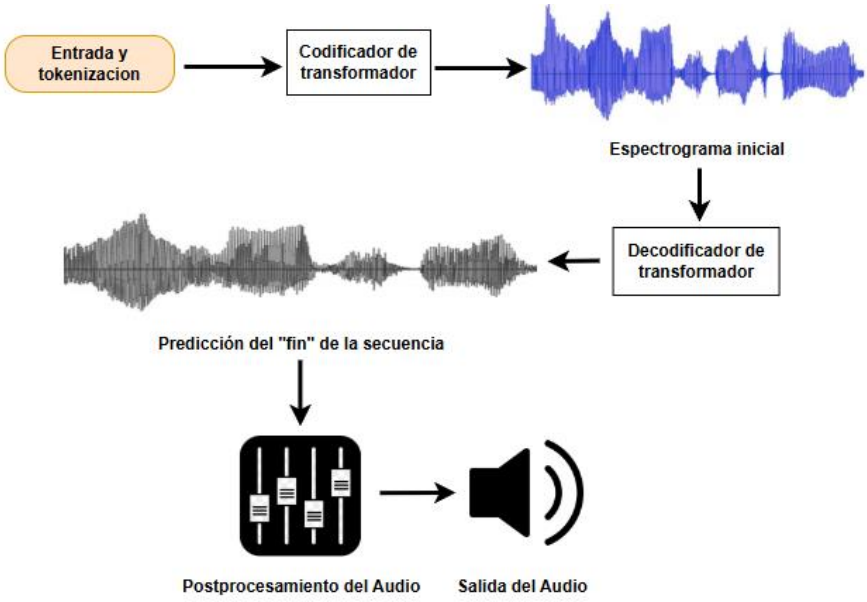

Figura 7. Flujo de Capa de Conversión de Respuesta
Fuente: Elaboración propia (2025)

En la Figura 7, se presenta el flujo de síntesis de texto a voz (TTS) basado en transformadores. El primer paso concierne a la entrada y tokenización del texto a ser sintetizado. La entrada se divide en tokens, que son las unidades lingüísticas básicas que el modelo puede procesar. Como tal, obtiene una representación numérica mientras se conserva el significado y la estructura del texto original. A continuación, la información se pasa al codificador transformador. La principal tarea del codificador transformador es capturar las relaciones contextuales entre los tokens y producir la primera versión de un espectrograma. Los espectrogramas son representaciones visuales de las frecuencias y amplitudes de los sonidos a medida que cambian con el tiempo. Por lo tanto, sirven como la descripción básica de cómo suena el habla durante un cierto periodo de tiempo.

A continuación, el decodificador del transformador toma el primer espectrograma y predice toda la secuencia de audio, incluyendo el momento en que la locución debe terminar (etiquetado como "fin de la secuencia"). De esta manera, el modelo aprende no solo a producir contenido fonético, sino también la duración del audio. El siguiente paso es uno de post-procesamiento, donde el espectrograma pasa por un procesamiento de audio, y se corrigen cualquier artefacto, discrepancia de volumen y otras características que distorsionan el sonido para hacerlo más realista y natural. En última instancia, este resultado se convierte en una salida de audio que corresponde a la versión hablada del texto inicial, de una manera que sea lo más natural y humana posible.

Actividad 2: Implementación del Asistente Virtual

La segunda actividad de esta investigación consistió en la implementación práctica del asistente virtual, desarrollada utilizando un enfoque basado en tecnologías modernas y librerías avanzadas. Este proceso se estructuró en varias etapas, comenzando con la configuración del entorno de desarrollo, el diseño de la interfaz de usuario y la integración de herramientas para la interacción dinámica entre el asistente y los usuarios.

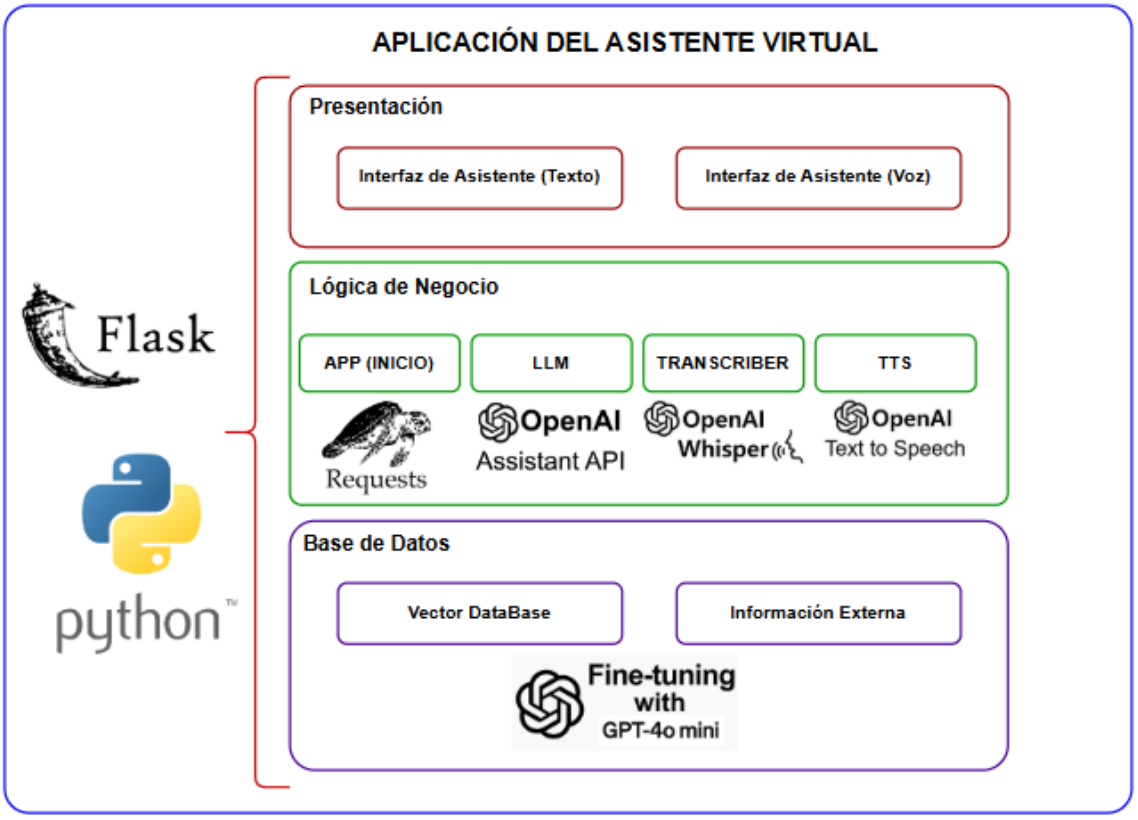

Figura 8. Tecnologías y Herramientas utilizadas en el Asistente Virtual
Fuente: Elaboración propia (2025)

En la Figura 8, se presenta la estructura general de la implementación del asistente virtual, organizada en tres capas principales: Presentación, Lógica de Negocio y Base de Datos. Cada una de estas capas desempeña un rol fundamental para garantizar la interacción eficiente entre los usuarios y el asistente, así como el procesamiento y la generación de respuestas adaptadas a las necesidades académicas. A continuación, se detalla la funcionalidad de cada capa:

Capa de Presentación

Proporciona la interacción directa entre el usuario y el asistente en dos modalidades principales:

- **Interfaz de Asistente (Texto):** Desarrollada con el framework Flask, permitiendo a los usuarios interactuar mediante mensajes de texto. Se emplean plantillas HTML y CSS para ofrecer una experiencia de usuario estilizada y receptiva.
- **Interfaz de Asistente (Voz):** Implementada mediante la integración de PyAudio para la grabación y reproducción de audio. Combina la conversión de voz a texto (Whisper) y texto a voz (TTS) para brindar una experiencia más inclusiva.

Capa de Lógica de Negocio

Constituye el núcleo funcional del asistente, gestionando la comunicación con el modelo de lenguaje y coordinando la lógica de la aplicación:

- **APP (Inicio):** Usa Flask y Flask-Session para controlar las rutas principales, la inicialización de sesiones y el historial de mensajes.
- **LLM (Modelo de Lenguaje):** Basado en la API de OpenAI, se encarga de generar respuestas personalizadas según las consultas del usuario. La comunicación de datos entre el servidor y el modelo se gestiona mediante la librería *requests*.
- **TRANSCRIBER:** Emplea el modelo Whisper de OpenAI para convertir grabaciones de audio en texto con gran precisión, a través de la clase *Transcriber*.
- **TTS:** Implementa la API de Text-to-Speech de OpenAI para convertir texto en archivos de audio (formato MP3). Esta funcionalidad se gestiona mediante la clase *TTS*.

Capa de Base de Datos

Se encarga de organizar y almacenar la información relevante para el asistente:

- **Vector DataBase:** Almacena los *embeddings* generados a partir de documentos institucionales y académicos, permitiendo búsquedas contextuales eficientes.
- **Información Externa:** Permite la integración con bases de datos relacionales y otros sistemas de almacenamiento estructurado, como PostgreSQL o MySQL. Esta capa complementa la información almacenada en la base de datos vectorial al proporcionar acceso a datos en tiempo real, asegurando que el asistente pueda recuperar información actualizada y responder de manera más precisa a consultas que requieren detalles externos.

Resultados y discusión

Los resultados obtenidos a partir de la matriz de confusión permiten evaluar la precisión del asistente virtual propuesto en la gestión y acceso a la información académica en la Universidad Técnica de Machala (UTMACH). La evaluación del desempeño del asistente virtual se basa en la clasificación de las respuestas generadas en las siguientes categorías:

- **Verdaderos Positivos (VP):** Respuestas generadas correctamente por el asistente, proporcionando información precisa y relevante con base en los datos disponibles.
- **Falsos Positivos (FP):** Respuestas incorrectas o irrelevantes, lo que sugiere una interpretación errónea de la información académica.
- **Falsos Negativos (FN):** Casos en los que el asistente no proporciona una respuesta, a pesar de que existe información relevante en la base de datos.
- **Respuestas Parcialmente Correctas:** Respuestas en las que al menos un **80% de la información** es veraz, pero presentan inexactitudes que afectan su fiabilidad.
- **Verdaderos Negativos (VN):** Preguntas a las que el asistente no responde debido a la ausencia de información académica relevante, evitando generar contenido especulativo o erróneo.

A partir de esta categorización, se analizó la precisión del modelo, determinando su capacidad para distinguir entre respuestas correctas y aquellas que requieren ajustes. La evaluación incluyó la identificación de posibles sesgos y errores sistemáticos, con el objetivo de optimizar el rendimiento del asistente mediante técnicas de

(*fine-tuning)*. Los resultados obtenidos proporcionan una base para futuras mejoras en la coherencia y precisión de las respuestas generadas.

Evaluación del desempeño del asistente virtual

La matriz de confusión presentada a continuación evidencia la distribución de las respuestas en correctas (VP), incorrectas (FP), no generadas (NF), respuestas correctamente evitadas por falta de información (VN) y respuestas parcialmente correctas (aproximadamente un 80% de información útil, pero con cierta información que afecta su veracidad).

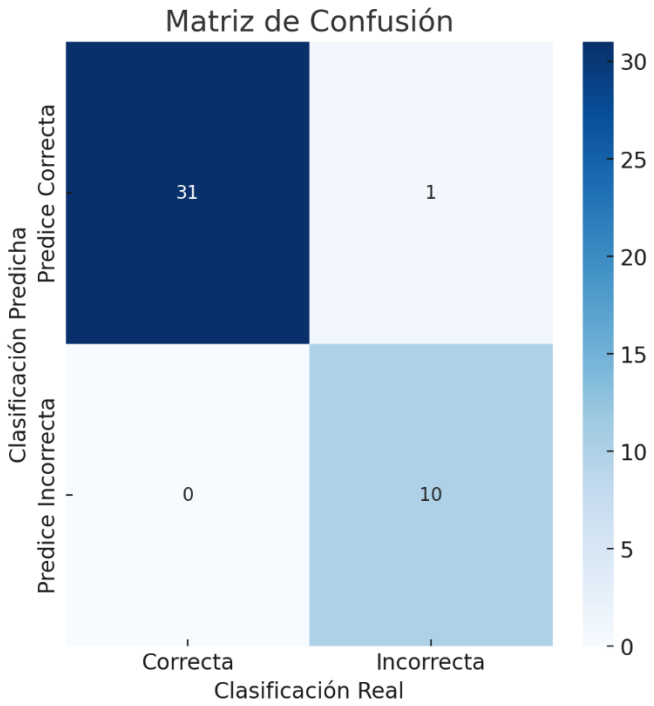

Figura 9. Resultados de la Matriz de Confusión
Fuente: Elaboración propia (2025)

La Figura 9 nos presenta la matriz de confusión resultante del desempeño del asistente, permitiendo visualizar la distribución de las respuestas generadas. En ella, se evidencia una alta tasa de respuestas correctas (VP = 31), lo que nos revela que el modelo ha logrado generar respuestas precisas en la mayoría de los casos. No obstante, la presencia de un Falso Positivo (FP = 1) manifiesta la existencia de errores en la recuperación de la información o en la generación de respuestas, potencialmente asociadas a sesgos en el entrenamiento del modelo o ambigüedades léxico-semánticas en las entradas de usuario. Cabe destacar que la inexistencia de Falsos Negativos (FN = 0) refleja que el asistente no omitió responder a ninguna pregunta válida con información disponible, mientras que la detección de 10 Verdaderos Negativos (VN) demuestra que el modelo tiene la capacidad de validar las consultas que carecen de información relevante.

La Tabla 1 expone el desglose detallado de las respuestas generadas por el asistente según cada categoría de consulta. Este análisis facilita la evaluación de la precisión estricta del modelo, determinada por las respuestas correctas y los falsos positivos, así como la precisión ajustada, que incorpora las respuestas parcialmente correctas.

Tabla 1. Análisis de las respuestas en la Matriz de Confusión.

Categoría	VP	FP	FN	VN	Parcialmente Correctas (≈80%)	Total Preguntas
Preguntas Generales	7	1	0	0	2	10
Carrera de Tecnología de la Información	8	0	0	0	2	10
Carrera de Ingeniería Civil	8	0	0	0	2	10
Carrera de Ingeniería Ambiental	8	0	0	0	2	10
Preguntas No Académicas	0	0	0	10	0	10
Total General	31	1	0	10	8	50

Nota: Se llevó a cabo una evaluación de 50 respuestas emitidas por el asistente.

La precisión estricta (VP) del modelo se calcula con la ecuación:

$$Precisión = \frac{VP}{VP + FP}$$

Sustituyendo los valores obtenidos:

$$Precisión = \frac{VP}{VP + FP} = \frac{31}{31 + 1} = \frac{31}{32} = 0.96875\ (96.88\%)$$

La precisión ajustada (VP + Parcialmente Correctas 80%) del modelo se calcula con la ecuación:

$$Precisión = \frac{VP + P}{VP + P + FP}$$

Sustituyendo los valores obtenidos:

$$Precisión = \frac{VP + P}{VP + P + FP} = \frac{31 + 8}{31 + 8 + 1} = \frac{39}{40} = 0.975\ (97.50\%)$$

Discusión de los resultados

Los resultados de la evaluación del asistente validan la viabilidad de una arquitectura basada en tecnologías de OpenAI, destacando su capacidad de adaptación a distintos entornos mediante modelos de Procesamiento del Lenguaje Natural (NLP) y técnicas de ajuste fino *(fine-tuning)*. En este estudio, la implementación del asistente en la Universidad Técnica de Machala (UTMACH) permitió evaluar su precisión en la gestión de información académica, proporcionando un análisis detallado de su desempeño.

El análisis mediante la matriz de confusión evidenció que el asistente alcanzó una precisión del 96.88%, lo que indica una alta capacidad para generar respuestas correctas y relevantes. Además, al considerar las respuestas parcialmente correctas, la precisión ajustada se incrementó a 97.50%, lo que sugiere que, pese a ciertos errores menores, el modelo sigue proporcionando información útil para el usuario.

En comparación con estudios previos, como el de Nguyen et al. (2021), quienes desarrollaron un chatbot utilizando el framework Rasa y el modelo DIET (Dual Intent and Entity Transformer) con una precisión del 81.6% en la detección de intenciones y extracción de entidades en consultas sobre admisión universitaria, el modelo propuesto en este estudio muestra una ventaja significativa. Esta mejora se atribuye a la integración de tecnologías avanzadas de OpenAI, las cuales ofrecen una comprensión más profunda del lenguaje natural y una mayor capacidad de adaptación a contextos específicos mediante *fine-tuning*.

Sin embargo, a pesar de los resultados favorables, persisten desafíos en la coherencia y precisión de las respuestas, particularmente en la interpretación de preguntas ambiguas o incompletas. En este sentido, futuras mejoras deberían enfocarse en optimizar el manejo de consultas complejas, reducir la generación de respuestas parcialmente correctas y fortalecer los mecanismos de validación semántica del modelo.

Conclusiones

Los hallazgos obtenidos tras la implementación y evaluación del asistente virtual basado en OpenAI confirman su eficacia como una solución para mejorar el acceso a la información académica en la Universidad Técnica de Machala (UTMACH). La arquitectura modular de múltiples capas, diseñada para integrar modelos avanzados de Procesamiento del Lenguaje Natural (NLP) y técnicas de fine-tuning, ha demostrado un desempeño altamente satisfactorio. Esta estructura ha facilitado la generación de respuestas precisas y contextualizadas, validando así la efectividad del enfoque adoptado.

El análisis de la matriz de confusión reveló que el asistente alcanzó una precisión estricta del 96.88% y una precisión ajustada del 97.50%, lo que reafirma su capacidad para proporcionar respuestas acertadas en el entorno académico. Sin embargo, a pesar de estos resultados positivos, se identificaron áreas de mejora, especialmente en la gestión de consultas complejas y la integración con los sistemas internos de la universidad. Para optimizar el rendimiento del asistente, se recomienda mejorar la calidad del entrenamiento del modelo, organizando los datos en formato JSONL de manera más estructurada. Esto implica la recopilación, estandarización y segmentación de datos, garantizando que las consultas abarquen un espectro más amplio de escenarios. Además, la adopción de técnicas avanzadas de ajuste fino y balanceo de datos podría minimizar errores semánticos y reducir la cantidad de respuestas parcialmente correctas, mejorando así la precisión global del asistente.

Referencias

Álvarez, M., & Prieto, P. (2023). Presentación del Dossier temático: “La educación superior en la era digital”. *Revista Educación Superior y Sociedad (ESS)*, *35*(2), 28-45. https://doi.org/https://doi.org/10.54674/ess.v35i2.879

Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., . . . Amodei, D. (2020). Language Models are Few-Shot Learners. *Computer Science - Computation and Language, abs/2005.14165*, 32-40. https://doi.org/https://doi.org/10.48550/arXiv.2005.14165

Cachay, A. W. (2024). Evaluation on embeddings application for spanish automatic text clustering. *Ingeniare. Revista chilena de ingeniería, 32*, 21. https://doi.org/http://dx.doi.org/10.4067/s0718-33052024000100221

Crespo, J., y Benavides, J. (2024). Beneficios y desafíos de los asistentes virtuales en el aprendizaje. *LATAM Revista Latinoamericana De Ciencias Sociales Y Humanidades, 5*(2), 685–700. https://doi.org/https://doi.org/10.56712/latam.v5i2.1909

Google Cloud. (2024). *Documentación de Dialogflow ES*. Google Cloud: https://cloud.google.com/dialogflow/es/docs?hl=es-419

Han, Y., Liu, C., y Wang, P. (2023). A comprehensive survey on vector database: Storage and retrieval technique, challenge. *ArXiv*, 1-13. https://doi.org/https://doi.org/10.48550/arXiv.2310.11703

Jain, S. M. (2022). *Introduction to Transformers for NLP: With the Hugging Face Library and Models to Solve Problems.* Apress, Berkeley, CA. https://doi.org/https://doi.org/10.1007/978-1-4842-8844-3

Jara, I. (Diciembre de 2015). *Infraestructura digital para educación. Avances y desafíos para Latinoamérica.* UNESCO: https://siteal.iiep.unesco.org/investigacion/1719/infraestructura-digital-educacion-avances-desafios-latinoamerica?utm_source=chatgpt.com

Langston, E., Hattakitjamroen, V., Hernandez , M., Soo Lee, H., Mason, H., Louis-Charles, W., . . . Boot, W. (2025). Exploring artificial intelligence-powered virtual assistants to understand their potential to support older adults' search needs. *Human Factors in Healthcare, 7*, 100092. https://doi.org/https://doi.org/10.1016/j.hfh.2025.100092

Mori, D., y Palomino, G. (2021). Análisis de la calidad de los servicios educativos en Latinoamérica. *Ciencia Latina Revista Científica Multidisciplinar, 5*(6), 12082-12097. https://doi.org/http://dx.doi.org/10.37811/cl_rcm.v5i6.1217

Múnera, M., Salazar , L., & Osorio , A. (2022). Estudio inicial de un chatbot para estudiantes de la modalidad virtual de la Escuela Interamericana de Bibliotecología. *Investigación bibliotecológica, 36*(90), 13-30. https://doi.org/https://doi.org/10.22201/iibi.24488321xe.2022.90.58452

Nguyen, M. T., Tran-Tien, M., Viet, A. P., Vu, H. T., y Nguyen, V. H. (2021). Building a Chatbot for Supporting the Admission of Universities. *2021 13th International Conference on Knowledge and Systems Engineering (KSE)*, 1-16. https://doi.org/https://doi.org/10.1109/KSE53942.2021.9648677

OGOSI AUQUI, J. A. (2021). Chatbot del proceso de aprendizaje universitario: Una revisión sistemática. *Alpha Centauri, 2*(2), 29–43. https://doi.org/https://doi.org/10.47422/ac.v2i2.33

OpenAI. (2021). *OpenAI API documentation*. OpenAI Platform: https://platform.openai.com/docs/overview

Peña-Torres, J. (2024). Towards an improved of teaching practice using Sentiment Analysis in Student Evaluation. *Ingeniería y competitividad, 26*(2), e-21013759. https://doi.org/https://doi.org/10.25100/iyc.v26i2.13759

Quesada, J. (2021). Calidad del servicio administrativo: impacto sobre el compromiso, la satisfacción y el rendimiento de estudiantes universitarios. *Revista Universidad & Empresa, 23*(41), 1-42. http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0124-46392021000200029&lang=es

Quinde, V., García, S., & Tenelanda , D. (2024). La Inteligencia Artificial y su utilidad en el campo Académico. Un Análisis desde la perspectiva del Universitario. *Conrado, 20*(99), 187-193. http://scielo.sld.cu/pdf/rc/v20n99/1990-8644-rc-20-99-187.pdf

Rubio, J., Neira, T., Molina, D., & Vidal, C. (2022). Proyecto UBOT: asistente virtual para entornos virtuales de aprendizaje. *Información tecnológica, 33*(4), 85-92. https://doi.org/https://dx.doi.org/10.4067/S0718-07642022000400085

Sebastian, R. (2023). *Machine Learning Q and AI.* Leanpub. https://soclibrary.futa.edu.ng/books/Machine%20Learning%20Q%20and%20AI%20(Sebastian%20Raschka,%20PhD)%20(Z-Library).pdf

Yépez, V., y Cruz, J. (2024). Inteligencia artificial en la transcripción de entrevistas. *Contratexto*(41), 183-202. https://doi.org/http://dx.doi.org/10.26439/contratexto2024.n41.6750