Pascual Research: Sistema Inteligente de Generación y Análisis de Información Académica Basado en Modelos de Lenguaje

Jaramillo Méndez Juliana

INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO
FACULTAD DE INGENIERÍA
INGENIERÍA EN DESARROLLO DE SOFTWARE
MEDELLÍN
2025

PASCUAL RESEARCH: SISTEMA INTELIGENTE DE GENERACIÓN Y ANÁLISIS DE INFORMACIÓN ACADÉMICA BASADO EN MODELOS DE LENGUAJE

Juliana Jaramillo Méndez

Trabajo de grado para optar al título de Ingeniero de Software

Asesor
Rubén Darío Fonnegra Tarazona
Doctor en Ingeniería

INSTITUCIÓN UNIVERSITARIA PASCUAL BRAVO
FACULTAD DE INGENIERÍA
INGENIERÍA EN DESARROLLO DE SOFTWARE
MEDELLÍN
2025

Contenido

Pág.

Tabla de contenido

Lista de figuras

Lista de tablas

Lista de anexos

Resumen

PASCUAL RESEARCH: SISTEMA INTELIGENTE DE GENERACIÓN Y ANÁLISIS DE INFORMACIÓN ACADÉMICA BASADO EN MODELOS DE LENGUAJE

Juliana Jaramillo Méndez

El presente trabajo surgió ante la necesidad de optimizar el acceso a la información científica perteneciente a los diversos documentos dentro de la Institución Universitaria Pascual Bravo (IUPB), donde la dispersión de la información y el creciente volumen de publicaciones representan un reto para la reutilización de las fuentes académicas para estudiantes e investigadores de la institución. Dado el auge, la gran comunidad y la expansión acelerada de la inteligencia artificial y, en particular, de los Grandes Modelos de Lenguaje Natural (LLM), se identificó la oportunidad de aprovechar estas herramientas como apoyo en la centralización y consulta de conocimiento académico de manera ágil y confiable.

Para lo anterior, se entrenó un modelo de lenguaje tipo chat con bases de datos generadas a partir de documentos académicos propios de la IUPB, orientado a responder preguntas con contenido relevante y verídico. Este proceso incluyó la recopilación y estructuración de un dataset institucional por medio de un proceso de generador y juez con apoyo de otras LLM existentes; las selección y fine-tuning del modelo más adecuado basado en el costo de recursos-beneficio; la evaluación de su desempeño mediante pruebas adaptadas al español del Benchmark GLUE; y la implementación de una interfaz conversacional en el servicio de Telegram que permitió una interacción práctica y gratuita con el modelo.

A pesar de los retos encontrados, se logró cumplir con los objetivos propuestos, pues el modelo demostró coherencia en sus respuestas, utilidad práctica y potencial para fortalecer procesos de investigación institucional lo que deja en evidencia el rápido avance del campo de la inteligencia artificial y deja abiertas las posibilidades a futuras mejoras relacionadas con la ampliación del presente proyecto.

Abstract

The current work arose from the need to optimize the access to scientific information contained in various documents within the Pascual Bravo University Institution (IUPB), where the dispersion of information and the growing volume of publications represent a challenge for the reuse of academic sources by students and researchers at the institution. Given the peak, the big community and the fast expansion of artificial intelligence and, particularly, of the Large Language Models (LLMs), an opportunity was identified to leverage these tools to support the centralization and consultation of academic knowledge in an agile and reliable way.

To this end, a chat-type language model was trained with databases generated from the IUPB's own academic documents, aimed at answering questions with relevant and accurate content. This process included the collection and structuring of an institutional dataset through a generator and judge pipeline with the support of other existing LLMs; the selection and fine-tuning of the most appropriate model based on the cost-benefit; the evaluations of its performance through tasks adapted to spanish from the GLUE Benchmark; and the implementation of a conversational interface in the Telegram service that allowed the practical and free interaction with the model.

Despite the challenges encountered, the proposed objectives were achieved, because the model demonstrated coherence on its responses, practical usefulness and potential to strengthen institutional research processes which highlights the fast progress in the field of artificial intelligence and opens up possibilities for future improvements related to the growth of this project.

Glosario

Datos sintéticos: datos generados de forma artificial mediante algoritmos, modelos estadísticos o modelos generativos. Se usan principalmente ante la dificultad de escasez de conjuntos de datos de entrenamiento. Buscan mantener la mayor cantidad de características similares con los datos reales.

Deep Learning: subcampo del Machine Learning que utiliza redes neuronales artificiales profundas con muchas capas para modelar patrones complejos en grandes volúmenes de datos. Permite resolver tareas complejas como reconocimiento de voz y procesamiento de lenguaje natural.

Fine-Tuning: proceso mediante el cual un modelo pre entrenado se ajusta según la necesidad usando un conjunto de datos adicional y específico. Permite mejorar el rendimiento del modelo en dominios particulares y minimizar la cantidad de recursos necesarios al no necesitar entrenarlo desde cero.

LLMs (Large Language Models): grandes modelos de lenguaje entrenados con grandes cantidades de datos, capaces de generar y comprender el lenguaje natural. Emplean arquitecturas profundas como Transformers y aprenden patrones para realizar tareas de razonamiento, conversación, clasificación, sentimiento y más.

LoRA Rank: parámetro clave del método LoRA que determina la dimensión de las matrices de bajo rango agregadas al modelo LLM durante el fine-tuning. Un rango (rank) menor reduce el número de parámetros que se entrenan y acelera el proceso, mientras que un mayor rank permite capturar patrones más complejos pero mayor uso de recursos.

NLP (Natural Language Processing): área de la inteligencia artificial enfocada en la comprensión, interpretación y generación de lenguaje humano por parte de computadoras. Incluye tareas como análisis de sentimiento, traducción automática, respuestas a preguntas y más.

QLoRA: Método de fine-tuning que combina LoRA con cuantización de 4 bits permitiendo entrenar LLMs al reducir significativamente el uso de memoria. La cuantización convierte los pesos originales en representaciones menores, manteniendo su desempeño y haciendo más eficiente el entrenamiento en hardware limitado.

Transformers: arquitectura de redes neuronales basado en mecanismos de atención que permite procesar secuencias de forma paralela. Ha sido un gran aporte a las tareas de NLP por su eficiencia y capacidad de procesamiento.

Introducción

El presente proyecto de grado tiene como objetivo el desarrollo de un sistema de chat basado en inteligencia artificial que permita optimizar el acceso y la consulta de información científica perteneciente a trabajos académicos, como trabajos de grados o proyectos de investigación, de la IUPB. Este trabajo nace como respuesta a la dificultad que enfrentan los estudiantes e investigadores de la institución al tener que realizar sus búsquedas en fuentes académicas dispersas en múltiples repositorios, situación que no solo implica una gran inversión en tiempo, sino que también limita la reutilización del conocimiento y la continuidad de procesos investigativos.

Aprovechando el auge de la inteligencia artificial y, en particular, las LLM, se propuso entrenar un modelo tipo chat conversacional capaz de comprender preguntas y peticiones de búsqueda y devolver información institucional de manera coherente y contextualizada. El proyecto se enmarca dentro de la categoría experimental tecnológica puesto que combina la exploración científica de nuevas tecnologías con la construcción de un prototipo funcional aplicable en entornos reales.

El desarrollo del trabajo se estructuró en 4 componentes principales: la estructuración de la base de datos, el entrenamiento del modelo mediante técnicas de fine-tuning, la evaluación del desempeño a través de pruebas adaptadas al español del Benchmark GLUE, y la implementación de una interfaz gráfica de interacción tipo chat que se alojó en el servicio gratuito de mensajería Telegram. De igual forma, en paralelo, se mantuvo un componente de revisión constante de trabajos emergentes en el área del procesamiento del lenguaje natural (NLP) y un componente de diseminación y divulgación de la información, garantizando la actualización teórica y la visibilidad de los aportes del proyecto.

Entre las limitaciones encontradas se destaca la dificultad para construir un dataset lo suficientemente amplio con información institucional y también la necesidad de adaptar las métricas de evaluación, dado que en la actualidad el Benchmark GLUE se encuentra en inglés y es prácticamente el único estándar de validación de pruebas para modelos LLM. Sin embargo, el proyecto logró consolidar un modelo funcional, demostrando el gran potencial que hay en las nuevas tecnologías basadas en LLM para fortalecer los procesos de investigación académica, siendo una oportunidad de pensar en futuras implementaciones dentro de la institución.

1. Planteamiento del problema

1.1 Descripción

En el ámbito académico, la producción de conocimiento se ve impulsada por el acceso eficiente a fuentes de información confiables y actualizadas. Sin embargo, la amplia cantidad de publicaciones científicas dispersas en múltiples bases de datos representa un desafío para investigadores y estudiantes, quienes deben invertir un tiempo considerable en la búsqueda, análisis y síntesis de información relevante, siendo lo anterior la problemática sobre la cual se basa éste proyecto.

De acuerdo con la Universidad Veracruzana de México (2020), se estima que en un semestre se le dedica al menos 4h diarias por 8 semanas, es decir 160 horas, a la búsqueda, lectura, investigación y revisión bibliográfica investigativa para la realización de una tesis lo cual implica una gran porción de tiempo que, incluso pensándolo en términos de créditos universitarios, hace referencia a lo que sería medio semestre académico de una clase de 4 créditos, incluyendo las horas de trabajo en clase y las horas de trabajo independiente.

Habiendo identificado la problemática y siendo conscientes del auge que representa actualmente el desarrollo de herramientas de inteligencia artificial, en especial los modelos LLM, estos ofrecen una solución innovadora. Estos modelos han demostrado su capacidad para procesar grandes volúmenes de texto, identificar patrones y generar conocimiento estructurado de manera eficiente. Integrar estos modelos con repositorios académicos confiables como Google Académico o bases institucionales internas permitirá no solo centralizar la información, sino también garantizar la calidad de las fuentes consultadas y mayor eficiencia del investigador.

1.2 Formulación

¿Es posible optimizar el acceso rápido, confiable y útil a la información científica generada en proyectos de investigación universitarios, ante la dispersión y el volumen creciente de publicaciones usando grandes modelos de lenguaje (LLMs) para fortalecer los nuevos trabajos académicos de los estudiantes e investigadores de la Institución Universitaria Pascual Bravo?

2. Justificación

Esta propuesta tecnológica tiene un impacto directo en la eficiencia del investigador pues al tener toda la información centralizada y tener un modelo que tenga la capacidad de realizar una devolución ante una petición de búsqueda, el tiempo de investigación se podrá reducir significativamente y además se podrá facilitar la continuidad de proyectos que, por falta de sistematización, podrían quedar inconclusos o con rango de mejora. En consecuencia, este proyecto es relevante tanto en términos tecnológicos como institucionales y prácticos, ya que facilita el acceso al conocimiento, optimiza el tiempo de investigación, potencia el desarrollo académico y propicia una posible herramienta de utilidad para la institución.

3. Objetivos

3.1 Objetivo general

Construir un Gran Modelo de Lenguaje Natural que integre y procese información proveniente de trabajos y fuentes académicas (tales como el repositorio institucional) para proporcionar apoyo y orientación a proyectos emergentes dentro de la Institución Universitaria Pascual Bravo.

3.2 Objetivos específicos

- Estructurar una base de datos de documentos técnicos sobre trabajos académicos de investigación provenientes de fuentes académicas para el entrenamiento de una LLM
- Entrenar una LLM para responder preguntas y sintetizar información basada en los documentos académicos recopilados.
- Evaluar la LLM entrenada usando el modelo de evaluación SpanishGLUE con el fin de medir su desempeño en tareas conversacionales.
- Implementar una interfaz de consulta amigable tipo chat para facilitar a los estudiantes e investigadores el acceso y orientación sobre la información académica en los proyectos emergentes.

4. Marco teórico

Los grandes modelos de lenguaje han demostrado un gran avance para la generación y comprensión de textos según Atkinson-Abutridy, J, 2023. Sin embargo, se enfrentan con retos como su alto costo computacional, la falta de actualización del conocimiento y la dificultad de encontrar datasets adecuados para el entrenamiento. A continuación se explorarán los conceptos teóricos necesarios para explicar su funcionamiento, así como la evolución, fortalezas, debilidades y clasificación de las LLM. Algunos de ellos incluyen estrategias aplicadas actualmente para mitigar dichos retos, tales como LoRA (Low-Rank Adaptation) para reducir la demanda de recursos y propuestas de hilos de trabajo como el de Generador-Juez que, a partir de PDFs, permite generar datos sintéticos ajustados a la necesidad propia.

4.1 Modelos de Deep Learning

El Deep Learning o aprendizaje profundo (LeCun et. al, 2015) es una subdisciplina del Machine Learning, su diferencia radica en que este emplea redes neuronales con gran número de capas, o redes neuronales profundas, para simular el funcionamiento del cerebro humano y por tanto su capacidad de entendimiento, toma de decisiones y reconocimiento de patrones (Sotelo, J. A. L., 2021). Desde la perspectiva biológica, según Cox & Dean, 2014, las redes neuronales se inspiran directamente en la estructura del sistema nervioso humano donde las redes de diferentes partes del cerebro están especializadas en unas funciones específicas. Esta analogía permite construir modelos computacionales en donde se destinen redes para funciones específicas, replicando parcialmente la forma en la que los humanos percibimos, procesamos y respondemos a estímulos. A diferencia de modelos convencionales con redes neuronales simples de una o dos capas, el Deep Learning hace uso de múltiples capas que van desde las tres en adelante, pero, en algunos casos, llegan hasta los cientos o miles para entrenar los modelos, lo que permite alcanzar niveles de precisión elevados.

Como lo menciona Pérez, H. R. A., estos modelos se caracterizan por su notable capacidad de trabajar con grandes volúmenes de datos no estructurados, lo que reduce en gran medida la intervención humana a la hora de extraer características. Esto es gracias a su arquitectura jerárquica que, se mencionó anteriormente, se distribuye por capas. Según Bobadilla, J., 2021, el

proceso inicia sobre una capa de entrada donde los datos comienzan su proceso de propagación hacia adelante, una vez ingresan, atraviesan un ciclo de ajuste y predicción hacia la siguiente capa; cada capa subsiguiente toma los datos de salida de la capa anterior como sus propios datos de entrada hasta llegar a la última capa que se denomina capa de salida, refinando continuamente la clasificación o predicción del modelo y elevando la predicción lo que resulta en la detección de patrones de alto nivel. Es por eso que el Deep Learning con las redes neuronales ha demostrado resultados notables en diversas áreas como el reconocimiento de imágenes, la traducción de textos y el procesamiento de lenguaje natural (Li, 2018; Li, 2022; Singh et. al, 2017).

Según Matich, D. J., 2001, el funcionamiento de una red neuronal profunda se asienta en dos procesos fundamentales que se desarrollan entre una capa de entrada donde ingresan los datos de entrada no estructurados y terminan en una capa de salida donde resulta la predicción o clasificación final, a éstas capas se les denomina capas visibles. El primer proceso en el funcionamiento de una red neuronal profunda es la propagación hacia adelante que ya se mencionó anteriormente, donde los datos viajan a través de capas subsiguientes para generar una predicción o clasificación. Luego está la retropropagación, el cual es un ajuste de los pesos y sesgos de la red mediante algoritmos de optimización como el gradiente descendiente con el fin de minimizar al máximo los errores en las predicciones o clasificaciones de cada capa. Éste ciclo iterativo permite el "aprendizaje" de la red neuronal y así mismo su continua mejora (Huang et. al, 2019).

Si bien en los modelos de Deep Learning se puede implementar el aprendizaje supervisado, uno de los factores diferenciadores más importantes es su capacidad para implementar el aprendizaje no supervisado pues esto permite que la red identifique de forma autónoma las características más relevantes de la información sin la necesidad de etiquetar previamente los datos de entrada, como explican Jara, F. A., & Lobato, D. H., 2018. Resultando en la optimización del proceso cuando estructurar y etiquetar la información resulta muy costoso o inviable. Otra ventaja destacable sobre los modelos de Deep Learning es su escalabilidad pues permite que en la medida en la que hay más información y más capacidad computacional, el modelo tiende a mejorar progresivamente lo cual lo destaca por sobre otros modelos que pueden encontrar un límite en su desempeño (Zohuri et. al, 2020, Chollet, 2027).

Finalmente, es importante destacar que el entrenamiento de modelos de Deep Learning

genera una alta demanda de recursos computacionales, especialmente por su vasta red de actividades. Es por ello que las GPU de alto nivel se vuelven unas aliadas esenciales en este tipo de desarrollo a quienes se les suman herramientas como frameworks especializados tales como TensorFlow (Abadi et. al, 2015), PyTorch (Paszke et. al, 2019) y JAX (Bradbury et. al, 2018) que ofrecen estructuras optimizadas para el desarrollo, entrenamiento y despliegue de modelos complejos.

4.1.1 Tipos de modelos de Deep Learning

Según Canle, E., 2023, los algoritmos de Deep Learning se clasifican con base a su forma de usar las redes neuronales para abordar los problemas o conjuntos de datos de la siguiente manera:

- Redes neuronales convolucionales (CNN)
- Redes neuronales recurrentes (RNN)
- Redes neuronales generativas adversarias (GAN)
- Modelos de difusión.
- Transformers.

En el presente proyecto solo se hablará de los modelos de transformers pues es el tipo de red neuronal que se usa para las LLM.

4.2 Transformers

Los Transformers, según Nasimba Tipan, A. F., 2023, son un tipo de arquitectura de red neuronal que hace uso de datos secuenciales, en especial texto, para realizar su procesamiento. A diferencia de las redes neuronales recurrentes que también se usan mucho en el campo del NLP, los Transformers no dependen de una secuencia temporal para procesar los datos, sino que hace uso de un mecanismo de “atención”, como lo expresa Ardila, H. J. F., 2024, que permite al modelo enfocarse en varias partes de la información de entrada al mismo tiempo encontrando de forma más eficiente conexiones y relaciones a largo plazo entre las diferentes palabras. Moya

Iratxeta, K., 2024, expresa que los transformers se basan en una estructura codificador-decodificador, **como se aprecia en la Figura 1**, donde el codificador transforma el texto en N cantidad de representaciones vectoriales y el decodificador se encarga de generar predicciones palabra por palabra. Una de sus principales ventajas es su capacidad de procesamiento en paralelo, reduciendo considerablemente el tiempo de entrenamiento en comparación a arquitecturas secuenciales; además, han demostrado ser altamente escalables y adaptables mediante diferentes técnicas de pre entrenamiento y Fine-tuning donde puede existir un modelo ya dado pero se puede volver a entrenar para mayor especificidad de la información. No obstante, también nos encontramos con algunos desafíos como el alto consumo de recursos computacionales y la necesidad de grandes cantidades de datos de calidad para entrenar modelos con buen desempeño.

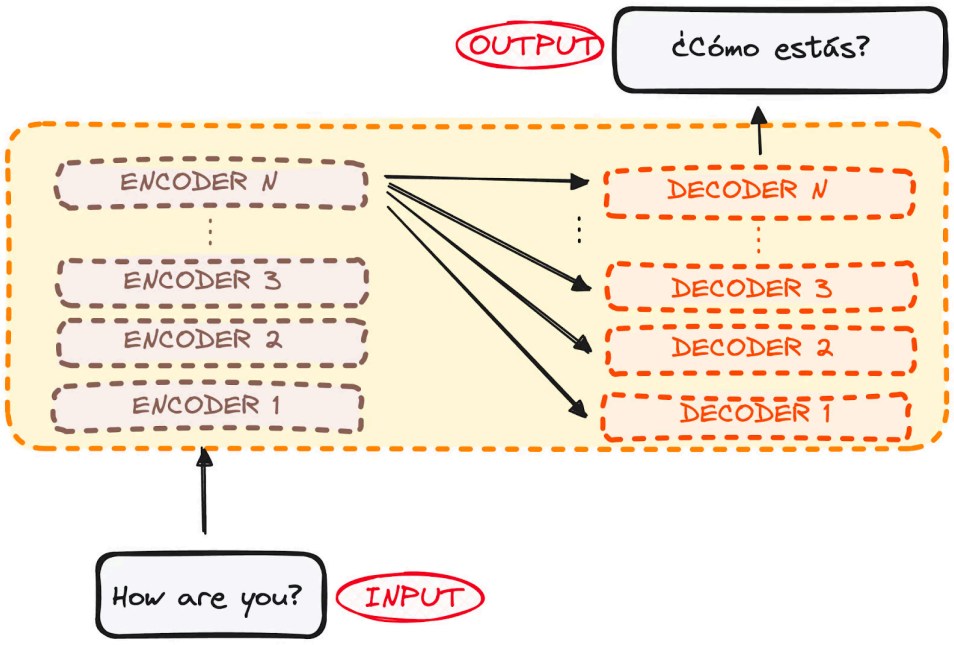

Figura 1 Esquema de funcionamiento de los Transformers

4.3 Grandes Modelos de Lenguaje Natural (LLM)

Los grandes modelos de lenguaje natural o también conocidos por sus siglas en inglés Large Language Models (LLM), son sistemas de inteligencia artificial con un tipo de estructura perteneciente al Deep Learning (Cárcamo Cabezas, N. M., 2024). Estos modelos están basados

en las redes neuronales del tipo transformers, las cuales fueron presentadas por primera vez en el 2017 por medio del artículo llamado *Attention is all you need* (Vaswani et. al, 2017) . Por medio de esta arquitectura, las LLMs se han entrenado con grandes volúmenes de datos mediante diferentes estructuras de entrenamiento donde, muchas veces, se aplica un entrenamiento autosupervisado, permitiendo al modelo aprender patrones del lenguaje y desarrollar habilidades generales para el procesamiento de lenguaje natural como la lectura y generación de texto, traducciones automáticas y respuestas a preguntas. Su aparición ha sido un cambio radical en el enfoque de los chats de inteligencia artificial en donde ya no se requieren modelos especializados en tareas concretas sino modelos de propósito general capaces de adaptarse a múltiples contextos (Ziegler et. al, 2019).

Como lo expone Gómez-Rodríguez, C., 2025, el funcionamiento de las LLM se basa en el aprendizaje a gran escala pues durante su entrenamiento, el modelo es expuesto a grandes cantidades de texto según para lo que se lo quiera entrenar, a partir de los cuales aprende a predecir las siguientes palabras. Esta tarea, basada en un principio "atencional", le permite construir, a nivel interno, representaciones complejas sobre el significado de las palabras, su contexto, relaciones gramaticales y patrones discursivos. Así, el modelo no "comprende" el lenguaje tal como un humano, pero si realiza conexiones e identifica patrones tan robustos que puede emular su comprensión.

Entre las ventajas de las LLMs está su gran capacidad de generar textos coherentes, contextuales y de alta calidad, simulando la interacción humana por chat y propiciando respuestas rápidas y claras antes un gran repertorio de preguntas y/o solicitudes, lo que ha potenciado asistentes conversacionales, sistemas de recomendación, traductores, etc. Gracias a técnicas como el aprendizaje por refuerzo con retroalimentación humana, la generación aumentada por recuperación (Retrieval-Augmented Generation o RAG) y el Fine-Tuning con datos específicos, estos modelos pueden adaptarse a tareas muy especializadas en diversos dominios como la medicina, la educación o incluso el derecho. Además, se destaca por su gran capacidad de realizar tareas repetitivas o complejas en segundos, lo cual lo convierte en una herramienta poderosa para el aumento de la productividad individual o en conjunto.

Aún con sus grandes avances, las LLMs también se enfrentan a retos importantes. Uno de sus principales desafíos es la demanda de recursos computacionales el cual puede llegar a ser una limitante significativa. Esta alta necesidad de cómputo tiende a llevar a la monopolización de la

creación de estas herramientas solo a manos de quienes tengan los recursos suficientes como empresas muy grandes, lo que genera preocupación sobre el monopolio tecnológico y el acceso desigual al conocimiento. Otro desafío importante es que, debido a su complejidad y número masivo de parámetros que lo componen, estos modelos suelen considerarse “cajas negras” cuya lógica interna es difícil de interpretar; esa falta de transparencia puede generar dificultad a la hora de rendir cuentas, especialmente cuando se tratan de contextos sensibles o con impacto social. Además, existen otros retos que actualmente no se reconocen como críticos pero que también hacen parte de la realidad de las LLM. Entre ellos está el sesgo de la información que podría acarrear riesgos éticos, morales y de desinformación; el riesgo siempre latente sobre la privacidad de los datos, y el impacto ambiental asociado al consumo alto de energía requerido para su entrenamiento.

Frente a todos estos desafíos han surgido diversas estrategias. Por un lado, está la creciente disponibilidad de modelos LLM de código abierto como Llama o DeepSeek, lo que descentraliza la información y permite que más investigadores y desarrolladores experimenten con estas nuevas tecnologías. De la mano con los modelos de código abierto, cada vez se desarrollan nuevos frameworks como LoRA que mitigan la demanda de recurso computacional por medio de la reducción de parámetros, permitiendo que, sin la necesidad de una inversión a gran escala en recursos técnicos, sea posible el entrenamiento de este tipo de modelos. Por otro lado, también se están desarrollando técnicas para auditar y mitigar sesgos como los modelos de evaluación, los datos curados, la validación humana y la evaluación de métricas éticas. Por último, algunos proyectos buscan reducir la huella de carbono de los LLMs mediante arquitecturas más eficientes o la reutilización de modelos; además, el establecimiento de marcos regulatorios y principios éticos y legales será clave para un desarrollo responsable en términos de privacidad de los datos.

4.4 Datos sintéticos

Los datos sintéticos, se reconocen como un recurso cada vez más implementado en los proyectos de ciencia de datos e inteligencia artificial, principalmente cuando se cuenta con una limitación en la recolección de datos específicos o en la estructuración de estos (Vallez, N et al,

2019). Según diversas fuentes como IBM, DataCamp y AWS, los datos sintéticos son aquellos datos no directamente provenientes de la recolección, observación o medición por parte de entidades reales, sino que han sido generados artificialmente por medio de algoritmos, simulaciones o modelos de inteligencia artificial, de modo que imitan ciertas propiedades, estructuras o semánticas de datos reales en los que se basan.

Los datos sintéticos, si bien se presentan como una alternativa bastante prometedora, tienen sus ventajas y desventajas. Dentro de sus ventajas más importantes, y la que es de gran utilidad para el presente trabajo, está el hecho de que supera la barrera de escasez de información de datos reales con estructuras adecuadas, como lo menciona Marshall Boehmwald, F., 2022, lo que permite cubrir casos infrecuentes; además, en otros contextos, al no contener identificadores de individuos reales, puede usarse de forma libre sin exponer datos personales, como lo explica Tamayo Urgilés, D. A., 2023; por otro lado, son una estrategia altamente económica en comparación a otras que puedan implicar mayor tiempo, planeación y recurso humano. Pero, no todo es tan positivo pues el mayor reto de los datos sintéticos es alcanzar la calidad de los datos reales ya que, en muchos casos, se pueden generar valores atípicos, datos sin sentido o información sesgada.

Existen varias formas de clasificar los datos sintéticos, pero para el presente proyecto se hablarán de dos clasificaciones útiles reconocidas por Villarroel González, G. J., 2023, y fuentes como IBM y DataCamp:

- Según el grado de síntesis:
    - Totales: Estos son aquellos conjuntos de datos generados totalmente desde cero sin que se incorpore información real.
    - Parciales: Conjuntos de datos que sustituyen partes de otro conjunto de datos de fuente real (como datos sensibles) por información generada artificialmente, de modo tal que se preserven algunos registros originales pero se reemplacen elementos como datos privados.
    - Híbridos: Este hace referencia a un conjunto de datos que combina registros reales con registros sintéticos para aumentar volumen y diversidad, dejando los registros reales como una especie de base.
- Según la estructura o modalidad de los datos:
    - Estructurados: Estos son datos organizados en formatos tabulares, bases de datos,

hojas de cálculo y/o con atributos definidos.

- No estructurados: Son datos sin un formato rígido como lo puede ser el texto libre, los documentos, las imágenes, los videos o audios.
- Semiestructurados: Hacen referencia a los datos donde el orden o la serie temporal tienen importancia como por ejemplo sensores que registran datos en el tiempo.

Para la generación de datos sintéticos se pueden implementar varias estrategias, como lo propone Rodríguez Reyes, R., 2018, desde técnicas estadísticas basadas en distribuciones y correlaciones matemáticas, hasta enfoques más avanzados como modelos Deep Learning, como por ejemplo los transformers. Por medio del funcionamiento de los transformers en el contexto de los datos sintéticos que implica que tengan una base de conocimiento y una orden (o prompt) para posteriormente generar ejemplos artificiales manteniendo las singularidades necesarias. Este esquema es comúnmente denominado Generador-Juez.

4.5 Hilo de trabajo “Generador-Juez”

En los sistemas de generación de datos sintéticos para tareas de pregunta-respuesta (QA) y modelos de lenguaje, una arquitectura emergente es el esquema Generador-Juez. Para este hilo de trabajo se cuenta con dos componentes principales que son:

- Un generador, que se encarga de recibir la información base y produce ejemplos sintéticos como, por ejemplo, pares de preguntas.
- Un juez, que tiene como tarea evaluar, filtrar o validar dichas instancias generadas, lo anterior para asegurar calidad, coherencia y utilidad para el entrenamiento.

Tal como lo describe el Dr. Eversberg, L (2024) en su artículo “How to Create a RAG Evaluation Dataset From Documents”, al implementar una pipeline de tipo Retrieval-Augmented Generation (RAG) que permite a los modelos de lenguaje acceder a conocimientos base externo, es decir, sin que haga parte de su entrenamiento, se logran obtener datasets de datos sintéticos con menor probabilidad de que estas LLMs alucinen o den información falsa. El objetivo es obtener un conjunto de datos con información verídica y coherente, bien estructurada y compuesto por contextos, preguntas y respuestas derivados de documentos no estructurados.

Este enfoque tiene la gran ventaja que, en comparación a otros métodos de generación de datos sintéticos, mejora la calidad de los datasets al introducir un mecanismo de validación (el juez) que filtra ruido o errores; pero además, se reconoce como un proceso que permite escalar la generación de datos y la diversificación de los ejemplos pues en muchos casos de aplicación de inteligencia artificial, la búsqueda de datasets en los formatos adecuados es una limitante constante. Pero, también se debe tener en cuenta que la calidad de los datos sintéticos depende tanto de los datos de entrada como de un buen manejo de prompts para las LLMs que se empleen pues si no se cuenta con estas dos condiciones, el resultado no será el esperado.

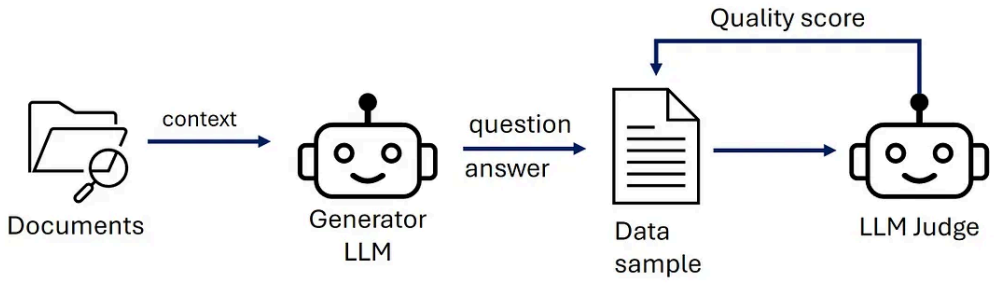

Figura 2 Flujo de trabajo para generar automáticamente mientras de datos de evaluación RAG a partir de documentos

Los hilos de trabajo Generador-Juez emplean la estrategia mostrada en la figura 2, la cual incluye los siguientes pasos:

a. ***Entrada documental:*** Se parte de documentos estrictamente en formato PDF que contienen información relevante para el caso del presente proyecto.
b. ***Generación:*** El Generador, una LLM de uso local y libre, recibe la información, la procesa, recibe un prompt y produce pares de pregunta-respuesta en un archivo JSON referentes a la información contenida en el Input.
c. ***Evaluación:*** El Juez, otra LLM de uso local y libre distinta a la del Generador, revisa los pares generados, recibe un prompt sobre lo que se espera como salida y entrega otro archivo JSON con una dictamen y una valoración en escala de números.

d. ***Consolidación:*** Aquellos pares de pregunta-respuesta que cumplan con la validación según criterios internos, se integran en un dataset final que será utilizado para el entrenamiento.

5. Metodología

5.1 Tipo de proyecto

Experimental tecnológico.

5.2 Método

Para la elaboración del presente proyecto se tuvo en cuenta 4 componentes de trabajo, como se muestra en la Figura 3, que se distribuyeron según su pertinencia en duración: 1. la estructuración de la base de datos, 2. el entrenamiento del modelo, 3. la evaluación del modelo y 4. la implementación de la interfaz.. A continuación se describen detalladamente cada uno de estos componentes.

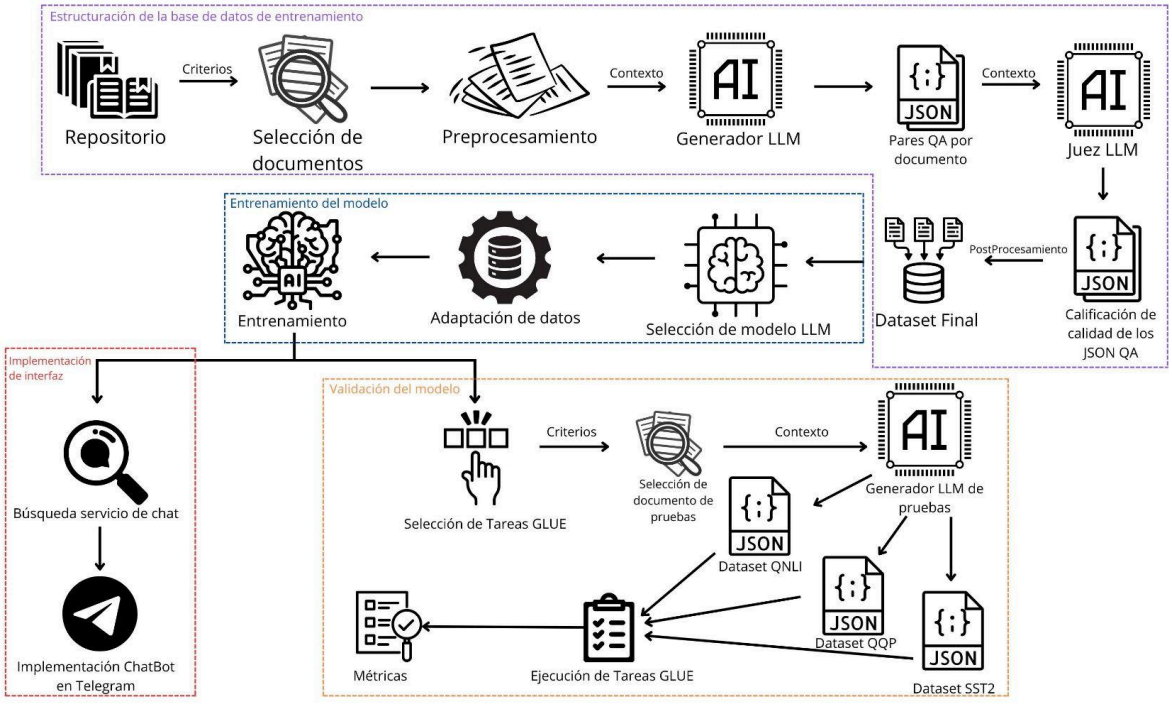

Figura 3 *Resumen gráfico de la implementación. Fuente: Elaboración propia.*

Estructuración de la base de datos de entrenamiento

Para la consecución de la base de datos se realizó un proceso sistemático de recolección,

selección, preprocesamiento y estructuración de la información que alimentó el modelo de lenguaje. Para lo anterior se llevó a cabo lo siguiente:

- **Recolección de información:** Se hizo la búsqueda de las fuentes como google Scholar, los repositorios de documentos de investigación de simposios y repositorios institucionales, para el proceso de recolección de información, siendo el repositorio universitario de la IUPB, el seleccionado por su pertinencia con el proyecto.
- **Definición de criterios y selección de documentos:** Una vez seleccionada la fuente de información, se definieron los criterios para la selección de documentos de la siguiente forma:
    - Los documentos académicos debían pertenecer a la tecnología en Desarrollo de Software o a la ingeniería de Software de la IUPB.
    - Se seleccionaron trabajos publicados posterior al año 2021.
    - De aquellos documentos duplicados, sólo se hizo uso de la versión más actualizada según su fecha.
    - Sólo se consideraron documentos de tipo PDF.
- **Filtro final:** Se hizo un filtro final en el que se eliminaron aquellos documentos que por alguna razón no cumplieron con los criterios requeridos pero de igual forma se guardaron en el repositorio.
- **Estructuración:** Se realizó el proceso de estructuración del dataset con LLMs de Ollama ya funcionales para generar datos sintéticos bajo el flujo de trabajo “Generador-Juez” en donde:
    - El generador, por medio de un prompt, hizo lectura de los diversos PDF, analizó la información y entregó un archivo JSON por cada PDF con la estructura Pregunta-Respuesta (QA)
    - El juez, también mediante un prompt, analizó los pares de pregunta-respuesta generados y evaluó si el generador realmente produjo el contenido solicitado en cuanto a estructura y características requeridas. A partir de esta comparación, el juez emitió nuevos JSON con una valoración de claridad, relevancia y veracidad, junto con una calificación final en una escala de 1 a 5 por cada archivo de pares de QA.
- **Eliminación de entradas de baja calidad semántica:** Todos aquellos pares de

pregunta-respuesta que tuviesen una calificación final menor a 3 con respecto a su calidad semántica y correspondencia entre ellas, se eliminaron de los archivos. Por último, los archivos han sido unificados en formato JSON para ser procesados por los modelos.

Entrenamiento del modelo

Para el entrenamiento del modelo se utilizó la LLM DeepSeek-R1-Distill-Qwen-7B (HuggingFace) pre entrenada de código abierto optimizada para diálogos en español. Aquí, se realizó un proceso de fine-tuning que hace referencia a una técnica de Deep Learning en la que un modelo pre entrenado se ajusta con más datos para tareas más específicas. Para cumplir con este objetivo se realizaron actividades tales como:

- **Selección del modelo base:** Se realizó una exploración alrededor de los diversos repositorios de LLMs de código abierto, como Hugging Face o GitHub. El objetivo fue identificar un modelo pre entrenado que sea multilingüe o especializado en español, que ofreciera una fácil implementación, tiempos de respuesta eficientes y respuestas relevantes.
- **Proceso de entrenamiento:** Se realizó el entrenamiento acompañado de herramientas de uso libre tales como Hugging Face Transformers, Pytorch y TensorFlow. Adicionalmente, con el propósito de simplificar el entrenamiento, buscando reducir los recursos computacionales y optimizar el uso de recursos, se implementaron técnicas de fine-tuning eficiente como LoRA.

Productos esperados:

- Modelo LLM entrenado.

Validación del modelo

Para validar el modelo entrenado se empleó una adaptación al español del Benchmark GLUE (General Language Understanding Evaluation o en español Evaluación General de la Comprensión del Lenguaje), con la intención de específicamente evaluar modelos en español mediante múltiples tareas entre las cuales se encuentran la inferencia de pregunta-respuesta, la similitud semántica, el análisis de sentimientos, etc. Este proceso de validación se rigió por las

siguientes actividades:

- **Selección de tareas relevantes:** Dado que GLUE se compone de múltiples tareas diseñadas para la evaluación de diferentes capacidades lingüísticas, se identificaron aquellas más pertinentes para los objetivos del proyecto.
- **Generación de dataset:** Se utilizó un documento académico distinto a los empleados para el entrenamiento, para generar datasets de datos sintéticos reutilizando el pipeline Generador-Juez de tal forma que cumplieran con la estructura adecuada para cada tarea. Para lo anterior se tuvo como referencia la estructura de los datasets propios del Benchmark GLUE.
- **Pruebas:** Se aplicaron las diversas tareas del modelo de evaluación GLUE en el modelo LLM ya entrenado.
- **Análisis de resultado:** Una vez realizadas las pruebas se midió el rendimiento mediante métricas como F1 (Entidad), exactitud (Accuracy) y MSE, para posteriormente analizarlas y determinar fortalezas, debilidades y posibles ajustes.

Productos esperados:

- Métricas de rendimiento.

Implementación de interfaz

Para la interacción con el modelo se implementó una interfaz tipo chat de uso libre que permitió al usuario final acceder de manera sencilla, amigable y práctica al modelo LLM. Para cumplir éste objetivo se llevaron a cabo las siguientes tareas:

- **Revisión de herramientas disponibles:** Se realizó una exploración de las principales plataformas disponibles para la implementación de chatbots, tales como Whatsapp, Telegram, Facebook Messenger, Discord y demás que ofrezcan entornos de desarrollo adecuados o APIs gratuitos.
- **Evaluación y selección:** Una vez identificadas las herramientas viables, se procedió a evaluar cada una de ellas en función de facilidad de uso, accesibilidad, costos y facilidad de implementación.
- **Implementación:** Se hizo la integración del bot con la opción seleccionada según los lineamientos técnicos y métodos de implementación que la herramienta requirió.

Productos propuestos:

- Interfaz de implementación del modelo.

5.3 Instrumentos de recolección de información

5.3.1 Fuentes primarias.

- Repositorio académico de la IUPB

5.3.2 Fuentes secundarias.

- Bases de datos disponibles en repositorios de datos libres.
- Artículos indexados de revistas científicas.
- Material de clase, visto en la Línea de profundización - Machine Learning.

6. Resultados

El proyecto produjo como resultado principal un modelo de conversación funcional especializado, entrenado y ajustado con precisión a preguntas relacionadas con trabajos de grado del área de desarrollo de software de la IUPB. Para lograrlo se construyó una arquitectura completa de generación y validación de datasets con datos sintéticos, necesarios para suplir la ausencia de conjuntos de datos con información académica en español y con la estructura indicada para el entrenamiento de la LLM seleccionada. Si bien la producción de datos sintéticos y la adaptación del Benchmark GLUE fueron componentes indispensables del proceso, estos constituyen a medios y no fines, que cumplieron su función de ser el soporte para obtener un modelo final funcional, coherente y evaluado.

6.1 Entrenamiento de la LLM

Como LLM base se seleccionó el modelo DeepSeek-R1-Distill-Qwen-7B, optimizado mediante QLoRA en 4 bits, lo que permitió ajustar la demanda de recursos con los recursos disponibles. Se entrenó durante 5 épocas y con batch size de 16, siendo esa una configuración más acertada para el entrenamiento teniendo en cuenta la cantidad de datos. Los hiperparámetros principales se presentan en la tabla 1:

Componente	Valor
Modelo Base	DeepSeek-R1-Distill-Qwen-7B
Técnica de ajuste fino con reducción de parámetros	LoRA (PEFT) + QLoRA 4bit
Épocas	5
Batch	16 (batch=1, grand_accum=16)
Learning Rate	2e-5
Weight Decay	0.01
Warmup Ratio	0.05
Precisión de cómputo	bfloat16
Gradient Checkpoint	Sí
Módulos LoRA	q_proj, k_proj, v_proj, o_proj

Tabla 1 *Configuración del proceso de entrenamiento.*

Con estas configuraciones el modelo resultante demostró coherencia, buena semántica y estabilidad en sus respuestas. Pero, cabe mencionar que los recursos del sistema fueron los siguientes:

Recurso	**Cantidad**
CPU	4 cores / 8 threads
GPU	RTX A4000 16GB
Disco	716GB
RAM	31GB

Tabla 2 *Recursos computacionales empleados*

6.2 Evaluación con tareas del Benchmark GLUE

Se aplicaron las tareas SST2 (análisis de sentimiento), QQP (paráfrasis) y QNLI

(inferencia causal y lógica), obteniendo métricas dentro de los rangos esperados para un modelo de tamaño medio entrenado con datos sintéticos. Los resultados se muestran en el gráfico

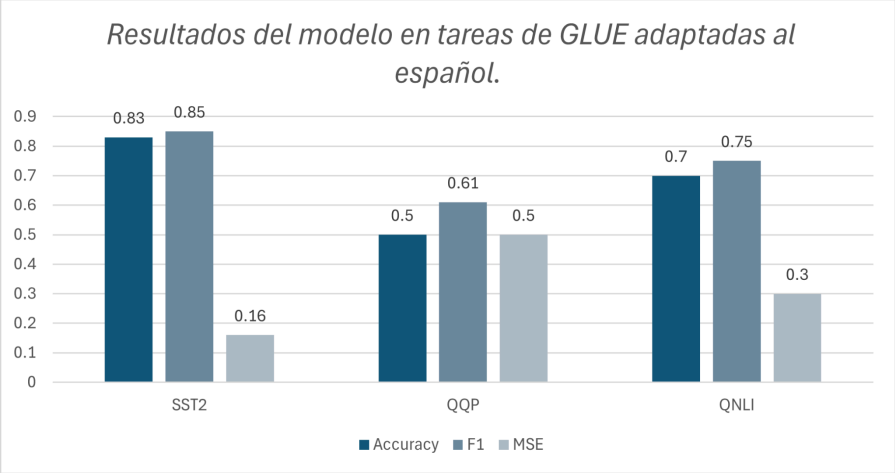

Figura 4 *Resultados del modelo en tareas de GLUE adaptadas al español.*

Estos valores indican que el modelo tiene mejor desempeño en inferencia y análisis de sentimiento que en tareas de paráfrasis lo cual, por medio de búsquedas, se puede atribuir al uso de los datos sintéticos y limitados en vez de datos reales y masivos.

6.3 Implementación del ChatBot

Finalmente, el modelo se integró como un ChatBot en el sistema de mensajería Telegram, donde se le permite interpretar consultas y generar respuestas explicativas basadas en la información con la que se entrenó. Éste producto se reconoce como la interfaz gráfica del sistema y se asemeja a otras investigaciones donde también hacen uso de este servicio por su facilidad de implementación. En la Figura 5 se pueden observar algunas interacciones realizadas con el modelo LLM en la interfaz gráfica, y en la Figura 6 se presenta el código QR para acceder al bot.

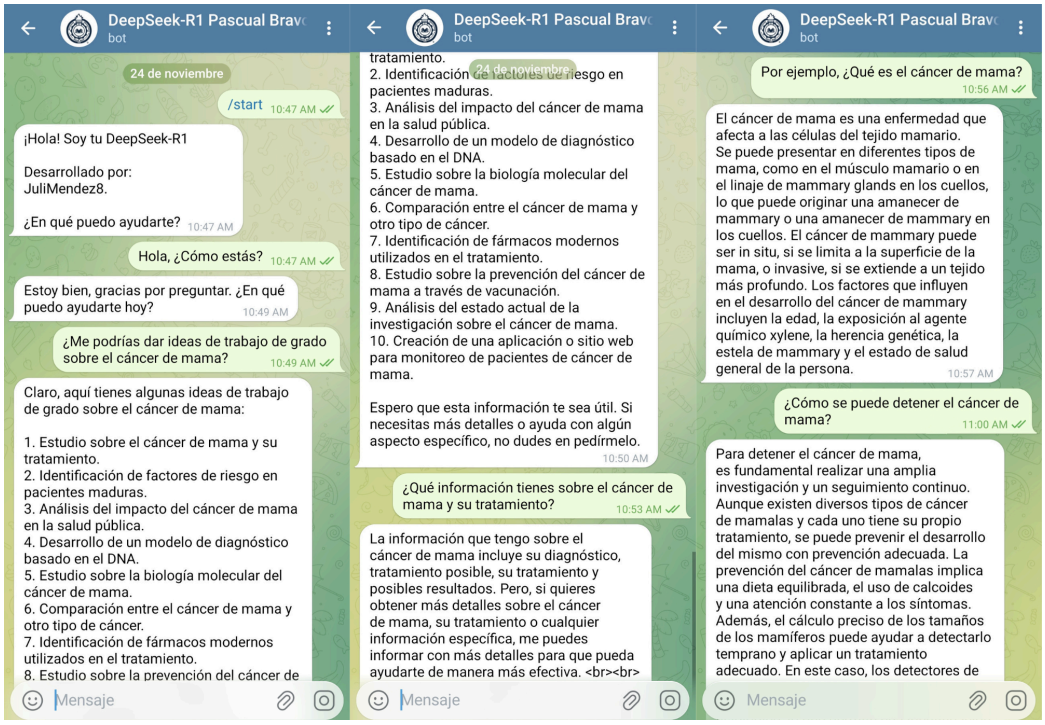

Figura 5 *Interacciones con el modelo en la interfaz gráfica.*

Figura 6 *Código QR del bot.*

7. Conclusiones

En conclusión, el desarrollo del presente trabajo permitió demostrar que es posible entrenar y ajustar un modelo de lenguaje de alcance medio, como lo es DeepSeek-R1-Distill-Qwen-7B, para responder preguntas especializadas en un contexto de trabajos de grados y trabajos de investigación académicos, incluso sin contar con datasets públicos de gran tamaño en español dedicados. La integración de un pipeline de generación y evaluación de datos sintéticos basado en el enfoque Generador-Juez fue un componente esencial y un completo acierto, permitiendo construir un dataset útil, verificable y alineado con las necesidades del proyecto.

Los resultados del entrenamiento evidenciaron que el proceso de fine-tuning mediante LoRA y QLoRA fue adecuada para las restricciones computacionales del proyecto, logrando un punto medio entre eficiencia y calidad, y permitiendo hacer uso de un modelo de tamaño medio que, una vez entrenado, demostró coherencia en sus respuestas, capacidad de razonamiento y pertinencia frente a preguntas derivadas de documentos académicos reales. Su evaluación mediante una adaptación al español del Benchmark GLUE permitió, además, identificar fortalezas y puntos por mejorar, lo cual aporta lineamientos para ajustes en proyectos o avances futuros.

Por otro lado, la integración del modelo como un ChatBot en el servicio de mensajería Telegram permitió validar su utilidad práctica, siendo un ejemplo claro de que los datos sintéticos y los modelos LLM ajustados pueden convertirse en una solución replicable para instituciones que requieran centralizar sus datos y hacerlos de fácil y rápido acceso. A nivel local, el proyecto aporta una metodología reproducible para la generación de conocimiento en español dentro de dominios específicos, y a nivel nacional, se alinea con las muchas investigaciones actuales sobre entrenamiento eficiente y uso de LLMs.

En conjunto, el presente trabajo evidencia que la propuesta de un chat especializado con información académica no solo es una posibilidad viable sino también una opción tangible para implementar dentro de la IUPB, además de ser escalable, sostenible y realmente útil para la comunidad universitaria.

8. Recomendaciones

A partir de los conocimientos adquiridos en el presente trabajo y considerando el gran potencial de crecimiento, se proponen las siguientes recomendaciones orientadas a fortalecer su continuidad y ampliar el impacto del sistema construido.

Inicialmente, se sugiere trabajar con un modelo de lenguaje de mayor tamaño puesto que, si bien el modelo utilizado demostró tener buenas capacidades para el propósito que se buscaba, el uso de arquitecturas más robustas permitirían mejorar la comprensión contextual, la capacidad de razonamiento del sistema e incluso la adición de nuevas capacidades como el aprendizaje por RAG lo que resultaría particularmente relevante si se busca escalar el proyecto hacia tareas más complejas, más información o incluso más público.

Además, se recomienda ampliar la generación de datos sintéticos incorporando trabajos de grado de todos los pregrados, posgrados y proyectos de investigación de la IUPB. Esta expansión permitiría construir un dataset más representativo de todo el conocimiento que se almacena en la institución y a su vez posibilitará extender el alcance a otras instituciones de educación superior, fortaleciendo la interoperabilidad y estandarización de datos académicos.

En tercer lugar, será imperativo incrementar los recursos computacionales pues, al trabajar con modelos de mayor tamaño y con volúmenes de datos significativamente mayores, se exige una capacidad de procesamiento, almacenamiento y aceleración de GPU mayor.

Finalmente, se recomienda que, en caso de escalar el proyecto, se considere el desarrollo de una interfaz gráfica propia, incluso se plantea la posibilidad de que esté articulada con la/las plataformas oficiales de la IUPB. A pesar de que la integración con Telegram funciona adecuadamente, una interfaz personalizada permitirá mejorar la usabilidad, adaptarla a los lineamientos institucionales y ofrecer funcionalidades específicas para estudiantes, docentes e investigadores.

Estas recomendaciones están orientadas al crecimiento del proyecto para que sirvan como una base para nuevas iniciativas académicas y tecnológicas que profundicen el uso de los modelos de lenguaje en contextos educativos.

9. Referencias bibliográficas

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., ... & Zheng, X. (2015, November). *TensorFlow: Large-scale machine learning on heterogeneous systems*.

Alahmid, M. (2024). *Evaluating Language Models on the GLUE Benchmark*. https://medium.com/@mustafaazzurri/evaluating-language-models-on-the-glue-Benchmark-cef507ea8c96

Ali, M. (2024). *Generación de Datos Sintéticos: Una guía práctica en Python.* DataCamp. Recuperado de https://www.datacamp.com/es/tutorial/synthetic-data-generation

Ardila, H. J. F. (2024). Procesamiento de Lenguaje Natural, los Transformers y los Bots Conversacionales. XIKUA Boletín Científico de la Escuela Superior de Tlahuelilpan, 12, 151-160.

Atkinson-Abutridy, J. (2023). Grandes modelos de lenguaje: Conceptos, técnicas y aplicaciones. Marcombo.

AWS. (s.f.). *¿Qué son los datos sintéticos?*. Recuperado de https://aws.amazon.com/es/what-is/synthetic-data/

Bobadilla, J. (2021). Machine learning y deep learning: usando Python, Scikit y Keras. Ediciones de la U.

Bradbury, J., Frostig, R., Hawkins, P., Johnson, M. J., Leary, C., Maclaurin, D., ... & Zhang, Q. (2018). *JAX: composable transformations of Python+ NumPy programs*.

Canle, E. (2023). *Tipos de Deep Learning: una guía completa.* Tokio School. Recuperado de https://www.tokioschool.com/noticias/tipos-deep-learning/

Cárcamo Cabezas, N. M. (2024). Usos de LLMS en la enseñanza universitaria: un análisis FODA.

Cox, D. D., & Dean, T. (2014). Neural networks and neuroscience-inspired computer vision. Current Biology, 24(18), R921-R929.

Chollet, F. (2017). The limitations of deep learning. Deep learning with Python.

Eversberg, L. (2024). *How to create a RAG Evaluation Dataset From Documents*. https://medium.com/data-science/how-to-create-a-rag-evaluation-dataset-from-documents-140daa3cbe71

Ferrer, J. (2024). *Cómo funcionan los transformadores: Una exploración detallada de la arquitectura de los transformadores.* Recuperado de https://www.datacamp.com/es/tutorial/how-transformers-work

Gómez-Rodríguez, C. (2025). Grandes modelos de lenguaje: de la predicción de palabras a la comprensión?. arXiv preprint arXiv:2502.18205.

Holdsworth, J & Scapicchio, M. (2024). *¿Qué es el deep learning?*. IBM. Recuperado de https://www.ibm.com/es-es/think/topics/deep-learning

Huang, K., Hussain, A., Wang, Q. F., & Zhang, R. (Eds.). (2019). *Deep learning: fundamentals, theory and applications.* Switzerland: Springer International Publishing.

IBM. (s.f.). *¿Qué son los datos sintéticos?*. Recuperado de https://www.ibm.com/es-es/think/topics/synthetic-data

Jara, F. A., & Lobato, D. H. (2018). APRENDIZAJE NO-SUPERVISADO CON MODELOS GENERATIVOS PROFUNDOS. Universidad Autónoma de Madrid.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). *Deep learning.* Nature, 521(7553), 436-444.

Li, H. (2018). *Deep learning for natural language processing: advantages and challenges.* National Science Review, 5(1), 24-26.

Li, Y. (2022, January). *Research and application of deep learning in image recognition.* In 2022 IEEE 2nd international conference on power, electronics and computer applications (ICPECA) (pp. 994-999). IEEE.

Marshall Boehmwald, F. (2022). Diseño de un modelo de generación de datos sintéticos para la aplicación de modelos de machine learning en proyectos interdisciplinarios asociados a salud.

Matich, D. J. (2001). Redes Neuronales: Conceptos básicos y aplicaciones. Universidad Tecnológica Nacional, México, 41, 12-16.

Moya Iratxeta, K. (2024). Predicción de la demanda de movilidad espacio-temporal del transporte público mediante transformers (Doctoral dissertation, ETSI_Informatica).

Nasimba Tipan, A. F. (2023). " Attention is all you need". Arquitectura Transformers: descripción y aplicaciones.

Nvidia. *Large Language Models Explained.* Recuperado de https://www.nvidia.com/en-us/glossary/large-language-models/

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., ... & Chintala, S. (2019). *Pytorch: An imperative style, high-performance deep learning library.* Advances in neural information processing systems, 32.

Pérez, H. R. A. El Impacto de la Inteligencia Artificial y el Machine Learning en la Valoración de Activos Financieros.

Ramirez, A. (2020). *¿Cuánto tiempo lleva hacer una tesis?*. Universidad Veracruzana. Recuperado de https://www.uv.mx/personal/albramirez/2020/10/10/tiempo-para-la-tesis/

Rodríguez Reyes, R. (2018). Herramienta para la creación de datos sintéticos en problemas de predicción con salidas múltiples integrado en MULAN (Bachelor's thesis, Universidad de las Ciencias Informáticas. Facultad 2).

Sotelo, J. A. L. (2021). Deep learning: teoría y aplicaciones. Alpha Editorial.

Singh, S. P., Kumar, A., Darbari, H., Singh, L., Rastogi, A., & Jain, S. (2017, July). Machine translation using deep learning: An overview. In 2017 international conference on computer, communications and electronics (comptelix) (pp. 162-167). IEEE.

Tamayo Urgilés, D. A. (2023). Construcción de un dataset de eventos de conducción utilizando modelos de generación de datos sintéticos mediante Generative Adversarial Networks (GAN) (Master's thesis, Quito: EPN, 2023.).

Vallez, N., Velasco Mata, A., Cotorro, J. J., & Deniz, O. (2019). ¿ Es posible entrenar modelos de aprendizaje profundo con datos sintéticos?.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Villarroel González, G. J. (2023). Data sintética privada, ejecución y evaluaciones de modelos.

Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., ... & Irving, G. (2019). *Fine-tuning language models from human preferences.* arXiv preprint arXiv:1909.08593.

Zohuri, B., & Moghaddam, M. (2020). *Deep learning limitations and flaws.* Mod. Approaches Mater. Sci, 2(3), 241-250.

Zuñiga, I. (2025). *¿Qué es el Deep Learning? Guía práctica con ejemplos*. Platzi. Recuperado de https://platzi.com/blog/deep-learning-guia-ejemplos/

10. Anexos

Anexo A. Dataset de entrenamiento Consolidado. Incluye un JSON consolidado para el entrenamiento que reúne los 37 datasets individuales de pares QA generados; estos JSON fueron previamente filtrados de acuerdo a su calificación de calidad, eliminando todas las entradas con calificación menor a 3. El resultado es un conjunto refinado y homogéneo de datos que garantiza un entrenamiento limpio y consistente.
Link: Anexo A

Anexo B. Datasets GLUE en español. Contiene los datasets adaptados al español con la estructura de tres tareas del Benchmark GLUE: QNLI (Question-Answer NLI), QQP (Quora Question Pairs) y SST-2 (Sentiment Analysis). Para cada una de las tareas se construyó un subconjunto de 60 muestras.
Link: Anexo B

Anexo C. Repositorio de GitHub. Enlace al repositorio oficial del proyecto en GitHub, contiene todo el código desarrollado durante el trabajo de grado y el último checkpoint resultante del entrenamiento. Allí están los scripts de procesamiento y generación de datos, la configuración del entrenamiento, el modelo ajustado, el script de pruebas y el script de implementación con la interfaz de chat. Este repositorio sirve como respaldo técnico.
Link: https://github.com/JuliMendez8/PascualResearchBot

Anexo D. Prueba en la interfaz de chat. Pantallazos resultantes de la fase de pruebas del modelo con la interfaz de chat. Las capturas de pantalla evidencian el funcionamiento del sistema, el comportamiento del modelo ante distintos tipos de consultas y la validación práctica de los resultados.
Link: Anexo D