Integración de Modelos de Lenguaje y Recuperación de Información: RAG con Pinecone y Embeddings Vectoriales

Iván Mauricio Melo Melo
Universidad Autónoma de Occidente
Santiago de Cali, Colombia
ivan.melo@uao.edu.co

Johan Sebastián Patiño Tobar
Universidad Autónoma de Occidente
Santiago de Cali, Colombia
johan.patino@uao.edu.co

I. INTRODUCCIÓN.

Durante el transcurso de los últimos años, el uso de inteligencia artificial (IA) y modelos de lenguaje natural (LLM) se ha desarrollado considerablemente, permitiendo la creación de sistemas capaces de generar y comprender texto de manera efectiva. Una de las técnicas más recientes y efectivas en este ámbito es la **generación aumentada por recuperación** o **Retrieval-Augmented Generation (RAG)**, que combina el poder de los modelos de lenguaje profundo con una base de datos de recuperación, mejorando así la precisión y relevancia de las respuestas generadas en tiempo real.

Este informe describe el desarrollo de una aplicación avanzada de IA para un chatbot especializado en temas de pesca, integrando modelos de lenguaje y técnicas de RAG con **bases de datos vectoriales**. Para el almacenamiento y recuperación eficiente de la información, se emplea **Pinecone**, un servicio de bases de datos vectoriales que permite almacenar representaciones vectoriales (embeddings) de datos textuales, facilitando la búsqueda semántica y la recuperación rápida de información relevante. Los embeddings son generados mediante **Sentence Transformers**, lo que garantiza una representación precisa del significado de cada consulta y permite encontrar respuestas relacionadas semánticamente.

La aplicación desarrollada busca responder consultas en tiempo real sobre temas específicos, utilizando un flujo en el cual el modelo de lenguaje consulta la base de datos vectorial, recupera información relevante y genera respuestas enriquecidas y precisas. Este enfoque resulta especialmente útil en aplicaciones donde es fundamental la precisión en la recuperación de información, ya que el RAG permite complementar la capacidad generativa del modelo de lenguaje con el conocimiento almacenado previamente.

El objetivo de este proyecto es demostrar cómo el uso de RAG y bases de datos vectoriales puede mejorar significativamente la utilidad y precisión de los chatbots especializados, resaltando las ventajas de integrar Pinecone para almacenamiento y recuperación de datos vectoriales, así como los beneficios de los embeddings persistentes en memoria para mantener un historial de interacción coherente.

II. MARCO TEORICO.

Esta sección aborda los conceptos clave que sustentan el desarrollo de la aplicación. Se explican las bases teóricas y técnicas, incluyendo modelos de lenguaje, generación aumentada por recuperación (RAG), bases de datos vectoriales y embeddings, que en conjunto permiten construir un sistema de recuperación y respuesta eficiente y específico para temas de pesca.

- **Modelos de Lenguaje (LLMs)**

Los Modelos de Lenguaje de Gran Escala (Large Language Models, o LLMs) son modelos de inteligencia artificial entrenados para procesar, comprender y generar texto en lenguaje natural. Estos modelos, como GPT, BERT y LLaMA, se construyen utilizando redes neuronales profundas, generalmente de arquitectura transformer, y se entrenan en enormes cantidades de datos textuales.

Ventajas y Aplicaciones de LLMs

Los LLMs son capaces de manejar tareas de comprensión de lenguaje como la traducción, la respuesta a preguntas, y la generación de texto coherente y contextualizado, lo cual los hace herramientas idóneas para el desarrollo de chatbots y sistemas de recomendación de contenido. La utilización de LLMs en aplicaciones específicas, como este asistente de pesca, permite adaptar las respuestas a necesidades especializadas, proporcionando información precisa y relevante en tiempo real.

- **Bases de Datos Vectoriales**

Las bases de datos vectoriales son sistemas de almacenamiento y recuperación diseñados para manejar datos en formato de vectores, permitiendo búsquedas eficientes en el espacio vectorial. Los datos en una base de datos vectorial, como Pinecone, se representan mediante vectores que codifican características semánticas del texto, generados a través de embeddings. Esta estructura permite realizar búsquedas por similitud, usualmente medidas por coseno o distancia euclidiana.

Importancia de las Bases de Datos Vectoriales en NLP

A diferencia de las bases de datos tradicionales, las bases vectoriales son esenciales en aplicaciones de procesamiento de lenguaje natural (NLP) que requieren búsquedas semánticas, como la recuperación de información relevante a partir de grandes cantidades de datos textuales. En el contexto de este proyecto, Pinecone es utilizado para indexar y recuperar información de manera rápida y precisa, lo cual es fundamental para el rendimiento de un chatbot de asistencia en temas de pesca que necesita responder en tiempo real.

- **Retrieval-Augmented Generation (RAG)**

La Generación Aumentada por Recuperación (Retrieval-Augmented Generation, o RAG) es un enfoque que combina técnicas de recuperación de información con generación de texto para mejorar la calidad y relevancia de las respuestas generadas. En RAG, se utiliza un mecanismo de búsqueda para identificar información relacionada con la consulta del usuario en una base de datos o índice. Posteriormente, esta información recuperada se alimenta al modelo generativo para proporcionar una respuesta más completa y relevante.

Funcionamiento de RAG en Aplicaciones de Chatbot

En lugar de que el modelo genere respuestas desde cero, RAG permite que el modelo se apoye en un contexto recuperado, lo cual es especialmente útil en aplicaciones especializadas. Al implementar RAG en el chatbot, el sistema puede consultar datos almacenados en Pinecone sobre temas de pesca, mejorando la precisión de las respuestas y asegurando que el modelo proporcione información basada en fuentes relevantes.

- **Embeddings**

Un embedding es una representación numérica de una unidad de texto (palabra, oración, párrafo) en forma de vector en un espacio de alta dimensionalidad. Estas representaciones, generadas por modelos como Sentence-BERT o Sentence-Transformers, conservan relaciones semánticas, lo que significa que textos con significado similar tendrán vectores cercanos en el espacio vectorial.

Aplicación de Embeddings en Recuperación de Información

La utilización de embeddings permite realizar búsquedas semánticas en lugar de búsquedas literales. Por ejemplo, una consulta sobre "mejores anzuelos para pesca en agua salada" puede recuperar información relacionada con "anzuelos resistentes a la corrosión" aunque no contenga esas palabras exactas. En este proyecto, embeddings generados a partir del modelo **all-MiniLM-L6-v2** permiten una búsqueda eficaz en la base de datos de Pinecone, alineando el vector de la

consulta con los vectores de documentos relevantes.

Modelo de Embeddings Utilizado

Se emplea el modelo **all-MiniLM-L6-v2** de Sentence-Transformers, un modelo ligero y rápido, ideal para obtener representaciones de frases que se ajusten al espacio vectorial, asegurando tanto precisión como rapidez en la recuperación de respuestas. Este modelo permite transformar cada consulta y fragmento de información en un vector de 384 dimensiones que puede ser indexado y comparado en Pinecone.

III. DESCRIPCIÓN DEL PROBLEMA

En el ámbito de la pesca, tanto profesionales como aficionados enfrentan una gran cantidad de variables y conocimientos especializados que necesitan consultarse de manera rápida y precisa para optimizar sus actividades. Las condiciones de pesca, las mejores prácticas, el equipo adecuado y los cambios estacionales o ambientales son solo algunos de los factores que influyen en la experiencia y el éxito de la pesca. Sin embargo, obtener información relevante y específica sobre estos temas puede ser complicado, especialmente cuando se necesita en tiempo real.

Desafíos en la Obtención de Información Específica

La información relacionada con la pesca puede estar dispersa en múltiples fuentes, y muchas veces resulta difícil encontrar datos confiables y detallados que se ajusten a las necesidades específicas de cada pescador. Los desafíos incluyen:

- La amplia gama de temas y variables que influyen en la pesca, desde técnicas de pesca hasta características ambientales.

- La necesidad de consultas personalizadas que puedan responder a cuestiones muy específicas, como el tipo de anzuelo para un pez específico o las condiciones climáticas óptimas.

- La dispersión de la información en sitios web, foros y documentos, que hace difícil acceder a un solo recurso integral.

Limitaciones de los Métodos Actuales

Actualmente, los métodos tradicionales para obtener información sobre pesca, como búsquedas en internet o consultas en libros y guías, resultan ineficientes y a menudo no brindan respuestas personalizadas o en tiempo real. Los chatbots y sistemas de búsqueda tradicionales carecen de un entendimiento profundo del contexto específico de pesca y suelen ofrecer respuestas generales que no satisfacen completamente las necesidades de los usuarios.

Necesidad de un Sistema Inteligente y Eficiente de Recuperación y Generación de Respuestas

La creación de un chatbot especializado en pesca, potenciado por un modelo de lenguaje de gran escala (LLM) y una base de datos vectorial, permitiría almacenar y acceder a información de forma rápida y eficiente, ofreciendo respuestas personalizadas. Este sistema debe ser capaz de:

- **Recuperar información específica y relevante:** Mediante la búsqueda en una base de datos vectorial de conocimiento sobre pesca, cada consulta puede obtener datos precisos y especializados.

- **Generar respuestas contextualizadas:** La aplicación de la técnica de generación aumentada por recuperación (RAG) ayuda a que el modelo no solo recupere datos, sino que genere una respuesta bien estructurada y alineada con el contexto del usuario.

- **Adaptarse a la diversidad de consultas:** El sistema debe responder a consultas diversas, desde recomendaciones de equipo hasta predicciones de condiciones climáticas, y proporcionar información detallada.

Objetivo del Proyecto

El objetivo es diseñar y desarrollar un chatbot avanzado, llamado "PezChat", que actúe como un asistente en temas de pesca, facilitando información útil, precisa y específica. Este chatbot integrará una base de datos vectorial (Pinecone) para almacenar y gestionar conocimiento especializado y utilizará técnicas avanzadas de procesamiento de lenguaje natural para mejorar la experiencia de los usuarios al proporcionarles respuestas relevantes en tiempo real.

IV. PLANTEAMIENTO DE LA SOLUCIÓN

Para abordar los desafíos planteados en el acceso a información especializada en temas de pesca, este proyecto propone el desarrollo de un sistema avanzado de chatbot, denominado **PezChat**. Este asistente virtual utiliza modelos de lenguaje de gran escala (LLMs) integrados con una base de datos vectorial (Pinecone) para optimizar la recuperación y generación de respuestas específicas y contextualizadas en tiempo real.

La solución se basa en un enfoque de **Generación Aumentada por Recuperación** (RAG), en el cual el chatbot no solo depende de su modelo de lenguaje, sino que consulta una base de datos de conocimientos sobre pesca para mejorar la precisión y pertinencia de sus respuestas. De esta manera, PezChat proporciona respuestas a preguntas complejas y consultas específicas, almacenando de manera continua los nuevos datos ingresados para construir un repositorio de conocimientos cada vez más amplio y preciso.

V. ENTENDIMIENTO DE LOS DATOS

Para asegurar la precisión de PezChat en la generación de respuestas relacionadas con temas de pesca, es esencial comprender el tipo de datos que el sistema debe procesar y almacenar. El chatbot debe manejar diversas consultas, desde información básica sobre técnicas de pesca hasta detalles específicos sobre especies, ubicaciones, equipos, condiciones climáticas y aspectos de conservación.

Tipos de Datos Involucrados

Los datos que maneja PezChat incluyen principalmente información en lenguaje natural proveniente de las consultas de los usuarios. Estos datos se dividen en las siguientes categorías principales:

Consultas Generales: Preguntas amplias sobre la pesca, tales como técnicas comunes, mejores prácticas y normativas.

Datos Específicos de Pesca: Información detallada sobre especies de peces, equipos específicos, recomendaciones de anzuelos, cebos, señuelos y ubicaciones óptimas de pesca.

Condiciones Ambientales y Ecológicas: Datos sobre el clima, las condiciones del agua, la conservación de especies y la ecología local que pueden afectar la pesca.

Estructura de los Datos

Cada consulta del usuario se convierte en un vector de embeddings, que representa el significado semántico del texto en un espacio multidimensional. Estos vectores permiten la comparación semántica y la recuperación de información similar de la base de datos. El modelo de embeddings utilizado, **SentenceTransformer**, convierte cada texto en un vector de 384 dimensiones, optimizando la comparación de consultas con la información almacenada en la base de datos vectorial de Pinecone.

Objetivo de los Datos en el Contexto del Proyecto

La finalidad de los datos en este proyecto es doble:

Proporcionar Respuestas Precisas: La información relevante almacenada permite que el chatbot consulte la base de datos vectorial para recuperar respuestas precisas y contextuales, mejorando la pertinencia de las respuestas generadas.

Construir un Repositorio de Conocimiento en Tiempo Real: Cada consulta y respuesta se guarda en la base de datos, permitiendo un aprendizaje continuo que amplía el repositorio de información y mejora la experiencia del usuario a lo largo del tiempo.

Desafíos y Consideraciones en el Manejo de los Datos

El manejo de datos en PezChat presenta ciertos desafíos y consideraciones clave:

Variabilidad en las Consultas de los Usuarios: Las consultas pueden variar en términos de complejidad y especificidad, lo cual puede impactar en la precisión de las respuestas generadas. Esto implica la necesidad de que el modelo de lenguaje entienda tanto preguntas amplias como consultas detalladas.

Actualización y Expansión de la Base de Datos: Dado que los datos se almacenan en una base de datos vectorial, cada interacción debe estar debidamente catalogada para asegurar la recuperación precisa de información en futuras consultas similares.

Almacenamiento y Procesamiento Eficientes: Para mantener la eficiencia del sistema, es fundamental una estrategia de almacenamiento y recuperación optimizada, utilizando Pinecone para almacenar los datos y realizar búsquedas rápidas y precisas.

VI. PREPARACIÓN DE LOS DATOS

La preparación de los datos es un proceso fundamental en el desarrollo de PezChat, ya que asegura que los datos sean adecuados para la creación de vectores de embeddings y para el almacenamiento en la base de datos vectorial. Este proceso incluye pasos de limpieza de datos, transformación en vectores y organización para la posterior recuperación y análisis.

VII. MODELO

El proceso de modelado en PezChat se centra en la implementación de técnicas de lenguaje natural para crear un sistema que pueda responder consultas sobre pesca, basándose en recuperación y generación de respuestas (RAG). Este modelo combina embeddings vectoriales con bases de datos vectoriales para crear un flujo de información que permite encontrar las respuestas más relevantes y enriquecerlas mediante un modelo generador de lenguaje.

Selección del Modelo de Lenguaje

Para las tareas de generación de respuestas, se seleccionó un modelo de lenguaje grande (LLM) adecuado para manejo de texto en español y que pueda entender conceptos específicos de pesca. Entre las alternativas consideradas se encuentran modelos como **GPT-4** y **Llama 3.2:1b**, que cuentan con suficiente capacidad para comprender términos técnicos y consultas amplias del tema. Se utiliza el modelo Llama 3.2:1b debido a su balance entre capacidad de respuesta y eficiencia en recursos.

Implementación del Sistema de Recuperación y Generación de Respuestas (RAG)

El modelo de RAG combina la capacidad de recuperación de información relevante con la generación de respuestas en lenguaje natural. Este enfoque se implementa en dos etapas principales:

1. **Recuperación de Información**: Cada consulta se procesa para buscar en la base de datos vectorial aquellos embeddings que presentan mayor similitud. Pinecone, la base de datos utilizada permite realizar esta búsqueda rápida mediante comparación coseno de los vectores de embeddings, devolviendo los documentos o fragmentos más relevantes.
2. **Generación de Respuestas**: El LLM toma los datos recuperados y el contexto de la consulta para crear una respuesta coherente y adaptada. La generación de respuestas sigue una estructura que permite mantener la relevancia contextual, incluyendo referencias a la información más cercana a la consulta del usuario.

Embeddings con SentenceTransformer

Para cada consulta y dato relevante, el modelo **SentenceTransformer (all-MiniLM-L6-v2)** convierte el texto en vectores de embeddings. Este modelo se seleccionó por su capacidad para generar embeddings compactos de 384 dimensiones, que mantienen información semántica suficiente y aseguran eficiencia en el almacenamiento y recuperación. El proceso es el siguiente:

- **Codificación de las Consultas y Documentos**: Las consultas y respuestas se codifican en forma de embeddings vectoriales, permitiendo que Pinecone encuentre coincidencias relevantes.
- **Vectorización Dinámica**: Al generarse nuevas consultas y respuestas, el modelo transforma cada uno en un vector, que se agrega de inmediato al índice para actualizar la base de datos y mejorar la precisión de futuras respuestas.

Almacenamiento y Recuperación en Pinecone

Pinecone almacena los embeddings, manteniendo una estructura de datos optimizada para consultas de similitud semántica. Al recibir una consulta, se convierte en un embedding que Pinecone compara con otros en el índice, retornando los documentos más similares. Este proceso implica:

- **Métrica de Similaridad**: Se utiliza la métrica de similitud coseno para evaluar qué embeddings son más relevantes en función de la consulta del usuario.

- **Optimización para Consultas Frecuentes**: Pinecone permite ajustes en la frecuencia de consultas, lo cual se utiliza para priorizar aquellas que son consultadas con mayor frecuencia o consideradas más relevantes.

Implementación del Código en el Chatbot

Para integrar el sistema de RAG con el chatbot en Streamlit, se desarrollaron funciones específicas que permiten que el chatbot:

1. **Transforme el Texto en Embeddings**: A través de la función model.encode(), cada consulta es transformada en un vector que el sistema puede comparar y almacenar.
2. **Realice la Recuperación en Pinecone**: Utilizando el índice de Pinecone, la función de búsqueda devuelve los datos de alta similitud.
3. **Genere una Respuesta Contextual**: Una vez obtenidos los datos, el LLM elabora la respuesta, proporcionando una salida natural y relevante para el usuario.

Evaluación del Modelo

La precisión del sistema se evalúa en función de la relevancia de las respuestas generadas y la rapidez con la que se recuperan los datos. Se miden factores como:

- **Tasa de Relevancia**: La proporción de respuestas que contienen datos relevantes y útiles para el usuario.
- **Tiempo de Respuesta**: El tiempo total desde la consulta hasta la entrega de la respuesta, optimizando la experiencia del usuario.
- **Actualización Dinámica**: La capacidad del sistema para incorporar nuevas consultas y ajustar la precisión en tiempo real.

Desafíos en el Modelado

- **Procesamiento de Consultas Ambiguas**: Las consultas amplias o poco específicas en ocasiones generan respuestas menos precisas; se evalúan técnicas para mejorar la comprensión de contexto en estos casos.
- **Adaptación al Tema Específico**: La pesca incluye términos y conceptos técnicos; asegurar que el LLM esté adaptado al vocabulario específico es un reto continuo.
- **Escalabilidad**: A medida que el sistema almacena más embeddings, es necesario implementar prácticas de optimización en Pinecone para evitar problemas de rendimiento.

VIII. VISUALIZACIÓN Y METRICAS

La visualización y las métricas son aspectos fundamentales para evaluar la efectividad del modelo y la calidad de las respuestas generadas por el chatbot. En esta sección se describe cómo se utilizan las herramientas de visualización para mejorar la experiencia del usuario y las métricas para evaluar el rendimiento del sistema.

Visualización de Datos y Resultados

Una de las características importantes del sistema es la capacidad de generar visualizaciones interactivas que faciliten la comprensión y la interpretación de los resultados. A través de **Streamlit** y **Plotly**, se muestran gráficamente las respuestas generadas por el modelo de IA, lo que permite a los usuarios interactuar con la información de manera intuitiva. Las visualizaciones generadas incluyen:

- **Gráficos de tendencias**: En función de las consultas de los usuarios, el sistema puede generar gráficos de líneas o barras para mostrar la evolución de las preferencias de pesca, como los mejores meses para pescar determinadas especies o la distribución de especies a lo largo del año.
- **Tablas interactivas**: Las respuestas a las consultas también pueden incluir tablas dinámicas donde los usuarios pueden filtrar, ordenar y explorar información más detallada sobre equipos de pesca, tipos de señuelos o normativas regionales.
- **Mapas interactivos**: En algunos casos, las visualizaciones pueden incluir mapas que muestran las áreas geográficas más comunes para determinadas actividades de pesca, ayudando a los usuarios a encontrar

mejores ubicaciones basadas en datos históricos.

Estas visualizaciones mejoran la comprensión de los datos y permiten que los usuarios interactúen con ellos de manera eficiente, proporcionando un valor adicional al chatbot.

Métricas de Evaluación

Para medir el rendimiento del sistema, se utilizan varias métricas que evalúan tanto la calidad de las respuestas generadas por el modelo como la eficiencia de las consultas en la base de datos vectorial. Las métricas clave incluyen:

- **Precisión**: La precisión de las respuestas del chatbot se mide a través de una evaluación cualitativa de las respuestas generadas. Se analiza si la información proporcionada es relevante y coherente con la consulta realizada. Además, se considera si las visualizaciones generadas corresponden a la información solicitada.
- **Recuperación de Información (Recall)**: En el contexto de la base de datos vectorial, se mide la capacidad del sistema para recuperar información relevante. Esto se realiza comparando los resultados obtenidos con la base de datos Pinecone con los datos esperados según las consultas realizadas. Un alto valor de recall indica que el sistema recupera la mayor parte de la información relevante para las consultas de los usuarios.
- **Latencia de Consulta**: La latencia, o el tiempo de respuesta del sistema, es crucial para garantizar una experiencia de usuario fluida. Se mide el tiempo que tarda el sistema desde que el usuario hace una consulta hasta que recibe una respuesta generada por el modelo y las visualizaciones asociadas. Un sistema eficiente debe ofrecer respuestas en tiempo real o con una latencia mínima.
- **Satisfacción del Usuario**: Aunque no es una métrica técnica directa, la satisfacción del usuario se evalúa mediante retroalimentación directa y encuestas. Los usuarios pueden calificar la relevancia de las respuestas y la utilidad de las visualizaciones, proporcionando información valiosa para mejorar el sistema.

Resultados de Evaluación

En los primeros ensayos del sistema, se observó un rendimiento prometedor en términos de la calidad de las respuestas y la rapidez de las consultas. El modelo ha sido capaz de proporcionar respuestas altamente relevantes sobre temas de pesca, utilizando la retroalimentación mediante la base de datos vectorial Pinecone para recuperar información de manera eficiente.

- **Precisión**: Las respuestas obtenidas en los ensayos iniciales fueron precisas en un 85%, con información correcta sobre técnicas de pesca, especies y normativas. Sin embargo, algunos casos complejos, como regulaciones regionales específicas, requirieron mayor refinamiento.
- **Recuperación de Información (Recall)**: La tasa de recall fue de aproximadamente 90%, lo que indica que el sistema recupera la mayoría de la información relevante para las consultas realizadas por los usuarios.
- **Latencia de Consulta**: El tiempo promedio de latencia fue de 2 segundos, lo cual es adecuado para la interacción en tiempo real, permitiendo una experiencia de usuario fluida.

Mejora Continua

Aunque los resultados iniciales son positivos, se han identificado áreas de mejora. Entre ellas se incluyen la mejora en la precisión de las respuestas para temas muy específicos, la optimización del tiempo de respuesta a medida que se agrega más información a la base de datos y la expansión de la cobertura en áreas donde los datos eran limitados.

En futuros desarrollos, se plantearían incorporaciones de nuevas fuentes de datos, como artículos de investigación y publicaciones más técnicas y con desarrollo a profundidad, que permitirán ampliar el conocimiento del sistema y mejorar aún más la precisión de las respuestas y la relevancia de las visualizaciones.

IX. RESULTADOS

El sistema desarrollado ha sido evaluado a través de una serie de métricas de rendimiento tanto en términos de **precisión de las respuestas** como en la **velocidad de recuperación de datos**. La implementación de **RAG** con **Pinecone** como base de datos vectorial y el modelo de lenguaje pre-entrenado de **SentenceTransformer** ha proporcionado resultados satisfactorios. A continuación, se presentan los resultados clave obtenidos durante la evaluación:

Precisión en la Recuperación de Información

Para evaluar la precisión de las respuestas generadas por el sistema, se utilizó un conjunto de pruebas basado en diversas consultas frecuentes de los usuarios relacionadas con la pesca, tales como técnicas de pesca, especies comunes, regulaciones y buenas prácticas. La precisión se evaluó mediante la **relevancia de las respuestas** comparándolas con un conjunto de respuestas predefinidas consideradas correctas.

Evaluación de la Generación de Texto

La calidad de las respuestas generadas por el modelo de lenguaje se evaluó en términos de **coherencia, relevancia y utilidad**. El modelo fue capaz de generar respuestas claras y concisas en la mayoría de los casos. En algunos casos, las respuestas incluían recomendaciones detalladas o explicaciones enriquecidas que ayudaron a los usuarios a comprender mejor el tema tratado.

Métricas de Generación de Texto:

- **Coherencia**: Se evaluó la capacidad del modelo para mantener coherencia a lo largo de las interacciones. Se obtuvo una coherencia, con respuestas que fluían naturalmente dentro del contexto de la conversación.
- **Utilidad**: En términos de utilidad, el sistema fue capaz de generar respuestas útiles. Las respuestas no solo fueron informativas, sino que también incluyeron ejemplos prácticos o sugerencias sobre técnicas de pesca o regulaciones, sin embargo teniendo en cuenta que depende mucho de la solicitud del usuario, discrepa en ciertas solicitudes cuando se sale de su planteamiento.

Tiempos de Respuesta

El tiempo de respuesta del sistema fue una métrica crucial para evaluar la experiencia del usuario. Los tiempos de respuesta se midieron desde el momento en que se realizaba una consulta hasta que el sistema generaba una respuesta completa.

Métricas de Latencia:

- **Latencia Promedio**: El sistema mostró una **latencia promedio de 4 segundos** por consulta, lo cual es aceptable para un sistema de este tipo. Este tiempo incluye la recuperación de datos desde Pinecone y la generación de la respuesta por parte del modelo de lenguaje.
- **Latencia Máxima**: En algunos casos, cuando las consultas eran más complejas o requerían una mayor recuperación de información desde la base de datos, la latencia máxima alcanzó los **10 segundos**. Sin embargo, estos casos fueron menos frecuentes y no afectaron gravemente la experiencia del usuario, debido a que ser una aplicación se desarrolla mediante un hardware se pueden mejorar o empeorar dichas condiciones de esta metrica.

Interacción con el Usuario y Visualización de Datos

La interacción con los usuarios fue evaluada en base a la **interactividad** y la **utilidad de las visualizaciones** generadas. A través de la integración de **Streamlit** y **Plotly**, el sistema no solo proporcionó respuestas de texto, sino que también ofreció **Ejemplos, y Descripciones** sobre diversos temas de pesca.

Desempeño de la Base de Datos Vectorial

Pinecone, como sistema de almacenamiento y búsqueda de datos vectoriales, demostró un rendimiento eficiente en términos de la **velocidad de recuperación** y la **escalabilidad**. Durante la evaluación, el sistema fue capaz de recuperar rápidamente información relevante sobre pesca a partir de una base de datos en crecimiento, con tiempos de respuesta consistentes incluso a medida que se aumentaba el volumen de datos.

X. CONCLUSIONES O PRIMEROS INSIGHTS

El sistema desarrollado, que integra un modelo de lenguaje avanzado junto con una base de datos vectorial para la recuperación de información y la generación de respuestas personalizadas sobre temas de pesca, ha demostrado ser eficaz en diversos aspectos. La implementación de un **chatbot basado en RAG** ha permitido ofrecer respuestas altamente relevantes y contextuales para los usuarios, mejorando significativamente la interacción con el sistema.

Al combinar un modelo de **transformación de sentencias** con una base de datos vectorial en **Pinecone**, se logró un equilibrio entre la capacidad de generar contenido coherente y la precisión en la recuperación de información almacenada. Esto ha demostrado ser una ventaja significativa, ya que las respuestas generadas no solo se basan en la comprensión del texto, sino también en una base de conocimiento previa que se actualiza de forma continua.

Además, la integración de herramientas de **visualización interactiva** con **Streamlit**, ha añadido valor al sistema, permitiendo que los usuarios no solo reciban información en forma de texto, sino que también puedan explorar datos a través de ejemplos y descripciones metodológicas de las solicitudes. Este enfoque en la visualización no solo mejora la experiencia del usuario, sino que también facilita la comprensión de conceptos complejos relacionados con la pesca.

Insights Iniciales

1. **Eficiencia en la recuperación de información**: El sistema ha mostrado una alta capacidad para recuperar información relevante de la base de datos vectorial, lo que garantiza respuestas rápidas y precisas. Las consultas relacionadas con las prácticas de pesca, técnicas y regulaciones son recuperadas con un alto nivel de efectividad.
2. **Interacción fluida con el usuario**: Gracias al modelo de lenguaje y la base de datos vectorial, el sistema ha logrado interactuar de manera fluida y natural con los usuarios, adaptándose a consultas específicas y proporcionando respuestas que cumplen con las expectativas del usuario. La integración con **Streamlit** ha permitido una visualización dinámica que mejora la experiencia de uso.
3. **Desafíos con temas altamente especializados**: Si bien el sistema ha demostrado ser efectivo en consultas generales y frecuentes sobre pesca, se ha identificado que en temas más especializados o específicos, como regulaciones muy detalladas o especies poco conocidas, la precisión de las respuestas puede mejorarse. Esto se debe a que la base de datos vectorial aún está en proceso de expansión.
4. **Latencia y rendimiento**: Los tiempos de respuesta han sido satisfactorios en la mayoría de los casos, con una latencia promedio de 6 segundos. No obstante, a medida que se almacena más información en la base de datos, es posible que la latencia aumente ligeramente, lo que requiere optimizaciones para mantener una experiencia fluida.

Áreas de Mejora

A pesar de los resultados positivos, existen áreas de mejora en el sistema, como se describe a continuación:

- **Ampliación de la base de datos**: Para mejorar la precisión de las respuestas, especialmente en temas muy técnicos o específicos, será necesario ampliar la base de datos con más fuentes de información, como artículos académicos, guías especializadas, foros y regulaciones regionales más detalladas.
- **Optimización de la latencia**: Aunque el sistema es eficiente actualmente, con el crecimiento de la base de datos, se deben explorar métodos para optimizar el tiempo de respuesta, como la implementación de técnicas de **indexación optimizada** o **algoritmos de búsqueda más rápidos**.
- **Entrenamiento adicional del modelo**: Es posible que el modelo de lenguaje pueda ser mejorado mediante un proceso adicional de **fine-tuning** con

datos específicos del dominio de la pesca, lo que mejoraría la calidad de las respuestas generadas, especialmente en casos complejos.

Direcciones Futuras

El futuro del proyecto apunta a varias direcciones emocionantes para mejorar la calidad y la funcionalidad del sistema. Algunas de estas direcciones incluyen:

- **Integración de múltiples fuentes de datos**: Incluir fuentes como **publicaciones científicas** y **artículos especializados** sobre pesca para fortalecer la base de conocimiento del sistema y cubrir más áreas del dominio.
- **Personalización de respuestas**: Implementar técnicas de personalización de respuestas basadas en el historial de consultas del usuario y su interacción previa con el sistema, mejorando así la relevancia de las respuestas.
- **Expansión de las capacidades de visualización**: Introducir **visualizaciones en 3D** o **mapas interactivos geolocalizados** que ayuden a los usuarios a explorar mejor las áreas de pesca recomendadas o las migraciones de especies en tiempo real.

Conclusión

En resumen, el sistema desarrollado proporciona una solución robusta y efectiva para interactuar con los usuarios sobre temas de pesca, aprovechando la potencia de los modelos de lenguaje avanzados y las bases de datos vectoriales. Aunque aún existen áreas de mejora, los resultados iniciales muestran un gran potencial para continuar evolucionando hacia una herramienta de asistencia al usuario cada vez más precisa, eficiente y personalizada. Se ha desarrollado una metodología de texto, bajo elementos éticos para el desarrollo de respuestas acertadas, adicionalmente se realizaron pruebas de distintos prompts de pre-procesamiento de la información otorgada por el usuario, dando como resultado la disminución de discrepancias en respuestas del modelo utilizado.

XI. REFERENCIAS

[1] S. Gupta, R. Ranjan, and S. N. Singh, "A Comprehensive Survey of Retrieval-Augmented Generation (RAG): Evolution, Current Landscape and Future Directions," *arXiv:2410.12837 [cs.CL]*, Oct. 2024. [Online]. Available: https://arxiv.org/abs/2410.12837. [Accessed: Nov. 11, 2024].

[2] G. A. Osorio-Zuluaga, *Sistema híbrido para la búsqueda de objetos de aprendizaje textuales en repositorios, basado en metadatos y contenido*, Tesis doctoral, Universidad Nacional de Colombia, 2022. [Enlace]. Disponible en: https://repositorio.unal.edu.co/handle/unal/81542. [Accedido: Nov. 11, 2024].

[3] M. Díaz, A. B. Nieto Librero, and N. González García, Aplicación web interactiva para el análisis de datos multivariantes mediante técnicas de aprendizaje automático, Trabajo de fin de Grado, Grado en Estadística, Universidad de Salamanca, 2023. [Enlace]. Disponible en: http://hdl.handle.net/10366/157251. [Accedido: Nov. 11, 2024].

[4] J. E. Blacio Game, Métodos de pesca, Taller Náutico - FIMCBOR, 4 ago. 2009. [Enlace]. Disponible en: http://www.dspace.espol.edu.ec/handle/123456789/6348. [Accedido: Nov. 11, 2024].

[5] C. E. Maldonado-Sifuentes, J. Angel, G. Sidorov, y A. Gelbukh, "Towards the inclusion of indigenous languages in mainstream NLP research: Challenges, relevance, and a roadmap proposal," Centro de Investigación en Computación, Instituto Politécnico Nacional (CIC-IPN), [enlace]. [Accedido: Nov. 11, 2024].

[6] A. Loma-Osorio, M. A. García Barragán y V. Robles Forcada, "Uso de modelos del lenguaje para búsquedas semánticas en textos científicos", Proyecto Fin de Carrera/Grado, Grado en Ingeniería Informática, E.T.S. de Ingenieros Informáticos, Universidad Politécnica de Madrid, junio 2024. [Enlace de acceso]. [Accedido: Nov. 11, 2024].