Get Talent – Semana 5

Challenge Final

Objetivos

- Utilizar de manera efectiva los conceptos y técnicas aprendidos a lo largo del curso.
- Desarrollar una solución innovadora utilizando utilizando una arquitectura RAG.
- Mostrar competencia en la creación y optimización de prompts para modelos de lenguaje.
- Combinar diversas tecnologías estudiadas para abordar y solucionar problemas del mundo real.

Criterios de Evaluación

- Originalidad y enfoque innovador de la solución presentada.
- Pertinencia y la novedad de la solución en relación con el problema abordado.
- Relevancia y originalidad de la solución propuesta.
- Calidad de la presentación y la claridad con la que se expone el proyecto.
- Efectividad y eficiencia en la resolución de los problemas planteados.
- Nivel de profesionalismo demostrado en la ejecución y presentación del proyecto.
- Cumplimiento los requisitos mínimos establecidos para este proyecto.
- Habilidad para aprender y aplicar nuevos conocimientos durante el desarrollo del proyecto.
- Capacidad para introducir ideas creativas e innovadoras en la solución propuesta.

Instrucciones

1. Definir el problema y los requisitos
    a. Identificar claramente el problema que se desea resolver. Pueden pensar un problema real que les sea de interés.
2. Recopilación y preparación de datos
    a. Recopilar un texto relevante que será utilizado.
    b. Limpiar y preprocesar el texto de ser necesario.
3. Implementación del Módulo de Recuperación
    a. Seleccionar una base de datos vectorial adecuada para almacenar y recuperar información.

Santa Rosa 320 Piso 5 A
Teléfono +54(351)5693075
Email: info@piconsulting.com.ar
www.piconsulting.com.ar

   b. Indexar los datos en la base de datos vectorial.
   c. Desarrollar algoritmos de búsqueda eficiente para recuperar información relevante basada en consultas.
4. Desarrollo del Módulo de Augmentación
   a. Diseñar un mecanismo para enriquecer la información recuperada con datos adicionales o contexto relevante.
5. Construcción del Módulo de Generación
   a. Seleccionar y entrenar un modelo de lenguaje adecuado para la generación de texto.
   b. Integrar el modelo de lenguaje con el módulo de recuperación y augmentación.
   c. Desarrollar prompts efectivos para guiar la generación de texto de alta calidad.
6. Integración y pruebas
   a. Integrar los módulos de recuperación, augmentación y generación de texto en una arquitectura RAG cohesiva.
   b. Realizar pruebas exhaustivas para asegurar que el sistema funciona correctamente y cumple con los requisitos establecidos.
   c. Ajustar y optimizar los componentes según sea necesario para mejorar el rendimiento y la precisión.
7. Documentación y presentación
   a. Documentar detalladamente el proceso de desarrollo, incluyendo decisiones técnicas y justificaciones.
   b. Preparar una presentación que explique la arquitectura RAG, su implementación y los resultados obtenidos.
8. Despliegue
   a. Deployar la solución en un entorno adecuado para su uso.

Requisitos mínimos obligatorios

Generales

1. Funcionamiento completo: la solución debe ser capaz de procesar una consulta del usuario y proporcionar una respuesta adecuada de principio a fin.
2. Implementación mediante API: la solución debe estar disponible para consultas a través de API.
3. Pruebas de ejemplo: deben estar testeadas al menos con 3 preguntas de ejemplo relacionadas con el tema elegido.

Documentación

1. Extensión del documento: el documento (o la suma de varios documentos) debe tener más de 100000 (cien mil) caracteres.
2. Base de datos vectorial: la base de datos vectorial debe ser persistente y estar precargada antes de la presentación.

Calidad de respuesta

1. Pertinencia de las respuestas: la solución solo debe responder a temas pertinentes a los documentos utilizados y no debe abordar otros tópicos.
2. Restricciones de formato:
    a. No se deben usar emojis en las respuestas.
    b. Las respuestas deben ser siempre en español, independientemente del idioma de la pregunta.
    c. La misma pregunta debe generar la misma respuesta en cada interacción.
3. Personalidad de la respuesta: libre, basada en el tema elegido.

API

1. Endpoint de consulta: debe haber un endpoint para realizar consultas y recibir respuestas en lenguaje natural.
2. Documentación de la API: Incluir documentación detallada de la API en el entregable final.

Sugerencias optativas de innovación

1. ***Incluir reranking:*** implementar técnicas de reranking para mejorar la relevancia de las respuestas recuperadas.
2. ***Incluir Groundedness para validación:*** utilizar dicha metrica para validar la precisión y relevancia de las respuestas generadas.
3. ***Guardar conversaciones:*** implementar un sistema para almacenar y gestionar el historial de conversaciones.
4. ***Crear una interfaz gráfica:*** desarrollar una interfaz gráfica de usuario (GUI) para facilitar la interacción con la API.

Santa Rosa 320 Piso 5 A
Teléfono +54(351)5693075
Email: info@piconsulting.com.ar
www.piconsulting.com.ar

5. ***Orquestador impulsado por LLM***: Integrar un orquestador basado en LLM que dirija el flujo de la conversación según las preguntas del usuario (por ejemplo, si el usuario dice "hola", el LLM responde directamente sin pasar por RAG).
6. ***Usar LangChain:*** emplear LangChain para gestionar diferentes etapas de la solución, optimizando el flujo de trabajo.
7. ***Explorar nuevas herramientas:*** utilizar herramientas adicionales no vistas en clase, para enriquecer la solución.
8. ***Mejorar la información augmentada con NLP:*** implementar técnicas avanzadas de procesamiento de lenguaje natural (NLP) para mejorar la calidad y precisión de la información augmentada.
9. ***Utilizar la metadata de la base de datos vectorial:*** aprovechar la metadata disponible en la base de datos vectorial para mejorar la recuperación y contextualización de la información.
10. ***Incorporar técnicas avanzadas de IA:*** explorar e implementar técnicas avanzadas de inteligencia artificial que no se hayan cubierto en el curso para aportar valor añadido a la solución.

Formato de Entrega

1. Fecha y hora de entrega:
    - El challenge deberá ser entregado el viernes 20/12 antes de las 14 hs.
2. Repositorio del código fuente:
    - Compartir el código fuente del proyecto en un repositorio, como GitHub. Deberá incluir todos los archivos necesarios para poder reproducir el proyecto en otra computadora.
3. Documentación detallada:
    - Incluir documentación detallada de la solución en formato PDF o Markdown.
4. Presentación oral:
    - Cada participante deberá presentar su desarrollo de forma oral en el horario asignado, con una duración máxima de 30 minutos.
5. Demostración en vivo:
    - La solución debe estar operativa para mostrar en vivo cómo responde a las consultas del usuario.