Actividad: Creación de un Sistema RAG Sencillo con IA Generativa

Contexto del Caso

Una pequeña empresa necesita implementar un sistema que permita a sus empleados consultar información interna de manera eficiente. El equipo de desarrollo ha decidido utilizar un enfoque RAG (Retrieval-Augmented Generation) apoyado en modelos generativos, para responder preguntas sobre documentos internos (por ejemplo, manuales, políticas o reportes).

Instrucciones para el Alumno

1. **Lee el caso**: Eres parte del equipo encargado de prototipar una solución RAG que permita responder preguntas sobre un conjunto de documentos internos (puedes usar archivos de texto o PDFs sencillos).
2. **Objetivo**: En 40 minutos, deberás crear un prototipo funcional que:
    - Permita cargar o indexar al menos 2 documentos.
    - Reciba una pregunta del usuario.
    - Utilice un modelo generativo (por ejemplo, vía Ollama o la API de OpenAI) para responder la pregunta, apoyándose en la información recuperada de los documentos.
3. **Herramientas sugeridas**: Python, LangChain, ChromaDB o similares (puedes usar ejemplos y plantillas de la documentación oficial).
4. **Entrega**: Al final, comparte el código fuente y una breve explicación de cómo lo implementaste.

Sugerencia de pasos para el alumno

- Usa técnicas de prompt engineering para definir la arquitectura del sistema RAG.
- Instala las librerías necesarias.
- Indexa los documentos usando una base de datos vectorial (como ChromaDB).
- Implementa una función que reciba una pregunta, recupere los fragmentos más relevantes y los pase al modelo generativo.
- Prueba tu sistema con al menos dos preguntas distintas.

Cuestionario de opción múltiple

1. **¿Cuál es la principal ventaja de un sistema RAG frente a un modelo generativo puro?**
    a. Menor consumo de memoria
    b. Respuestas más precisas y fundamentadas en datos
    c. Mayor velocidad de respuesta
    d. Menor costo de implementación
2. **¿Qué componente se encarga de buscar los fragmentos relevantes en un sistema RAG?**
    a. El modelo generativo
    b. El indexador vectorial
    c. El frontend
    d. El compilador

3. **¿Cuál es una limitación importante de la IA generativa en el contexto de generación de código?**
    a. No puede generar código en Python
    b. Puede inventar información o cometer errores de lógica
    c. Solo funciona con prompts en inglés
    d. No puede integrarse con bases de datos
4. **¿Qué es Ollama en el contexto de IA generativa?**
    a. Un editor de texto
    b. Un modelo de base de datos
    c. Una plataforma para ejecutar modelos generativos localmente
    d. Un sistema operativo
5. **¿Por qué es importante revisar el código generado por IA antes de usarlo en producción?**
    a. Porque siempre es incorrecto
    b. Para asegurarse de que cumple con los estándares y es seguro
    c. Porque la IA no puede escribir código
    d. Para ahorrar tiempo en pruebas

Pregunta de reflexión

¿Cómo crees que la integración de sistemas RAG y modelos generativos puede transformar la forma en que los desarrolladores acceden y utilizan la información en su trabajo diario? Explica con tus palabras y da un ejemplo.