Apuntes (simulados) — Embeddings, Vector DB y RAG

Apuntes Profe · 2026-02-17

Estos apuntes son un documento de ejemplo para practicar ingesta (ingest) de PDFs y consultas semánticas con ChromaDB usando embeddings generados por Ollama. El objetivo es tener un PDF con contenido realista: definiciones, pasos y ejemplos.

1. Embeddings (representación vectorial)

Un embedding es un vector numérico que representa el significado de un texto. Textos con significado parecido tienden a generar vectores cercanos en el espacio. En esta práctica se usa un modelo de embeddings de Ollama (por ejemplo: nomic-embed-text).

2. ChromaDB (base de datos vectorial)

ChromaDB almacena documentos junto con sus embeddings y metadatos. Permite consultar por similitud: das un texto, lo conviertes a embedding y recuperas los documentos más cercanos (top-k). Los metadatos sirven para filtrar resultados (por ejemplo, autor, source o tags).

3. RAG (Retrieval-Augmented Generation)

RAG combina recuperación (retrieval) + generación (generation). Primero se recuperan fragmentos relevantes desde la base vectorial; después, un modelo de chat (ej: llama3.1) genera una respuesta usando ese contexto. Si el contexto no contiene la información, el asistente debería reconocer la limitación.

Apuntes (simulados) — Ejemplos prácticos (CLI)

Autor: Profe · 2026-02-17

4. Flujo típico del proyecto

Paso A: ingestar documentos (json/txt/md/pdf) con chunking. Paso B: consultar por similitud para obtener contexto. Paso C: ejecutar RAG: construir prompt con contexto y pedir respuesta al modelo de chat.

5. Ejemplos de comandos

```
python main.py init
python main.py ingest --path data/apuntes.pdf --autor "Profe" --tags apuntes,tema1
python main.py query --q "vector database" --k 5
python main.py rag --q "¿Qué es RAG?" --where "{\"source\":\"apuntes.pdf\"}" --k 5
python main.py rag --q "Explica embeddings" --autor "Profe" --k 3
```

Nota sobre tags: si se guardan como CSV (por ejemplo: "apuntes,tema1"), el filtro por tag puede hacerse con post-filtrado en cliente. El filtro por autor se puede hacer con where (coincidencia exacta).