Manual Técnico: Agente de IA para RAG en n8n

Juan David Torres Avila

06/06/2025

1. Resumen General

El flujo "RAG 2" implementa un agente de IA con capacidades de Recuperación Aumentada por Generación (RAG), utilizando:

- **n8n** para la orquestación del flujo.
- **Qdrant** como vector store para almacenamiento y recuperación semántica.
- **Ollama** para embeddings y modelos de lenguaje.
- Documentos planos como fuente de conocimiento.

2. Componentes del Flujo

A continuación, se describen los nodos involucrados, su tipo, función, y conexiones relevantes:

2.1. Ingesta y Vectorización de Documentos

- **Schedule Trigger**: Activa el flujo en intervalos programados.
- **Read/Write Files from Disk**:
  - Ruta del archivo: `D:\Descargas\Codigo 2\Prueba-datos\inteligencia_artificial.txt`
- **Embeddings Ollama**: Convierte texto a vectores mediante el modelo `nomic-embed-text`.
- **Recursive Character Text Splitter**: Fragmenta texto en trozos de 800 caracteres con superposición de 100.
- **Default Data Loader**: Carga fragmentos como documentos para su procesamiento posterior.
- **Qdrant Vector Store**:
  - Colección: `Prueba-datos`
  - Operación: Inserción de vectores

2.2. 2. Recepción de Preguntas vía Webhook

- **When chat message received**:
  - Activa webhook al recibir un mensaje.
  - Webhook ID: generado automáticamente por n8n.
- **Edit Fields (Set)**:
  - Extrae y asigna variables:

```
chatInput = $json?.chatInput || $json.body.chatInput
sessionId = $json?.sessionId || $json.body.sessionId
```

2.3. 3. Agente IA con RAG

- **AI Agent**: Nodo central que une componentes:
  - LLM (Ollama Chat)
  - Memoria (Simple Memory)
  - Herramientas (Vector Store Tool)
- **Ollama Chat Model**:
  - Modelo conversacional para la respuesta final.
- **Simple Memory**:
  - Guarda contexto de conversación.
  - Clave fija: `fafcfe91178e4ccd850c21cb8b5d27d4`
- **Qdrant Vector Store1**:
  - Colección: `ia_txt`
  - Uso: Recuperación semántica.
- **Embeddings Ollama1**:
  - Convierte la consulta del usuario en vector.
- **Ollama Model**:
  - Motor de generación de texto.
- **Answer questions with a vector store**:
  - Interfaz para búsqueda en Qdrant.
  - Conectado como herramienta al Agente.

3. Conexiones Clave del Flujo

- Lectura de archivo → Vectorización → Qdrant
- Mensaje vía Webhook → Extracción de campos → Agente IA
- Agente IA combina:
  - Consulta a Qdrant con embeddings
  - LLM Ollama
  - Memoria de conversación

4. Datos Técnicos Importantes

- **ChunkSize**: 800 caracteres
- **Overlap**: 100 caracteres
- **Modelo de Embeddings**: `nomic-embed-text`
- **LLMs**: Ollama con configuración por defecto
- **Cargas a Qdrant**:
  - `Prueba-datos`: carga inicial del documento
  - `ia_txt`: utilizada en tiempo de consulta

5. Endpoint para Integración

Webhook URL: generado automáticamente por n8n

Formato de entrada JSON:

```
{
  "chatInput": "  Qu   es la inteligencia artificial?",
  "sessionId": "usuario123"
}
```

6. Consideraciones Finales

- Asegúrese que Ollama esté ejecutando el modelo requerido localmente.
- Las colecciones deben existir previamente en Qdrant.
- Este flujo puede adaptarse fácilmente a múltiples documentos y colecciones.