Spring AI – Documento Otimizado para RAG

Este documento foi estruturado especificamente para uso em sistemas de Retrieval Augmented Generation (RAG). O conteúdo está organizado em seções curtas, semanticamente coesas e com linguagem objetiva, facilitando a divisão em chunks e a recuperação vetorial.

Spring AI – Definição

Spring AI é um projeto do ecossistema Spring que fornece abstrações para integração de modelos de Inteligência Artificial Generativa em aplicações Java. Ele reduz o acoplamento com provedores específicos de LLMs e segue os princípios de configuração e extensibilidade do Spring Framework.

O foco do Spring AI é produtividade, padronização e facilidade de manutenção em aplicações corporativas.

Large Language Models (LLMs)

LLMs são modelos estatísticos treinados para prever a próxima palavra de uma sequência textual. Eles utilizam grandes volumes de dados para aprender padrões linguísticos e semânticos.

Esses modelos são a base da IA Generativa, permitindo geração de texto, classificação, resumo e resposta a perguntas.

ChatClient no Spring AI

O ChatClient é a principal abstração do Spring AI para comunicação com LLMs. Ele encapsula mensagens do sistema, do usuário e do assistente.

Também permite configurar parâmetros de inferência como temperatura, top-p e penalidades, influenciando diretamente o estilo e a previsibilidade das respostas.

Embeddings e Representação Vetorial

Embeddings são representações numéricas de textos em um espaço vetorial multidimensional. Eles capturam significado semântico, permitindo comparar textos por similaridade.

No Spring AI, embeddings são usados principalmente para indexação e recuperação de documentos.

VectorStore

VectorStore é o componente responsável por armazenar embeddings e realizar buscas por similaridade. Ele é essencial para arquiteturas RAG.

O Spring AI suporta múltiplas implementações de VectorStore, como Qdrant, PGVector e Redis.

Ingestão de Documentos

A ingestão de documentos envolve leitura de arquivos, extração de texto e enriquecimento com metadados.

Apache Tika é frequentemente utilizado para extrair texto de PDFs e outros formatos binários.

Chunking e Text Splitters

Chunking é o processo de dividir documentos em partes menores semanticamente consistentes.

No Spring AI, TextSplitters como TokenTextSplitter permitem configurar tamanho do chunk e overlap para melhorar a recuperação.

Retrieval Augmented Generation (RAG)

RAG combina recuperação de informações externas com geração de texto por LLMs.

Esse padrão reduz alucinações e aumenta a precisão das respostas ao fornecer contexto relevante durante a inferência.

Boas Práticas para RAG com Spring AI

Utilizar chunks pequenos e coesos, enriquecer documentos com metadados e aplicar thresholds de similaridade melhora significativamente os resultados.

Separar ingestão, recuperação e geração torna a arquitetura mais escalável e testável.