RAGシステム技術ガイド

第1章: RAGの基本概念

RAG（Retrieval-Augmented Generation）は、大規模言語モデル（LLM）の回答精度を向上させるための技術です。従来のLLMは学習データに含まれる知識のみで回答を生成しますが、RAGでは外部のナレッジベースから関連情報を検索し、その情報をコンテキストとしてLLMに提供することで、より正確で最新の回答を生成できます。

RAGの主要なコンポーネントは以下の3つです:
1. ドキュメントの前処理とチャンク分割
2. ベクトル埋め込みと類似検索
3. プロンプト構築とLLM生成

第2章: チャンク分割戦略

効果的なチャンク分割はRAGシステムの性能に大きく影響します。主な分割戦略には以下があります:

固定サイズ分割:
テキストを一定の文字数で区切る最もシンプルな方法です。実装が簡単で処理速度が速い反面、文脈が途中で切れるリスクがあります。オーバーラップを設けることでこの問題を軽減できます。

構造ベース分割:
段落やセクションなどの文書構造に基づいて分割します。文脈の保持に優れていますが、段落のサイズが不均一になる可能性があります。

セマンティック分割:
文の意味的な類似度に基づいて分割します。最も高度な方法ですが、計算コストが高くなります。

第3章: ベクトル検索

ベクトル検索はRAGの中核技術です。テキストをベクトル空間に埋め込み、クエリとの類似度を計算して関連ドキュメントを取得します。

主要な埋め込みモデル:
sentence-transformers/all-MiniLM-L6-v2: 軽量で高速、384次元
text-embedding-ada-002 (OpenAI): 高精度、1536次元
multilingual-e5-large: 多言語対応、1024次元

ベクトルデータベース:

pgvector: PostgreSQL拡張、既存インフラとの統合が容易

Pinecone: マネージドサービス、スケーラビリティに優れる

Weaviate: オープンソース、ハイブリッド検索対応