Explainable RAG Document Q&A; System

Purpose of the Document

The purpose of this document is to describe the design and implementation of an explainable Retrieval-Augmented Generation (RAG) system for document-based question answering. The system aims to improve answer accuracy while reducing hallucinations by grounding responses in retrieved document content.

Problem Statement

Large Language Models often generate incorrect or unverifiable answers when responding to questions about private or domain-specific documents. This problem becomes critical in enterprise and research settings where accuracy and traceability are required.

Methodology and Approach

The system follows a Retrieval-Augmented Generation approach. Documents are first split into token-based chunks of approximately 300 to 500 tokens with overlapping regions. Each chunk is then converted into a numerical embedding representation. When a user submits a query, the system retrieves the most relevant chunks using vector similarity search and generates an answer using only the retrieved context.

Technology Stack

The system is implemented using Python. SentenceTransformer models are used to generate embeddings, and FAISS is used as the vector database for fast similarity search. The user interface is built with Streamlit, and Gemini is used as the language model for answer generation when an API key is available.

System Components and Steps

The main steps in the pipeline include document ingestion, text chunking, embedding generation, vector indexing, semantic retrieval, prompt construction, and answer generation with citations. Each retrieved chunk includes metadata such as document source and page number.

Results and Outcomes

Evaluation on a fixed question set demonstrated that retrieval tuning and prompt optimization improved answer accuracy significantly. The system was also able to maintain low latency, processing most queries within a few seconds.

Limitations and Challenges

The system depends on the quality of document text extraction and embeddings. Poorly formatted documents or very small datasets may reduce retrieval effectiveness. Additionally, evaluation based on keyword matching provides only a coarse estimate of answer quality.

Future Work

Future improvements include adding hybrid retrieval methods, reranking retrieved chunks using cross-encoders, incorporating more advanced evaluation metrics, and deploying the system as a scalable cloud-based service.