Medical Chatbot using Retrieval-Augmented Generation (RAG) and LLaMA

This project addresses the challenge of efficiently retrieving accurate and contextually relevant medical information from extensive collections of reference literature. In the medical domain, both practitioners and learners frequently require rapid access to verified information from clinical guidelines, research papers, and authoritative sources. Manual search within such large datasets is time-intensive and susceptible to human error, potentially delaying decision-making. The objective of this system is to automate information retrieval and synthesis, thereby ensuring precision, minimizing hallucinations, and improving the accessibility of reliable medical knowledge.

The dataset for the chatbot comprises medical reference PDFs stored in the data/ directory. These documents are ingested using LangChain's DirectoryLoader and PyPDFLoader. The raw text is segmented into overlapping chunks of 500 characters with an overlap of 50 characters through the RecursiveCharacterTextSplitter to preserve semantic continuity. Each chunk is then transformed into dense vector embeddings via the sentence-transformers/all-MiniLM-L6-v2 model from HuggingFace. The embeddings are stored in a FAISS vector database (vectorstore/db_faiss) to enable high-speed similarity-based retrieval.

The core architecture follows a Retrieval-Augmented Generation (RAG) framework. User queries, submitted through a Streamlit-based interface, are processed by the FAISS retriever, which identifies the top three most relevant document chunks. These chunks are then incorporated into a custom-designed prompt template that explicitly instructs the language model to confine its response to the provided context and return an "I don't know" statement when the answer cannot be derived. The response generation is carried out by the Groq-hosted **LLaMA-4 Maverick 17B** model, configured with a temperature setting of 0.0 to ensure deterministic, fact-based outputs.

Technically, the system integrates Python, LangChain, Streamlit, HuggingFace embeddings, FAISS vector storage, and Groq's hosted LLaMA large language model. The design supports scalability, enabling the incorporation of new medical documents into the retrieval pipeline without retraining the generative model. The chatbot delivers concise, context-grounded answers accompanied by source document references, thereby ensuring transparency and trustworthiness. This makes it a practical tool for healthcare professionals, medical students, and informed patients seeking dependable, real-time access to authoritative medical information.