Healthcare Knowledge Assistant

Date: Oct 6, 2025

Author:

MD. Golam Mostofa

golammostofa10001@gmail.com

Table of Contents

1. Overview

The **Healthcare Knowledge Assistant** is a **Multilingual Retrieval-Augmented Generation (RAG)** system designed to help healthcare professionals and researchers' access, retrieve, and generate medical information in multiple languages.

It combines **semantic search**, **language translation**, and **context-aware AI generation** to bridge multilingual healthcare knowledge bases.

The system is built with **FastAPI** and integrates:

- **Sentence Transformers** for multilingual embeddings
- **FAISS** for high-speed vector similarity search
- **LangDetect** for automatic language detection
- **Deep Translator** for cross-language understanding
- **LLMs** for generating coherent, context-driven responses

2. Objective

The primary goal of this system is to:

- Enable **multilingual access** to healthcare knowledge.
- Provide **semantic retrieval** of the most relevant document chunks.
- Generate **contextual, accurate AI responses** based on retrieved medical documents.
- Serve as a **backend API** for intelligent healthcare assistants, research tools, or chatbots.

Specific Objectives

- Automate **document ingestion** and embedding creation.
- Enable **cross-lingual search**, allowing users to query in any supported language.
- Ensure **data security** through API key-based access control.
- Maintain **scalability and modularity** for future model upgrades.

3. Architecture

The system follows a modular and layered architecture designed around the RAG framework.

Major Components

Layer	Component	Description
Data Layer	**FAISS Vector DB**	Stores document embeddings and metadata for semantic retrieval.
Processing Layer	**MultilingualRAG Engine**	Handles ingestion, embedding, retrieval, and generation processes.
Service Layer	**FastAPI Routers** (`ingest`, `retrieve`, `generate`)	RESTful endpoints that expose the RAG functionalities.
Security Layer	**API Key Middleware**	Ensures only authorized requests access the system.
Integration Layer	**LLM / Translator**	Provides language translation and response generation.

3.1 Architecture Workflow

Below is the process architecture diagram:

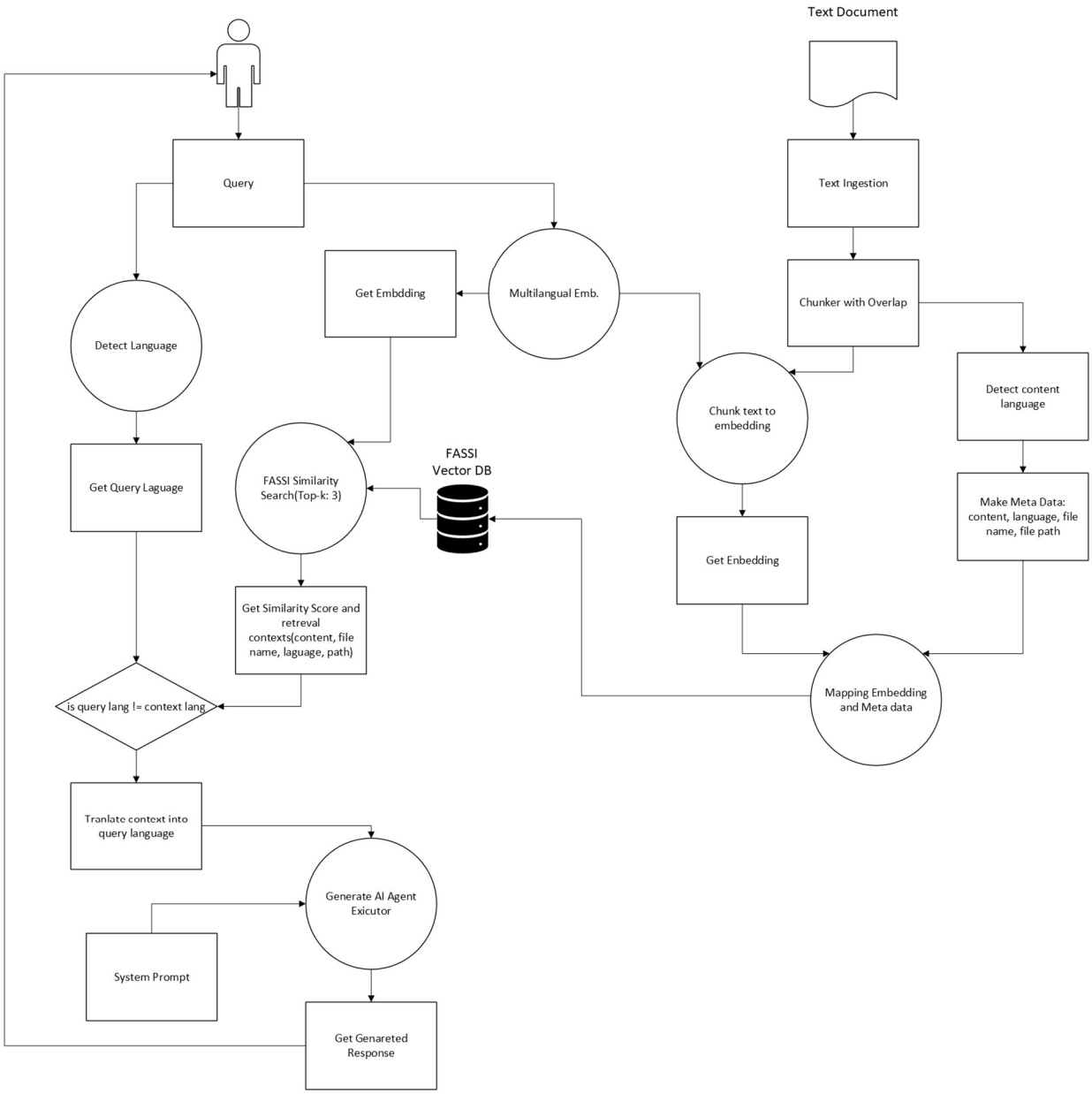

Fig. 1: Architecture of Healthcare Knowledge Assistant

A. Document Ingestion Flow

1. **Text Ingestion** → Upload documents through `/ingest/`.
2. **Language Detection** → Determine document's primary language.
3. **Chunking with Overlap** → Split documents for better retrieval granularity.
4. **Embedding Generation** → Convert chunks into vector embeddings.
5. **Metadata Mapping** → Store content, file name, language, and path.

6. **Vector Storage** → Save embeddings and metadata into FAISS and JSON store.

B. Query Processing Flow

1. **Language Detection** → Identify query language.
2. **Embedding Generation** → Convert the query into an embedding vector.
3. **Similarity Search (Top-k=3)** → Retrieve most relevant chunks from FAISS.
4. **Cross-Language Check** → If document and query languages differ, translate context to match query language.

C. Response Generation Flow

1. **Context Preparation** → Merge retrieved text chunks.
2. **Prompt Construction** → Combine context and query into a prompt template.
3. **AI Generation** → LLM generates a coherent, multilingual response.
4. **Response Delivery** → Return the result in the user's query language.

4. System Details

Core Class: `MultilingualRAG`

Handles the end-to-end RAG pipeline:

- Embedding creation (`SentenceTransformer`)
- Language detection and translation (`LangDetect`, `TranslatorAdapter`)
- FAISS-based similarity search
- AI response generation (via external LLMs)

Data Storage

File	Description
`data/faiss_index.faiss`	Stores vectorized embeddings for similarity search
`data/doc_store.json`	Stores metadata and file context

`media/uploads/`	Raw uploaded documents

APIs

Endpoint	Method	Purpose
`/ingest/`	POST	Upload and embed text documents
`/retrieve/`	POST	Retrieve top-k similar text chunks
`/generate/`	POST	Generate multilingual contextual responses
`/`	GET	Health check endpoint

Technology Stack

Category	Technology
Framework	FastAPI
Embeddings	Sentence Transformers
Vector Database	FAISS
Language Detection	LangDetect
Translation	Deep Translator
LLM Integration	HuggingFace Open Source
Containerization	Docker
Dependency Manager	UV
CI/CD	GitHub Actions

6. Limitations

Despite its modular architecture, the current implementation has the following constraints:

1. **Limited to Text Files:** PDF or image-based document ingestion is not yet supported.
2. **Limited Context Size:** Retrieval and generation rely on truncated context (top-k=3 chunks).
3. **Static Translation Layer:** Uses single-threaded translation; may slow down for large multilingual corpora.

4. **No Conversational Memory:** Each query is treated independently; multi-turn context is not retained.
5. **Hardware-Dependent Performance:** FAISS and embedding generation performance may vary with CPU/GPU availability.
6. **Healthcare Disclaimer:** The AI-generated responses are not verified by medical experts; for reference use only.

5. Future Improvements

Improvement Area	Description
Multi-format Support	Add ingestion for PDF, DOCX, and scanned documents (OCR).
Hybrid Search	Combine FAISS (semantic) with BM25 (lexical) for better relevance.
Conversational Memory	Enable multi-turn dialogue context for continuous Q&A.
Async Translation	Parallelize multilingual translation to improve speed.
Advanced Caching	Cache embeddings and LLM responses to reduce latency.
Fine-tuned LLM	Train domain-specific LLM on healthcare corpora for better accuracy.
Dashboard UI	Add a lightweight front-end for document management and query tracking.

6. Project Resources

Resource	Link
Video	https://drive.google.com/file/d/1pb-g3R7t7ENNPMsFfAF2TqxdmjyA9cPw/view?usp=sharing
GitHub Repository	https://github.com/shuvo881/Healthcare-Knowledge-Assistant
Docker Hub Image	https://hub.docker.com/r/mdgolammostofa705/healthcare-knowledge-assistant
Embedding Model	paraphrase-multilingual-MiniLM-L12-v2

Generation Model	MiniMax-M2

7. Discussion

The **Healthcare Knowledge Assistant** demonstrates a robust and scalable multilingual RAG architecture that can be adapted to various domains, not limited to healthcare.

Its modularity allows flexible upgrades—such as plugging in new LLMs, switching vector databases, or extending translation backends.

Key Strengths:

- Multilingual semantic understanding
- Efficient retrieval using FAISS
- Clear separation between ingestion, retrieval, and generation
- Secure access control via API keys

Areas for Further Research:

- Enhancing factual accuracy in AI-generated responses
- Optimizing FAISS for distributed indexing
- Incorporating reinforcement learning for adaptive knowledge retrieval

7. References

1. Multilingual RAG: Cross-Language Document Retrieval
2. https://medium.com/data-science-at-microsoft/building-and-evaluating-multilingual-rag-systems-943c290ab711
3. https://zilliz.com/blog/building-multilingual-rag-milvus-langchain-openai
4. https://www.elastic.co/search-labs/blog/building-multilingual-rag-with-elastic-and-mistral