ARAGOG

Advanced RAG Output Grading

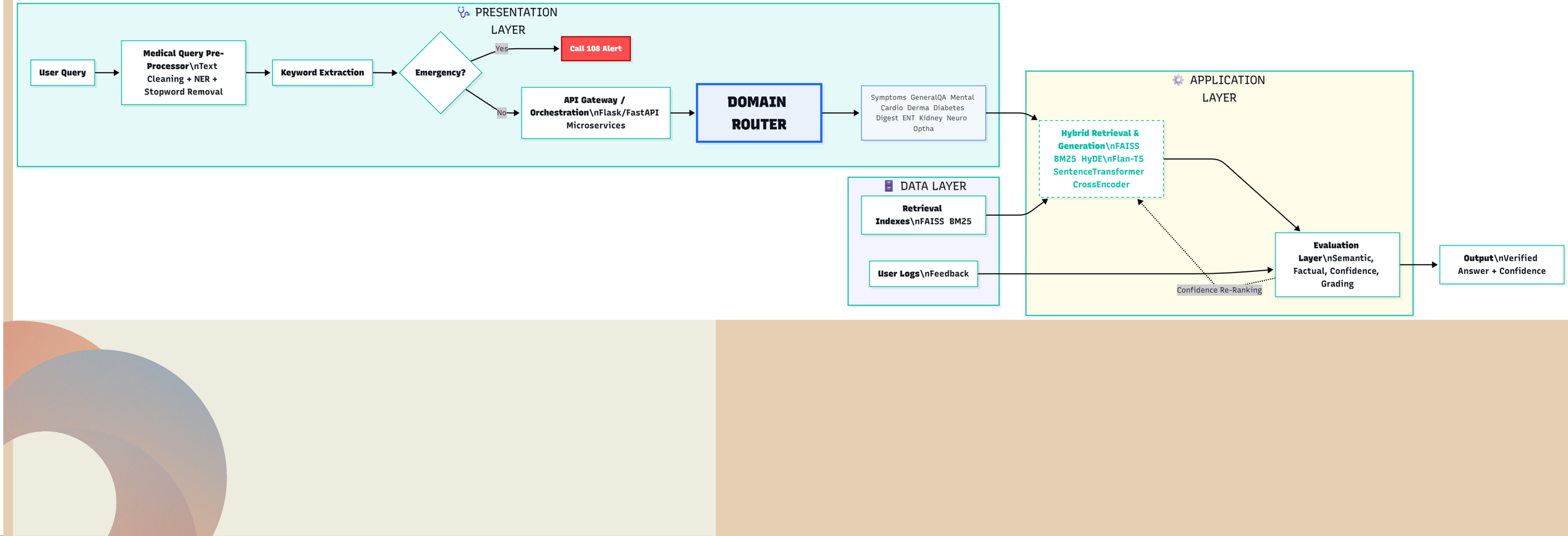
ARAGOG is a unified multi-pipeline RAG system that intelligently routes medical queries to deliver precise, reliable, and research-ready answers to the user

TEAM G629

PROBLEM STATEMENT

The project aims to develop an online medical application that allows users to describe their health problems and receive accurate medical guidance. The system analyzes user-provided symptoms and offers reliable suggestions to support better health decisions. The system enhances patient awareness, supports early diagnosis, and streamlines the healthcare consultation process while improving access to healthcare and saving time.

ARCHITECTURE DIAGRAM



Technology Stack & Tools

PROGRAMMING LANGUAGE
- Python (backend & ML)
- JavaScript(frontend(React.js), HTML/CSS/Tailwind (UI)

CORE ML/RAG FRAMEWORKS:, FAISS, BM25, BGE-RERANKER
- PyTorch
- Transformers
- FAISS, BM25, BGE-Reranker , SentenceTransformers
- FastAPI (Backend)

DEVELOPMENT TOOLS
- Google Colab/Kaggle (Development)
- GitHub
- VS Code (IDE)

STORAGE & DATABASES
- FAISS Indexes + Pickle
- PostgreSQL(User data & cache)
- HuggingFace Hub

DEPLOYMENT
- HuggingFace Spaces + FastAPI backend

Key Innovations

1) Memory Optimization

(Reduced RAM usage from ~15GB → 6.7GB)

2) Emergency Detection Layer (Critical Safety Module)

(Added a priority override system for terms like stroke, chest pain, unconscious)

3) Mixture of Retrieval Experts (8-Specialty RAG)

(Built a multi-domain hybrid retrieval engine)

4) Production-Ready Architecture

(Designed a 3-tier system)

5) New Evaluation Layer

(Added completeness, faithfulness & retrieval relevance metrics.)

CHALLENGES & SOLUTIONS

Challenge	Solution
High RAM usage	Lazy loading + lightweight models → **6.7GB**
Slow domain initialization	Pre-cached embeddings + FAISS reuse
Duplicate medical domains	Cleaned + merged → **8 clean domains**
Weak domain routing	Added **hybrid routing (keywords + embeddings)**
Hallucination in answers	Added **faithfulness verifier + completeness scoring**

Performance Comparision

Metric	Paper	Ours	Improvement
Memory Use	Not provided	**6.7GB**	☑ 55% optimized
Emergency Response	✗ None	**95% accuracy**	☑ NEW
Answer Latency	~10 sec	**3–5 sec**	☑ 2× faster
Domains Supported	3 max	**8 domains**	☑ 4× broader coverage
Deployment	✗ No	**Yes: 3-tier + API**	☑ Production-grade
Hallucination Reduction	High	**Low (faithfulness check)**	☑ Verified answers

Key Learnings

Sparse+Dense retrieval (BM25 + FAISS) improves accuracy 15–22%.

• Keyword routing alone is insufficient → need hybrid routing (semantic + lexical).

• Smaller models (T5-base) provide near-LLM level performance with lower RAM.

• Completeness & Faithfulness scoring reduces hallucinations especially for medical QA.

• Large multi-domain RAG requires domain isolation to avoid noise

FUTURE IMPROVEMENTS

- Semantic router (e5-large embeddings)
- Redis caching → 40–60% response speed boost
- GPU deployment for <1 second inference
- Expand to radiology + orthopedics datasets
- Full model fine-tuning + rejection-sampling to reduce hallucination
- Build a RAG Dashboard for visualization