© APR 2026 | IRE Journals | Volume 9 Issue 10 | ISSN: 2456-8880
DOI: https://doi.org/10.64388/IREV9I10-1716796

EduRAG: A Multi-Format Retrieval-Augmented Generation Assistant for Academic Question Answering

M. SYAM PRAKASH[1], G. V. CHARAN KUMAR[2], D. GAYATHRI[3]
[1, 2]*Department of Computer Science and Engineering, SCSVMV University, Kanchipuram, Tamil Nadu, India*
[3]*Assistant Professor, Department of CSE, SCSVMV*

Abstract- ***Large Language Models (LLMs) can produce fluent responses, but they often provide unsupported information when answers rely on specific documents. This paper introduces EduRAG, a Retrieval-Augmented Generation (RAG) assistant designed for answering student questions based on uploaded academic materials. The proposed system combines multiple file formats, semantic chunking, dense embedding generation, vector similarity retrieval, and context-based response synthesis. EduRAG supports various educational formats, including PDF, DOCX, PPTX, TXT/MD, CSV/XLSX, and image-based text through OCR. This allows practical use across different classroom resources. The backend is built using Flask and includes APIs for health monitoring, document upload, indexing, querying, file listing, and deletion. Sentence-Transformers are used for creating semantic embeddings, and FAISS provides efficient nearest-neighbor retrieval. For generation, the architecture supports both cloud-hosted and local LLM options through configurable providers. Index persistence and metadata storage allow for reusable sessions and quicker follow-up queries. Experimental results on a mix of academic documents show that EduRAG enhances answer relevance and grounding compared to direct LLM prompting without retrieval. It maintains acceptable latency for interactive educational use. The system shows that retrieval-based prompting significantly reduces the risk of generating false information and increases trustworthiness in academic assistants. EduRAG offers a low-cost, flexible foundation for institutional learning support and can be further improved with reranking, citation tracing, and hybrid retrieval methods.***

Index Terms—Retrieval-Augmented Generation, Educational Chatbot, FAISS, Semantic Search, Document Question Answering, Large Language Models.

I. INTRODUCTION

Students frequently use lecture notes, slides, reports, and tabular materials for learning and preparing for exams. Traditional keyword searches often fall short for conceptual queries. Direct LLM-based chat can generate plausible answers, but they may lack a solid foundation. Retrieval-Augmented Generation (RAG) tackles this issue by fetching relevant document context before generating answers.

This work presents EduRAG, an educational assistant that answers questions based solely on uploaded study resources. The primary goal is to improve factual grounding while ensuring clarity and usability within actual academic workflows.

II. RELATED WORK

RAG architectures have shown strong performance in knowledge-heavy NLP by combining retrieval and generation [1]. Dense retrieval methods enhance semantic matching for open-domain question answering [2]. Sentence-BERT embeddings provide effective vector representations at the sentence level [3], and FAISS facilitates scalable nearest-neighbor searches in vector spaces [4].

In education, many assistants are conversational, but they lack solid document grounding. EduRAG aims for practical multi-format ingestion and retrieval-based answer generation for student materials.

III. PROBLEM STATEMENT

Students need an assistant capable of answering based on their own academic documents in various file types. Current systems often fall short because they either support a limited range of formats or provide answers without reliable source grounding. The challenge is to create and implement a deployable academic assistant that is document-grounded, semantically searchable, and responsive.

IV. PROPOSED SYSTEM

EduRAG has a modular structure consisting of three main layers:

- API layer for upload, query, and file management
- Parsing layer for text extraction from different formats
- RAG core for chunking, embedding, retrieval, and generation

Workflow: Uploaded file → extracted text → overlapping chunks → embeddings → FAISS index → top-k retrieval for query → context-based LLM answer with source documents.

V. METHODOLOGY

A. Document Parsing

The parser handles PDF, DOCX/DOC, PPTX, TXT/MD, CSV/XLSX, and OCR for images, covering a wide range of academic content.

B. Text Chunking

The extracted text is divided into overlapping word chunks (chunk size 500, overlap 50) to keep semantic continuity across boundaries.

C. Embedding and Indexing

Chunks are embedded using *all-MiniLM-L6-v2* (384 dimensions) and stored in a FAISS IndexFlatL2 vector index.

D. Retrieval and Generation

When a question is posed, top-k relevant chunks are retrieved and included in a controlled educational prompt. The system supports both OpenAI and local Ollama options for generation.

VI. IMPLEMENTATION

The backend is developed in Python using Flask and Flask-CORS. Core modules include:

- *app.py*: API endpoints (/api/health, /api/upload, /api/query, /api/files, delete endpoint)
- *file_parser.py*: handling multi-format parsing and OCR
- *rag_engine.py*: managing chunking, embedding, FAISS retrieval, prompting, and LLM inference

Index and metadata storage allow for repeated session usage.

VII. RESULTS AND DISCUSSION

In comparison to direct LLM prompting, EduRAG enhances answer relevance and source grounding for document-specific academic questions. Retrieval adds some latency but keeps response times suitable for interactive use. OCR-enabled inputs work well for scanned content, although performance can vary based on image quality. The current system delivers a robust baseline performance with straightforward deployment.

VIII. CONCLUSION

This paper presented EduRAG, a retrieval-based educational assistant for multi-format academic documents. By merging semantic retrieval with controlled generation, the system improves factual reliability and decreases the risk of generating false information compared to direct prompting. The architecture is modular, affordable, and adaptable for use in educational institutions. Future work will focus on reranking, citation-level evidence extraction, hybrid retrieval methods, and multilingual support.

ACKNOWLEDGMENT

The authors thank D. Gayathri, Assistant Professor, Department of CSE, SCSVMV, for her ongoing guidance and support throughout this project.

REFERENCES

[1] P. Lewis et al., “Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks,” Advances in Neural Information Processing Systems, 2020.

[2] V. Karpukhin et al., “Dense Passage Retrieval for Open-Domain Question Answering,” EMNLP, 2020.

[3] N. Reimers and I. Gurevych, “Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks,” EMNLP-IJCNLP, 2019.

[4] J. Johnson, M. Douze, and H. Jegou, “Billion-scale similarity search with GPUs,” IEEE Transactions on Big Data, 2019.

[5] Flask Documentation. [Online]. Available: https://flask.palletsprojects.com

[6] FAISS Documentation. [Online]. Available: https://faiss.ai

[7] OpenAI API Documentation. [Online]. Available: https://platform.openai.com/docs

[8] Ollama Documentation. [Online]. Available: https://ollama.com