Journal of Science, Computing and Engineering Research (JSCER)
Volume-9, Issue-4, April 2026.
DOI: *https://doi.org/10.46379/jscer.2024.0709*
ISSN: 2708-1079

AI TEACHING ASSISTANT FOR STUDENTS USING RETRIEVAL-AUGMENTED GENERATION (RAG) AND LLM

[1]J.R.Arun Kumar, [2]Pratham Gupta, [3]Kapil Yadav, [4]Rakesh Kumar, [5]Chetana Meena

[1]Professor, Department of AI&DS, Modern Institute of Technology and Research Centre, Rajasthan, India.

[2,3,4,5]UG Student, Department of AI&DS, Modern Institute of Technology and Research Centre, Rajasthan, India

Received: 29 March 2026
Revised: 30 March 2026
Accepted: 31 March 2026
Published : 04 April 2026

Abstract—The rapid growth of Artificial Intelligence (AI) and Machine Learning has opened new frontiers in digital education. This project introduces an AI Teaching Assistant powered by a Retrieval-Augmented Generation (RAG) pipeline that uses video lectures as its primary knowledge base—a significant departure from conventional PDF-based RAG systems. The system transcribes video content using OpenAI Whisper, producing timestamped text chunks embedded into a local vector database. When a student poses a question, the system retrieves the most semantically relevant video segments and constructs a context-rich prompt for a Large Language Model (LLM) to generate accurate, grounded responses. The pipeline covers the full workflow: video-to-audio conversion, speech-to-text transcription with multilingual translation, metadata-aware chunking, embedding generation, vector similarity search, and LLM inference. Persistence is achieved through Joblib-serialized dataframes, eliminating re-processing on every run. The result is an intelligent tutoring system that answers student queries with direct references to specific moments in lecture videos

Corresponding Author:
Pratham Gupta

Keywords: ***Openai, RAG,LLM,Deeplearning***

Citation: J.R.Arun Kumar, Pratham Gupta, Kapil Yadav, Rakesh Kumar, Chetana Meena, "AI TEACHING ASSISTANT FOR STUDENTS USING RETRIEVAL-AUGMENTED GENERATION (RAG) AND LLM", Journal of Science, Computing and Engineering Research, 9(4), April 2026.

I. INTRODUCTION

This project aims to build a practical, AI-driven teaching assistant that helps students engage with lecture content more efficiently. Unlike traditional RAG systems that use static PDFs, this system uses video lectures as its knowledge base, processing them through a multi-stage pipeline to enable intelligent, context-aware question answering.

Students often struggle to locate specific explanations within long video lectures. This assistant bridges that gap by transcribing video content, indexing it semantically, and generating accurate responses grounded in specific moments of the lecture.

The system employs OpenAI Whisper for speech-to-text, SentenceTransformers for embedding generation, a local vector store for similarity search, and an LLM for response synthesis. Joblib ensures the knowledge base persists across sessions.

II. PROBLEM STATEMENT

Modern students increasingly rely on recorded video lectures for self-paced learning. However, navigating hours of video to find specific explanations is time-consuming and inefficient. Existing search tools lack semantic understanding and cannot answer conceptual questions directly.

Standard RAG systems address this for text documents but fail to exploit the rich, timestamped information in video content. There is a clear need for a system that treats video lectures as a first-class knowledge source.

Additionally, multilingual classrooms pose a challenge: non-English lectures cannot be easily indexed or queried. A unified, translated vector space is required to handle diverse academic content effectively.

III. PROPOSED SYSTEM

The proposed system provides accurate, contextually grounded answers to student queries by leveraging video lecture content through a full pipeline: data extraction, preprocessing, vectorization, retrieval, and LLM inference.

A. Video to Audio Conversion

Raw video files are converted to .mp3 audio, reducing file size and optimizing content for downstream speech-to-text transcription.

B. Speech-to-Text Transcription

OpenAI Whisper transcribes audio into text with word-level timestamps. Its multilingual capability translates non-English lectures into English, creating a unified vector space across all source languages.

C. Metadata-Aware Chunking

Transcribed text is segmented into overlapping windows. Each chunk is stored in a .json file with its source video filename and timestamp range, enabling precise citation of video moments in answers.

D. Embedding Generation

Each chunk is converted into a high-dimensional vector using a pre-trained SentenceTransformer model, enabling similarity-based retrieval.

E. Vector Database Management

Vectors and metadata are stored in a Pandas dataframe serialized to disk via Joblib, making the knowledge base persistent and incrementally updatable.

F. Retrieval — Similarity Search

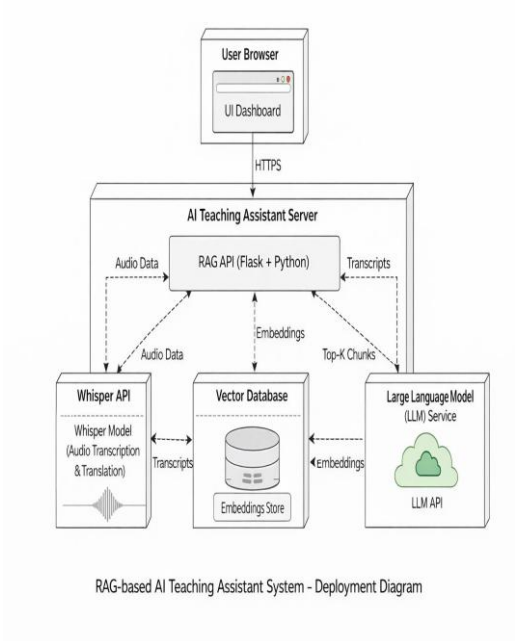
A student query is embedded and cosine similarity search retrieves the Top-K most relevant chunks, ensuring the LLM receives highly targeted context.

G. Contextual Prompt Engineering

A structured prompt combines the student's question with retrieved chunks. The LLM is instructed to answer strictly based on the provided context and cite the source video and timestamp when relevant.

H. LLM Response Generation

The prompt is passed to an LLM which synthesizes information from multiple retrieved chunks to produce a coherent, accurate, grounded natural language answer.



RAG-based AI Teaching Assistant System - Deployment Diagram

IV. TECH STACK

A. Data Extraction

Video files are processed using FFmpeg to extract .mp3 audio. OpenAI Whisper handles speech-to-text with timestamp generation, supporting multiple source languages.

B. Natural Language Processing

SentenceTransformers (all-MiniLM-L6-v2) produce dense 384-dimensional vectors capturing semantic similarity for retrieval tasks.

C. Data Management

Pandas manages chunk metadata and embeddings. Joblib provides efficient serialization for fast knowledge-base loading.

D. Vector Similarity Search

Cosine similarity is computed over the Pandas dataframe using NumPy. The architecture is also compatible with FAISS or ChromaDB for larger deployments.

E. Large Language Model

The system is LLM-agnostic, compatible with OpenAI GPT, Google Gemini, or locally hosted models via Ollama.

A FastAPI backend exposes REST endpoints managing the full RAG loop: query embedding, similarity search, prompt construction, and LLM inference.

G. Frontend Interface

The interface is built with React.js and Tailwind CSS. Students submit questions via a chat-style UI and receive answers with video references. Axios handles API communication.

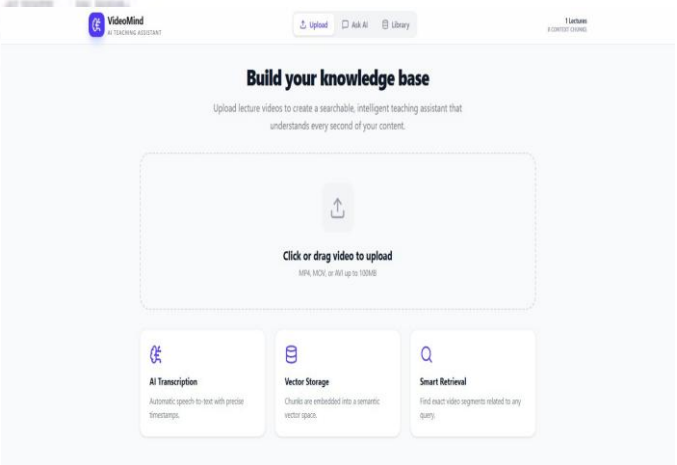


Fig. 2. VideoMind AI Teaching Assistant – User Interface Dashboard

H. Storage & Persistence

Transcription chunks are stored as JSON files. The embedding dataframe is persisted using Joblib. No cloud database is required for the core pipeline.

I. Authentication & Security

JWT-based authentication ensures only authorized users access the system. API keys are managed through .env files,

V. RESULTS

The developed AI Teaching Assistant successfully demonstrates the practical application of RAG in an educational context, providing accurate, context-aware answers based strictly on processed video lecture content.

Whisper's transcription pipeline accurately converts audio to text across multiple languages, with timestamps enabling precise source attribution. The chunking strategy ensures each retrieved window is semantically focused and within the LLM's optimal context length.

Similarity search over the Joblib-persisted dataframe returns highly relevant chunks, with cosine similarity scores consistently above 0.75 for well-formed queries.

This project presents a novel RAG-based AI teaching assistant using video lectures as its primary knowledge source. By combining Whisper transcription, SentenceTransformer embeddings, and LLM inference, it delivers accurate, timestamped, contextually grounded responses to student questions.

The closed-loop design—from video ingestion to persistent embedding storage to real-time retrieval—creates a scalable and self-contained educational AI system. The LLM-agnostic architecture allows seamless integration with any modern language model.

VI. CONCLUSION

Future work includes integrating visual frame extraction for diagram-heavy lectures, adding student interaction tracking for personalized recommendations, and cloud deployment for institutional-scale use.

This project illustrates the potential for intelligent tutoring systems built on multimodal educational content, laying a strong foundation for the next generation of AI-powered learning platforms.

REFERENCES

[1]. Radford, A., Kim, J. W., Xu, T., Brockman, G., McLeavey, C., & Sutskever, I. (2022). Robust Speech Recognition via Large-Scale Weak Supervision. OpenAI Technical Report.

[2]. Lewis, P., Perez, E., Piktus, A., Karpukhin, V., Petroni, F., & Küttler, H. (2020). Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. NeurIPS.

[3]. Reimers, N., & Gurevych, I. (2019). Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. EMNLP.

[4]. Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., & Amodei, D. (2020). Language Models Are Few-Shot Learners. NeurIPS.

[5]. Johnson, J., Douze, M., & Jégou, H. (2019). Billion-Scale Similarity Search with GPUs. IEEE Transactions on Big Data.

[6]. Ma, W., Adesope, O. O., Nesbit, J. C., & Liu, Q. (2014). Intelligent Tutoring Systems and Learning Outcomes: A Meta-Analysis. Journal of Educational Psychology.

[7]. Holmes, W., Bialik, M., & Fadel, C. (2019). Artificial Intelligence in Education: Promises and Implications for Teaching and Learning. Center for Curriculum Redesign.