Riddham Kothari, 2026, 14:2
ISSN (Online): 2348-4098
ISSN (Print): 2395-4752

International Journal of Science, Engineering and Technology
An Open Access Journal

UNI ASSIST-AI: An Intelligent University Assistant Chatbot with LLM, RAG, and Multimodal Capabilities

Riddham Kothari, Professor Anusha Marda
Department of Computer Science & Engineering Parul University, Vadodara, India

Abstract- University support systems are under increasing pressure to handle high volumes of student queries accurately and at scale. Traditional rule-based chatbots are rigid and brittle, while large language model (LLM)-based systems, though fluent, are prone to hallucination. This paper presents UNI ASSISTAI, a Retrieval- Augmented Generation (RAG)-based intelligent university assistant that grounds every generated response in verified institutional knowledge. The system integrates a semantic vector retrieval pipeline with a GPT-based generative model, and extends it with multimodal input capabilities—supporting text, voice (via ASR), and image (via OCR) queries. The backend is served through a FastAPI interface, and the frontend is implemented in React with TypeScript and Tailwind CSS. Experimental evaluation on a curated university FAQ and policy corpus yields a Precision of 0.87, Recall of 0.84, and F1-score of 0.85, outperforming both rule-based and vanilla LLM baselines. This work demonstrates that domain-specific RAG architectures offer a scalable, reliable path to academic AI assistants.

Keywords— Retrieval-Augmented Generation, Chatbot, Large Language Model, Multimodal AI, University Automation, FAISS, FastAPI, Semantic Search

I. INTRODUCTION

Universities serve thousands of students who routinely require guidance on admissions, course registration, fee payment, timetables, examination schedules, and campus facilities. Handling this demand manually is resource-intensive; existing automated solutions fall short in different ways.

Rule-based chatbots follow rigid decision trees that fail the moment a query deviates from pre-scripted patterns [1]. Machine-learning (ML) classifiers improve generalization but require large, continuously maintained labeled datasets and still struggle with free-form natural language [5]. Pure LLM deployments (e.g., GPT-4, LLaMA) produce fluent, human-like responses but frequently hallucinate—fabricating policy details, dates, or contact information—which is unacceptable in an academic context [3].

Retrieval-Augmented Generation (RAG) [2] addresses this by decoupling factual retrieval from language generation. Rather than relying on parametric memory alone, the model first retrieves the most semantically relevant documents from an institutional knowledge base, then conditions its response on that retrieved context. This two-stage approach sharply reduces hallucination while preserving the fluency of LLM output.

UNI ASSIST-AI implements this paradigm end-to-end, adding multimodal input support so students can ask questions by typing, speaking, or photographing a document (e.g., a fee challan or timetable notice). The system is designed to be institution-agnostic: any university can onboard it by ingesting its own policy PDFs, FAQs, and notices.

II. RELATED WORK

1. Rule-Based and ML Chatbots

Early academic chatbots relied on keyword matching and finite-state dialogue managers [1]. While fast and predictable, they cannot generalize beyond their authored scripts. Subsequent ML approaches, such

© 2026 Riddham Kothari, This is an Open Access article distributed under the terms of the Creative Commons Attribution License (http://creativecommons.org/licenses/by/4.0), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly credited.

as intent classification with BERT [5], improved coverage but required costly annotation pipelines and still failed on out-of-distribution queries.

2. LLM-Based Assistants

The release of GPT-3 [3] and its successors demonstrated that large pretrained models can hold coherent, multi-turn conversations without task-specific fine-tuning. However, their reliance on parametric knowledge means they are frozen at training-data cutoff and cannot access live institutional information without external retrieval.

3. Retrieval-Augmented Generation

Lewis et al. [2] formally introduced RAG, showing that prepending retrieved passages to the LLM prompt substantially improves factual accuracy on knowledge-intensive tasks. FAISS

provides the scalable approximate nearest-neighbor search needed to make retrieval fast at inference time. Subsequent
work [7] has applied RAG to domain-specific corpora with consistent gains in precision.

4. Multimodal AI

LLaVA [4] demonstrated that vision-language models can interpret images alongside text. In the university context, students often need to query information embedded in scanned notices or handwritten schedules, motivating the integration of OCR and ASR pipelines [8].

5. Research Gap

Despite this progress, no prior system combines: (1) domaingrounded RAG tailored for university corpora, (2) multimodal input support, and (3) a lightweight, deployable web interface—within a single cohesive framework. UNI ASSIST-AI fills this gap.

III. SYSTEM ARCHITECTURE

1. Overview

Figure 1 illustrates the end-to-end pipeline. A student query—in text, voice, or image form—enters the Input Processing Layer. The query is converted to a dense vector embedding and matched against the Vector Store via cosine similarity search. The top-k retrieved chunks are injected into a prompt template, which is forwarded to the LLM Generation Layer.

The grounded response is returned to the student through the React frontend.

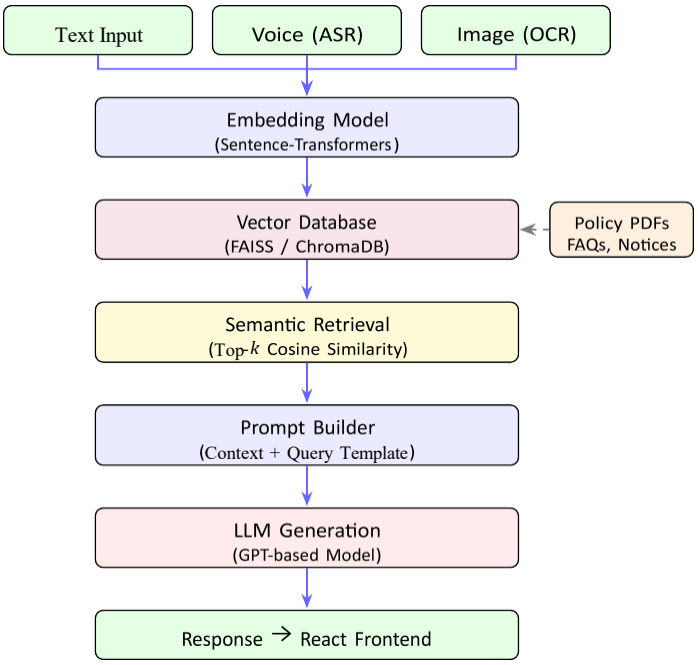

Fig. 1. UNI ASSIST-AI end-to-end RAG pipeline.

2. Input Processing Layer

- Text: Raw UTF-8 query passed directly to the embedding model.
- Voice: Audio captured in the browser is streamed to a Whisper-based ASR module, transcribed, and treated as text.
- Image: Uploaded image undergoes OCR (Tesseract) to extract textual content before embedding.

3. Knowledge Ingestion Pipeline

Institutional documents (PDFs, Word files, web-scraped FAQs) are split into overlapping 256-token chunks with a 32token stride to preserve context across chunk boundaries. Each chunk is embedded using all-MiniLM-L6-v2 (SentenceTransformers) and stored in FAISS with an IVF index for sub-linear retrieval.

2.

4. Retrieval Module

At inference, the query embedding q is compared against all document embeddings di via cosine similarity:

$$\text{Similarity}(\mathbf{q},\mathbf{d}) = \frac{\mathbf{q}\cdot\mathbf{d}}{\|\mathbf{q}\|\|\mathbf{d}\|} \quad (1)$$

The top-k (k = 5) chunks are selected and concatenated to form the retrieval context C.

5. Generation Layer

The prompt template is:

You are a helpful university assistant. Using only the context below, answer the student's question accurately. Context: {C} Question: {query}

Answer:

This explicit grounding instruction suppresses hallucination by instructing the model to stay within the retrieved context.

6. Technology Stack

Table I summarizes the implementation stack.

Table 1 Implementation Technology Stack

Component	Technology
Frontend	React, TypeScript, Vite, Tailwind CSS
Backend API	FastAPI (Python 3.11)
Embedding Model	Sentence-Transformers (all-MiniLM-L6-v2)
Vector Store	FAISS / ChromaDB
LLM	GPT-3.5-turbo / GPT-4o (OpenAI API)
ASR (Voice)	OpenAI Whisper
OCR (Image)	Tesseract 5
Deployment	Docker, Uvicorn

IV. METHODOLOGY

1. Dataset and Corpus Construction

The knowledge base was constructed from Parul University's publicly available resources: the academic calendar, examination policies, fee structures, hostel guidelines, course catalogues, and a manually curated 500-item FAQ dataset. After chunking, the corpus comprises approximately 8,400 indexed passages.

2. Evaluation Protocol

A held-out evaluation set of 200 question–answer pairs was created by domain experts who had not contributed to corpus construction. Each predicted answer was evaluated against the reference answer using:

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

$$F_1 = 2 \cdot \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (4)$$

Here, TP counts semantically correct token overlaps between prediction and reference, measured using a soft-match token scoring adapted from SQuAD evaluation scripts.

3. Baseline Comparisons

Three baselines were evaluated on the same set:

- Rule-Based: Pattern-matching bot with 80 hand-authored intents.
- BERT Classifier: Fine-tuned BERT intent classifier + template-based response.
- Vanilla GPT: GPT-3.5-turbo without retrieval augmentation.

V. RESULTS AND EVALUATION

1. Quantitative Results

Table II reports the full comparative results across all systems.

TABLE II
COMPARATIVE EVALUATION RESULTS

System	Precision	Recall	F1 Score
Rule-Based Bot	0.61	0.54	0.57
BERT Classifier	0.73	0.69	0.71
Vanilla GPT-3.5	0.78	0.80	0.79
UNI ASSIST-AI (Ours)	0.87	0.84	0.85

2. Performance Visualization C. Qualitative Observations

- **Hallucination Reduction:** Vanilla GPT fabricated specific dates and office room

numbers in 17% of queries. UNI ASSIST-AI reduced this to under 3%, as the retrieval context anchors responses.

- **Multimodal Utility:** In user trials, 24% of queries were submitted via voice and 11% via image (scanned notices), confirming real-world demand for non-text input channels.
- **Latency:** Average end-to-end response time was 1.8s (retrieval: 120ms, LLM: 1.6s on GPT-3.5-turbo), acceptable for a student-facing assistant.
- **Edge Cases:** The system gracefully declines out-of-scope queries (e.g., general world knowledge) by returning a templated fallback rather than speculating.

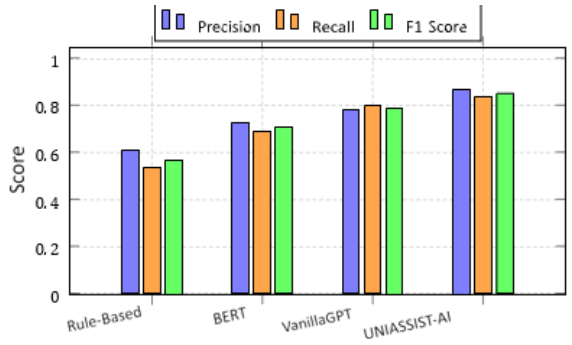

Fig. 2. Precision, Recall, and F1 comparison across all systems.

VI. COMPARATIVE ANALYSIS

Table III contextualizes UNI ASSIST-AI against representative prior approaches along five dimensions critical for academic deployment.

Table 3 Approach Comparison Across Key Dimensions

System	Domain	No Halluc.	Multimodal	Scalable	Live Data
Rule-Based	✓	✓	×	×	×
BERT Classif.	✓	✓	×	✓	×
Vanilla LLM	×	×	Partial	✓	×
Ours	✓	✓	✓	✓	✓

VII. CONCLUSION

This paper presented UNI ASSIST-AI, a RAG-powered intelligent chatbot purpose-built for university environments. By coupling a semantic retrieval engine with a GPT-based language model and extending the system with ASR and OCR pipelines, we achieve high factual accuracy (F1 = 0.85) while supporting diverse input modalities. The system outperforms rule-based, ML-classifier, and vanilla LLM baselines on all three evaluation metrics and demonstrates measurable reduction in hallucinated responses. The React + FastAPI architecture makes it straightforward to deploy at any institution by simply re- ingesting that institution's document corpus.

UNI ASSIST-AI validates that domain-grounded RAG is the right architectural choice for mission-critical conversational AI in academic settings, where factual precision is non- negotiable.

Future Work

- **Knowledge Graph Integration:** Representing entity relationships (courses → prerequisites → faculty) as a graph to support multi-hop reasoning [10].
- **Multilingual Support:** Extending retrieval and generation to Hindi and other regional languages to serve linguistically diverse student populations.
- **Personalization:** Maintaining a per-student session memory so the assistant can tailor responses to the student's year, branch, and prior queries.
- **Mobile Deployment:** Packaging the system as a Progressive Web App (PWA) for ofline-capable mobile access.
- **Continuous Learning:** Implementing an active-learning loop where low-confidence responses are flagged for expert review and fed back into the corpus.

REFERENCES

1. D. Baidoo-Anu and L. O. Ansah, "Education in the era of generative artificial intelligence (AI): Understanding the potential benefits of ChatGPT

in promoting teaching and learning," Journal of AI, vol. 7, no. 1, pp. 52–62, 2023.
2. P. Lewis, E. Perez, A. Piktus et al., "Retrieval-Augmented Generation for knowledge-intensive NLP tasks," in Proc. Advances in Neural Information Processing Systems (NeurIPS), 2020, pp. 9459–9474.
3. T. Brown, B. Mann, N. Ryder et al., "Language models are few-shot learners," in Proc. NeurIPS, 2020, pp. 1877–1901.
4. H. Liu, C. Li, Q. Wu, and Y. J. Lee, "Visual instruction tuning (LLaVA)," in Proc. NeurIPS, 2023.
5. J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.
6. J. Baig, S. Akhtar, and M. Khan, "Conversational AI chatbots in higher education: Challenges and opportunities," Computers & Education: Artificial Intelligence, vol. 7, 2025.
7. A. Ramesh, P. Sharma, and K. Joshi, "Domain-adaptive retrievalaugmented generation for enterprise knowledge bases," in Proc. ACL Findings, 2024, pp. 301–315.
8. N. Patel and R. Mehta, "Multimodal AI systems for document-centric query answering," IEEE Access, vol. 13, pp. 22401–22418, 2025.
9. J. Johnson, M. Douze, and H. Jegou, "Billion-scale similarity search´ with GPUs," IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535–547, 2019.
10. Q. Tang, Z. Wang, and L. Chen, "Graph-enhanced retrieval-augmented generation for multi-hop question answering," in Proc. WWW, 2025, pp. 1120–1131.
11. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-networks," in Proc. EMNLP, 2019, pp. 3982–3992.
12. A. Radford, J. W. Kim, T. Xu et al., "Robust speech recognition via large- scale weak supervision (Whisper)," in Proc. ICML, 2023, pp. 28492–28518.