SJIF Impact Factor (2025): 8.688| ISI I.F. Value: 1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 10 | Issue: 8 | August 2025 - Peer Reviewed Journal

A REVIEW ON RAG-BASED STUDENT ASSISTANT CHATBOT USING LANG CHAIN

Dharshan S[1], Puneeth KS[2], Sanjay G J[3], Vivek V[4], Dr Anitha DB[5]
[1]*Dept of CSE-Data Science, ATME College Engineering, Mysuru*
[2]*Dept of CSE-Data Science, ATME College Engineering, Mysuru*
[3]*Dept of CSE-Data Science, ATME College Engineering, Mysuru*
[4]*Dept of CSE-Data Science, ATME College Engineering, Mysuru*
[5]*Associate Professor &Head of Dept, Dept of CSE-Data Science, ATME College Engineering, Mysuru*

Article DOI: https://doi.org/10.36713/epra23698
DOI No: 10.36713/epra23698

ABSTRACT
In the evolving landscape of educational technology, students face ongoing challenges in accessing timely and accurate academic information. Traditional query resolution systems-ranging from FAQs to administrative desks-are often inefficient. This review paper explores the application of Retrieval-Augmented Generation (RAG) and the LangChain framework to build intelligent, responsive, and domain-specific chatbots for academic institutions. Through the integration of a vector database, retriever modules, and large language models (LLMs), RAG-based systems ensure contextual relevance and data-grounded responses. This paper surveys existing literature, evaluates methodologies, and highlights the significance, implementation strategies, and expected outcomes of such systems in the educational sector.

KEYWORDS-*Retrieval-Augmented Generation (RAG), LangChain, Vector Database, LLM*

I.INTRODUCTION

As educational institutions and enterprises increasingly adopt AI-driven tools, the need for accurate and contextually rich conversational agents has become more apparent. Simple rule-based or generative-only chatbots often fall short when tasked with delivering factual and nuanced information grounded in institutional knowledge.

Retrieval-Augmented Generation (RAG) has emerged as a hybrid solution, combining the strengths of information retrieval and generative modeling. Paired with frameworks like LangChain, which facilitate modular pipeline construction and seamless integration with vector databases, RAG-based systems have demonstrated significant improvements in user interaction quality, response accuracy, and domain adaptability.

This review paper compiles and analyzes key findings from recent literature that apply RAG-based chatbots in educational and technical domains, offering insights into design choices, implementation outcomes, and limitations.

II.LITERATURE SURVEY

To systematically analyse the existing body of work on Retrieval-Augmented Generation (RAG)-based student chatbots using LangChain, the reviewed papers have been grouped into thematic categories. Each category represents a distinct dimension of chatbot development and deployment, such as educational applications, system architecture, local LLM deployment, retrieval strategies, and evaluation methodologies. This classification helps highlight specific trends, common challenges, and unique innovations across different implementations. By organizing the literature in this manner, we aim to provide clearer insights into how RAG and LangChain are being utilized to enhance educational support systems and address domain-specific requirements.All papers are classifying into three ways:

a. LangChain-Based RAG Chatbots
b. Intelligent Educational Assistants
c. Novel Extensions and Architectures

A. LangChain-Based RAG Chatbots

This group of papers focuses on chatbots built using the LangChain framework. They highlight how LangChain connects LLMs with document retrievers, embeddings, and vector stores. These systems mainly aim to improve chatbot architecture, modularity, and retrieval performance.

SJIF Impact Factor (2025): 8.688| ISI I.F. Value: 1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 10 | Issue: 8 | August 2025 - Peer Reviewed Journal

Zhang et al. proposed a LangChain-powered chatbot that connects LLMs with locally stored documents such as PDFs and video subtitles. The system uses FAISS for similarity-based document retrieval and OpenAI embeddings for vectorization. A memory mechanism maintains conversational context, and test question generation is supported from chat history. The interface is developed using Gradio for ease of use. One limitation is reliance on cloud-hosted LLMs, which poses privacy concerns. The authors plan to transition to local LLMs to enhance data security and reduce external dependencies [1].

Kumar et al. developed a student assistance platform utilizing LLMs locally deployed via Ollama and Meta's 7B models. It integrates FAISS/Milvus for embedding storage and uses Flask as the backend to serve responses from a LangChain pipeline. A custom web UI provides an interactive interface. The use of open-source components helps reduce vendor lock-in.
Their Retrieval-Augmented Generation (RAG) model effectively answers student questions using embedded course materials. The platform promotes privacy by avoiding dependence on third-party APIs. It is especially suited for academic institutions with data privacy constraints [2].

B. Intelligent Educational Assistants
These studies center on chatbots made for students and faculty in academic settings. They are designed to answer syllabus, timetable, and policy-related queries using college documents. The goal is to improve access to institutional information and reduce manual support.

Richard et al. introduced a client-server-based educational chatbot that combines LLMs with trusted academic documents stored in vector form. The system uses MiniLLM for efficient inference and FAISS for vector search. JWT-based authentication ensures secure, stateless sessions, while FastAPI handles scalability. They implemented the Mixtral 8x7B Mixture of Experts model, which optimizes inference by activating only relevant sub-models. The chatbot is capable of handling multiple users simultaneously. It supports various formats like PDFs and lecture slides and aims to provide contextually accurate answers [3].

Wijaya and Purwarianti designed a QA system embedded in a web-based intelligent tutoring platform aimed at enhancing programming education. The architecture combines LangChain, Redis memory for chat history, and a vector store for semantic retrieval. They use multiple LLMs such as Llama3 and Code Gemma and apply CO-STAR prompt engineering to guide response tone and structure. Their evaluation covers faithfulness, helpfulness, and friendliness, with Llama3 performing best. The system supports both single-turn and multi-turn interactions. It showcases the impact of combining memory, prompt strategy, and RAG for educational QA [4].

Calfoforo et al. built a document-based QA chatbot to serve university staff by answering policy-related questions using handbooks and internal documents. Their architecture includes FAISS, PDF parsing, and embedding with Sentence Transformers (all-mpnet-base-v2). They employ QLoRA for low-resource fine-tuning of LLMs, achieving high performance at lower computational costs. The system supports chat history, conversational queries, and returns source document references for transparency. It outperforms manual lookup in speed and accuracy. This work underlines how RAG models can significantly reduce administrative workload in universities [5].

Ananya and Vanishree implemented a PDF-based chatbot that supports question answering through RAG with models like Llama2 and GPT-3.5. The pipeline uses LangChain, Hugging Face Transformers, and PyTorch. Texts are preprocessed, chunked, embedded, and stored in FAISS for semantic retrieval. The chatbot uses Chainlit for an interactive frontend, enabling real-time response generation. Reinforcement learning is used to manage token costs and optimize model performance. The system shows high accuracy in resolving academic queries. It is designed for educational and corporate knowledge access [6].

C. Novel Extensions and Architectures
This section covers advanced chatbot designs that go beyond standard RAG setups. These include models with web scraping, knowledge graphs, QLoRA fine-tuning, and improved evaluation tools. They showcase innovations in real-time data use and performance enhancement.

Sriram et al. introduced ScrapeTalk, a chatbot that fetches real-time data from the web using Beautiful Soup, ChromaDB, and WebBaseLoader. Built on LangChain and Stream lit, the chatbot uses OpenAI's GPT-3.5 Turbo for generation. It can parse and index web content dynamically, allowing it to provide up-to-date answers. The system focuses on replacing manual web searches with intelligent conversation. Streamlit ensures an intuitive user experience. ScrapeTalk shows strong potential in reducing research effort and enhancing answer relevance through live data integration [7].

Gijre et al. combined RAG with knowledge graphs to improve question answering accuracy in long-form, domain-specific documents. Using Google Vertex AI for embedding and Neo4j for graph storage, they create nodes with relationships such as "PART-OF" and "NEXT." LangChain is used for processing, and queries retrieve relevant graph sections based on semantic similarity.

SJIF Impact Factor (2025): 8.688| ISI I.F. Value: 1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 10 | Issue: 8 | August 2025 - Peer Reviewed Journal

Evaluations using the RAGAS framework show high scores in faithfulness and relevancy. The approach excels at preserving context across large document sets. Their system is suited for complex domains like law and education [8].

Jaiswal et al. designed a chatbot that can ingest and process multiple PDF documents for contextual Q&A. They use PyPDF2 for text extraction, chunk the data using RecursiveCharacterTextSplitter, and embed with Gemini 1.0 Pro. The system is hosted on Streamlit and supports persistent chat history. It addresses prior limitations of poor multi-document handling and offers a scalable, extensible solution. Their focus is on improving information accessibility for both academic and business documents. Results show high responsiveness and semantic accuracy [9].

Swacha and Gracel conducted a meta-survey of 47 research papers on RAG-based chatbots in educational settings. They categorized implementations by function: academic help, administrative assistance, and domain learning. Most systems targeted students, followed by faculty and support staff. Evaluation tools like RAGAS are mentioned but underutilized. The authors emphasize the lack of performance studies tied to actual educational outcomes. Their paper maps out research directions and calls for more rigorous, goal-specific evaluations. It serves as a foundational guide for future research [10].

Zhang et al. explored RAG optimization for PDF-based chatbots in the automotive industry, but their evaluation methods and insights are applicable to education. They highlight that LLMs may produce correct responses without truly understanding prompts. Through logical reasoning tests, they expose gaps in prompt adherence and response consistency. The authors call for evaluation frameworks that go beyond surface-level accuracy. Their findings urge caution when deploying LLMs in high-stakes environments like education. The study advocates deeper interpretability in LLM-based systems [11].

Neupane et al. developed a chatbot designed to help university students access campus resources like course registration, administrative support, and facilities. The system uses domain-specific knowledge bases and ranking techniques to deliver context-aware responses. It supports natural dialogue flow and follow-up queries. Real-world testing revealed high user satisfaction and efficiency. The system is modular, allowing easy updates with new data. Future improvements include mobile app integration and multilingual support to expand reach [12].

Antico et al. proposed “Unimib Assistant,” a student-centric RAG chatbot built to address university-level queries. It integrates NLP and retrieval models to handle ambiguous or partial queries using follow-up clarification. The system connects to real-time databases for updated content. It is designed with scalability and modularity in mind. Their pilot implementation showed a positive impact on both student and administrative experiences. Planned upgrades include personalization, multilingualism, and intelligent push notifications for student engagement [13].

Gehrmann et al. from Microsoft analyzed the entire RAG pipeline, from retrieval and chunking to grounding and generation. They explored dense vs sparse retrievers, chunking strategies, and prompt design. The authors revealed how weaknesses in one component can cascade, degrading final response quality. Feedback loops in retrieval can amplify hallucinations if not managed properly. They introduced new evaluation metrics for factual consistency and completeness. Their benchmark, FACTS, is proposed as a new gold standard for evaluating RAG-based chatbots [14].

III.OUTCOME OF LITERTURE REVIEW

The literature reviewed reflects a rapidly growing interest in the application of Retrieval Augmented Generation (RAG) combined with Large Language Models (LLMs) for developing intelligent, context-aware chatbot systems. A common trend across the works is the adoption of LangChain as a versatile orchestration framework that seamlessly integrates retrieval, generation, memory, and external tools, making it central to most implementations. Educational environments emerge as a particularly prominent domain of application for RAG based chatbots. Numerous studies demonstrated how RAG improves contextual accuracy, reduces hallucinations, and bridges the gap between static knowledge bases and conversational systems. These chatbots provide students and faculty with streamlined access to academic resources, ranging from course materials to institutional policies. Many systems incorporated PDF document retrieval, video-based learning content extraction, and integration with institutional data repositories, showcasing the versatility of the approach.

IV.HOW RAG WORKS WITH LANG CHAIN

Here’s the common pipeline used in Lang Chain for a RAG-based chatbot:

a. User Query Input: User query input process begins when a user inputs a query or a question into the system. This query serves as the starting point that initiates the entire retrieval-augmented generation (RAG) workflow.
b. Query Embedding: Next, Query Embedding is raw text query is passed to an embedding model, often based on a pretrained transformer architecture. The embedding model transforms the query into a dense vector representation within a semantic space. This vector encapsulates the meaning of the query, making it suitable for similarity search against stored knowledge.
c. Retriever:VectorSearch: The query embedding is then sent to the Retriever component. The Retriever performs a similarity search within a vector database, which stores precomputed vector embeddings of documents, passages, or knowledge chunks.

SJIF Impact Factor (2025): 8.688| ISI I.F. Value: 1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 10 | Issue: 8 | August 2025 - Peer Reviewed Journal

Using cosine similarity or another distance metric, the Retriever identifies and returns the top-k documents that are semantically closest to the user's query.

d. Documents Returned: Retriever system collects these top relevant documents or text chunks retrieved by the Retriever. These documents contain information that is most related to the user's query and serve as the contextual knowledge for the subsequent generation step.
e. Prompt Construction: After retrieving the documents, the system constructs a prompt by combining the user's original query with the retrieved documents. This usually involves concatenating the retrieved texts with additional instructions or templates to guide the language model response.
f. Language Model (LLM) Generation: Constructed prompt is then provided to the Language Model. The LLM generates a response by leveraging both the retrieved documents and the original query. The presence of external knowledge from the retrieved documents allows the LLM to produce more accurate, relevant, and factually grounded answers.
g. Generated Answer Returned: Finally, the generated answer from the LLM is presented to the user as the system's response. This response is designed to be relevant, context aware, and informative, providing the user with a high-quality answer based on both the LLM's capabilities and the retrieved knowledge.

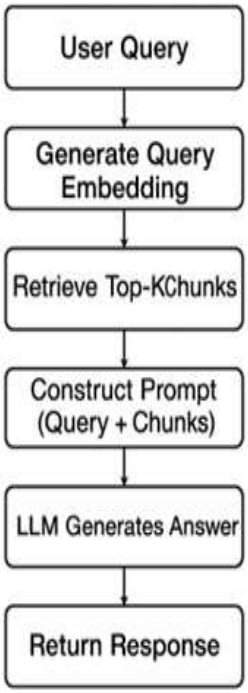

Fig 1 RAG With Lang chain Pipeline

V. SUMMARY AND RESEARCH GAPS

These studies demonstrate that RAG-based chatbots using LangChain significantly enhance information retrieval, contextual accuracy, and user satisfaction in educational environments. Key components across papers include vector search, local LLMs, real-time UIs and memory persistence

.

However, gaps remain in:

a. Explainability of model outputs
b. Handling ambiguous or multilingual queries
c. Real-time multimodal data integration
d. Long-term user context and personalization

Future research should address these limitations while exploring domain-specific fine-tuning, reinforcement learning, integration and mobile deployment for accessibility.

VI. LIMITATIONS AND CHALLENGES

Here the Limitations and Challeges of RAG-Based Student Chatbot using Langchain
LLM Hallucination: Despite context grounding, LLMs may generate inaccurate content.
Latency:Large models increase inference time, affecting real-time interaction.
Data Privacy:Sensitive institutional documents require secure storage and access protocols.

SJIF Impact Factor (2025): 8.688| ISI I.F. Value: 1.241| Journal DOI: 10.36713/epra2016 ISSN: 2455-7838(Online)
EPRA International Journal of Research and Development (IJRD)
Volume: 10 | Issue: 8 | August 2025 - Peer Reviewed Journal

Multilingual and Multimodal Gaps: Most chatbots are limited to English and text-based inputs with the Multilingual.

VII.CONCLUSION

The literature reviewed demonstrates that RAG-based chatbots using LangChain offer a scalable and reliable solution to information access challenges in educational and technical domains. By integrating structured retrieval with the generative capabilities of LLMs, these systems bridge the gap between static documentation and dynamic user interaction.

Future research directions include enhancing explainability, integrating multimodal data, supporting multilingual communication, and enabling offline local deployments for privacy and resilience. The surveyed work collectively represents a significant step toward building context-aware, trustworthy, and user-centric chatbot systems.

Acknowledgment: We are deeply thankful to our ATME college of Engineering,Mysuru for providing the necessary resources and a supportive environment to carry out our research. We sincerely appreciate the guidance and encouragement from our teachers throughout the process. We also extend our gratitude to our friends for their constant support and motivation during the course of this work.

REFERENCES

1. *H. Zhang, Z. Li, F. Liu, Y. He, Z. Cao, and Y. Zheng, "Design and Implementation of LangChain-based Chatbot," in Proc. 2024 Int. Seminar on Artificial Intelligence, Computer Technology and Control Engineering (ACTCE)-2024.*
2. *Kumar P, Harish M et al Department of CSE – "Development of Interactive Assistance for Academic Preparation Using Large Language Models",Rajalakshmi Engineering College Chennai, India.*
3. *Rohan Paul Richard, Ebenezer Veemaraj , et al –"A Client-Server Based Educational Chatbot for Academic Institutions", Karunya Institute of Technology and Sciences Coimbatore, India*
4. *Owen Christian Wijayaand and Ayu Purwarianti-"An Interactive Question-Answering System using Large Language Model and Retrieval-Augmented Generation in An Intelligent Tutoring System on the Programming Domain",School of Electrical Engineering and Informatics Bandung Institute of Technology Bandung, Indonesia.*
5. *Josie C. Calfoforo and Dr. Rodolfo C. Raga, Jr. – "Unleashing AI in Education: A Pre-Trained LLMs for Accurate and Efficient Question-Answering Systems", National University Manila, Philippines.*
6. *[6] Ananya G & Dr. Vanishree K -" RAG based Chatbot using LLMs", Department of ISE R V College of Engineering Bengaluru, India*
7. *Sri Ram G,Rahul S et al-"SCRAPETALK: CHATBOT CONVERSATION WITH WEB SCRAPED INSIGHTS" School of Computer Science and Engineering Vellore Institute of Technology Vellore, India.*
8. *Supraj Gijre, Rishi Agrawal and et al –"Enhancing Question-Answering with Knowledge Graph Retrieval and Generation using LLMs" School of Computer Science and Engineering Ramdeobaba University Nagpur, India.*
9. *Anuj Jaiswal, Garima Tiwari and et al-" Retrieval Augmented Generation Approach for Multipdf Chatbot using LangChain", Computer Department G H Raisoni College of Engineering & Management, Pune ,India.*
10. *Mandar Kulkarni,Praveen Tangarajan,kyung Kim–"Reinforcement Learning for Optimizing RAG for Domain Chatbots",Flipkart Data Science,Seattle,Washington,USA.*
11. *Fel Liu,Zejun Kang and Xing Han –"Optimizing RAG techniques for Automotive industry PDF Chatbots: A case Study With Locally Depolyed ollama Models",china Automotive Technology & Research center.*
12. *Subash Neupane, Elias Hossain,Jason Keith et al – "From Questions to Insightfull Answers:Building an informed Chatbot for University Resources",Department of computer science & Engineering, Mississippi State University.*
13. *Chiara Antico,stefano Giordano,Cansu koyuturk-"Unimib Assistant: Designing a student-friendly RAG-Based chatbot for all their needs", Milan, Italy.*
14. *Rama Akkiraju,Anbang Xu, Deepak Bora et al –"FACTS About Building Retrieval Augmented Generation – Based Chatbots",NVIDIA-2024.*