ISSN: 2456-8503

International Journal for Development of Science and Technology (IJDST)

Educational rag system for specialized textbook

RESEARCH

Muthupandi[1], *Praveen*[1], *Praveen*[1], *Mahalakshmi*[1*], *Senthil Prakash*[1*]

Abstract

Large Language Models (LLMs) have shown strong abilities in understanding and generating natural language, but they often produce inaccurate or misleading responses when applied to specialized educational domains. This paper introduces an Educational Retrieval-Augmented Generation (RAG) system tailored for textbook-based learning, where accuracy is critical. Instead of depending completely on the model's internal knowledge, the method recovers appropriate knowledge from a curated collection of domain-specific textbooks. It utilizes a fusion recovery methodology that links keyword matching with semantic similarity to detect the most appropriate content, which acts as context for a locally deployed LLM to make precise answers. Developed as a web-based platform using frameworks like Django, the system minimizes dependency on external APIs and includes a preprocessing mechanism to convert visual elements like diagrams into descriptive text. Overall, the system provides accurate, explainable, and domain-focused educational support.

Keywords: *Retrieval-augmented generation, large language models, hybrid retrieval, semantic search, hallucination reduction.*

1. Introduction

In today's digital learning environment, Artificial Intelligence has notably improved in what manner learners engage with scholastic reserves. However, while transformer-based LLMs are highly effective in generating natural language, they often struggle with specialized educational material and produce incorrect information.

Commonly known as hallucinations such inaccuracies are problematic in academic environments where correctness and trustworthiness are essential. Traditional keyword-based search systems also fall short as they lack semantic understanding and fail to interpret the context of user questions. To deliver these tasks, this document offers an Educational RAG system that maintains a structured repository of carefully selected textbooks as the primary knowledge source. By ensuring all created outputs are supported by authoritative academic sources, the system functions as a dependable digital academic assistant. Recent advancements in transformer-based architecture have notably better natural language processing potentials [1].

[1]*Department of Computer Science and Engineering, Shree Venkateswara Hi-Tech Engineering College (Autonomous), Tamilnadu, India.*

[2]*Department of Computer Science and Engineering, Shree Venkateswara Hi-Tech Engineering College (Autonomous), Tamilnadu, India.*

[3]*Department of Computer Science and Engineering, Shree Venkateswara Hi-Tech Engineering College (Autonomous), Tamilnadu, India.*

[4]*Assistant Professor, Department of Computer Science and Engineering, Shree Venkateswara Hi-Tech Engineering College (Autonomous), Tamilnadu, India.*

[5]*Professor, Head of the Department, Department of Computer Science and Engineering, Shree Venkateswara Hi-Tech Engineering College (Autonomous), Tamilnadu, India.*

**Corresponding Author: jtysp14@gmail.com*

2. Background and related work

2.1. Retrieval-Augmented Generation (RAG)

Previous research introduced the RAG framework to combine retrieval methods with generative models for knowledge-intensive tasks [2]. While this notably reduces factual errors, the methodology depends heavily on retrieval quality and does not specifically deliver domain-restricted environments like textbook based learning.

2.2. Large Language Models and Hallucinations

Studies on Large language models have proved strong natural language understanding potential, but they often produce hallucinated outputs in domain-specific applications [3]. Making standalone conversational AI systems less fit for academic systems requiring high accuracy.

2.3. Information Retrieval Techniques

Transformer-based models for example BERT improve deep bidirectional language understanding for answering questions [4]. Additionally, Dense Passage Retrieval (DPR) utilizes vector vectors to retrieve information based on semantic similarity. But these are often constructed for open-domain data rather than verified academic sources.

3. Proposed system architecture

The intended system results in a web-based client-server model utilizing a Service-Oriented Architecture. Instead of generating an answer directly from pre-trained knowledge, the approach recovers the most appropriate textbook content using a hybrid retrieval mechanism combining keyword matching and semantic similarity search.

Table 1: Key components of the intended architecture

Component	Function	Implementation	Benefit
Collective Learning Pipeline	Ingests new textbooks, extracts text, performs cleaning, chunking, and prepares data for processing	Text extraction tools (PyPDF), preprocessing methods, and text segmentation methods.	Ensures continuous knowledge expansion and scalability of the system.
Embedding Module	Converts textual data into numerical vector representations for semantic understanding.	Transformer-based embedding models (Sentence Transformers	Captures contextual meaning beyond exact keyword matching.
Vector Database	Stores and indexes embedding vectors for efficient retrieval.	ChromaDB / FAISS vector storage systems.	Enables fast and accurate similarity-based search.
RAG Module	Performs hybrid retrieval (keyword semantic) and constructs context-aware prompts.	LangChain framework integrating retrieval pipelines.	Ensures responses are grounded in appropriate textbook content.
LLM Conversational System	Generates final responses strictly based on retrieved context.	Locally deployed Large Language Model (LLM).	Reduces hallucinat ion, improves accuracy, and eliminates dependency on external APIs.

(Table 1) presents the essential factors of the Educational RAG system architecture including functions, implementation methods, and benefits. The Collective Learning Pipeline is responsible for ingesting and preprocessing textbook data, ensuring scalability. The embedding module transforms written facts into mathematical vectors to capture semantic meaning. The vector database enables efficient similarity-based retrieval of appropriate content. The RAG module integrates retrieval and prompt construction, ensuring context-aware responses. Finally, the LLM conversational system creates accurate answers strictly based on retrieved textbook content, thereby reducing hallucination and improving reliability.

- *Collective Learning Pipeline (The Data Gatherer):* This is the entry point for new information. When a new textbook is uploaded, this pipeline extracts the raw text, cleans it up, and chops it into smaller, logical segments called chunks. This chunking is necessary because language models can only process a limited volume of passage at one time, and it ensures easy scalability.
- *Embedding Module (The Translator):* Computers don't inherently understand human language; they understand math. This module uses transformer models (like sentence transformers) to convert the text chunks into mathematical representations called numerical vectors. Word embedding methods enable numerical representation of text for similarity comparison [5]. Sentence embedding methods for example sentence-BERT enable efficient semantic similarity computation [6]. This is an essential step because it captures the deep relative significance of the passage by not only matching instead of allowing the system to understand the concepts.
- *Vector Database (The Smart Filing Cabinet):* Once the text is transformed into numbers, it needs specialized storage. A vector catalog (for example ChromaDB or FAISS) organizes and indexes these numerical vectors. Because of how vectors work in multi-dimensional space, this database can perform highly efficient similarity searches to instantly find the textbook chunks that are conceptually closest to a user's question.
- *RAG Module (The Orchestrator):* Powered by frameworks like lang chain, this module acts as the traffic controller. When a user asks a question, the RAG module runs a hybrid search to find the best textbook chunks from the vector database. Keyword-based retrieval is an implemented using BM25 ranking algorithm [6]. This ensures accurate matching of exact terms. In parallel, semantic similarity search is performed using vector-based retrieval methods [7]. To identify contextually appropriate information it then bundles the user's question and those retrieved facts together to construct an augmented, context-rich prompt.
- *LLM Conversational System (The Speaker):* This is the in-house installation of Large Language Model that actually talks to the user. Recent large language models for example LLM have proved strong potential in generating context-aware responses [8]. Instead of trying to answer the question from its own pre-trained memory, which can lead to false information, this system reads the prompt prepared by the RAG module and generates an answer strictly based on that verified textbook material. Because it is deployed locally, it keeps institutional data secure and doesn't rely on third-party internet APIs.

(Table 2) provides a comparative analysis between the proposed educational RAG system, traditional keyword-based search systems, and standalone LLM chatbots. The comparison highlights key performance factors for example context understanding, source reliability, hallucination risk, traceability, response accuracy, and data privacy. The proposed system outperforms conventional methodology by combining semantic search with verified textbook sources, resulting in highly accurate and context-aware responses. Additionally, it notably reduces hallucination by restricting the language model to retrieved content and ensures better traceability by providing source references. The system also enhances data privacy through local deployment, commanding it further appropriate for academic environments.

Table 2: Comparison with conventional systems

Feature	Proposed Educational RAG System	Traditional Keyword Search	Standalone LLM Chatbots
Context Understanding	High (Semantic similarity-based understand ing of queries)	Low (Exact keyword matching only)	High (Natural language understanding)
Source Reliability	High (Uses verified and curated textbooks)	Medium (Mixed or unverified sources)	Low (Based on pre-trained internal data)
Hallucination Risk	Very Low (Responses grounded in retrieved content)	None (Does not generate new content)	High (May generate incorrect information)
Traceability	High (Provides references from textbooks)	Low (Manual search required)	Low (No proper citations provided)
Response Accuracy	High (Context-aware and domain-specific)	Medium (Depends on keyword match)	Medium to High (But may unreliable)
Data Privacy	High (Local deployment, no external API dependenc y)	Medium (Depends on platform used)	Low (Often relies on cloud-based APIs)

3.1. Context Understanding

- *Traditional Keyword Search (Low):* Standard search engines look for exact matching words or phrases. If you search for a concept using slightly different terminology than what is in the book, a keyword search might fail because it doesn't understand the significance behind your query.
- *Standalone LLM Chatbots (High):* Generic AI chatbots are excellent at understanding natural language and the intent behind your questions.
- *Educational RAG System (High):* The RAG system gets the best of both worlds by using semantic similarity. It translates text into numerical vectors to understand the context and significance of a query. Allowing it to find appropriate textbook sections even if you use different words.

3.2. Source Reliability

- *Traditional Keyword Search (Mixed/Unverified):* Searching a digital library or the open internet often returns a massive volume of unstructured data, leaving the student to find which traces are accurate or appropriate to their specific syllabus.
- *Standalone LLM Chatbots (Pre-trained internal data):* Standard chatbots generate answers built on a massive, hidden pool of data they were trained earlier. They don't retrieve information from verified academic sources in real-time, meaning their answers might not align with a specific textbook's curriculum.
- *Educational RAG System (Verified Textbooks):* This system restricts its knowledge base exclusively to a curated repository of approved, domain-specific textbooks.

3.3. Hallucination Risk

- *Standalone LLM Chatbots (High):* Because they rely on pre-trained data, generic LLMs often hallucinate meaning they confidently invent incorrect or misleading information when they don't actually know the answer.
- *Traditional Keyword Search (N/A):* Search engines don't write new text; they just return existing documents, so they cannot hallucinate.
- *Educational RAG System (Very Low):* The RAG system solves the AI hallucination problem by mathematically restricting the language model. It is programmed to only generate responses using the verified textbook chunks it just retrieved, notably reducing the chance of it making things up.

3.4. Traceability

- *Standalone LLM Chatbots (Low):* Standard AIs rarely provide accurate citations, making it incredibly difficult for a student to verify if the information is correct.
- *Traditional Keyword Search (Low):* While you realize which article connected on, you still need to manually read through the entire text to find the specific answer.
- *Educational RAG System (High):* The RAG system acts as an explainable AI. Whenever it generates an answer, it provides the exact source information (like the textbook name and section) so the student can easily trace the fact back to its origin and verify its authenticity.

4. Implementation methodology

The application of the Educational RAG system used a modular methodology with Python-based technologies. The base structure utilizes Django for backend routing and secure user authentication.

- *Preprocessing & Embedding:* Textual content is extracted using PyPDF, cleaned, and divided into chunks. Each portion is transformed into strong vectors by Sentence Transformers and collected in ChromaDB.
- *Hybrid Retrieval:* Client questions are transformed into vectors, and a dual retrieval process initiates a BM25 keyword-based search for exact matches, and a FAISS semantic similarity search. The system uses FAISS and embedding-based retrieval methods for effective resemblance seek [9].
- *Response Generation:* Retrieved segments are combined with the user query to form an augmented prompt a locally deployed LLM processes this prompt, ensuring the generated response relies solely on the provided context.

5. Results and discussion

The system was rigorously tested using Unit Testing for individual modules (APIs, embedding functions), integration testing for component interactions, and functional testing. Real-world academic questions, for example explain Ohm's Law with an example and define stack and queue in data structures, were used during functional evaluation. The response generated was compared against original textbook explanations, proving high accuracy, relevance, and clarity performance metrics indicated stable query processing and similarity search latency under normal usage conditions [10].

6. Challenges and future scope

While the system successfully recovers and generates accurate textbook information, it currently processes each query independently without retaining previous interactions. Furthermore, the system is primarily restricted to English-based textual content. Future enhancements will focus on integrating Multi-Turn conversational memory to handle follow-up questions and maintain context across a session. Additionally, introducing Multilingual Support will allow for cross-language semantic retrieval. To deliver subjects heavily reliant on graphs and charts, future iterations will explore Multi-Modal retrieval, utilizing image vectors to automatically interpret diagrams.

7. Conclusion

The educational RAG System for specialized textbooks establishes an effective application of artificial intelligence in academic knowledge retrieval. By shifting away from standalone pretrained knowledge and integrating a hybrid semantic search mechanism, the system successfully grounds its language model generation academic sources. This architecture provides a reliable, scalable, and explainable platform that reduces misleading outputs and enhances the digital learning experience for students.

Conflict of interest statement: The authors declare that there is no conflict of interest regarding the publication of this research paper.

Funding information: This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Data availability statement: The data used in this study were generated during the testing phases of the project architecture and are available upon request.

Ethical approval statement: This study does not involve human participants, animals, or any sensitive personal data. The research is based on system design, implementation, and evaluation using publicly available and academic textbook data. Therefore, ethical approval was not required for this study.

Acknowledgement: The authors sincerely thank the Management and the Department of Computer Science and Engineering at Shree Venkateshwara Hi-Tech Engineering College, Gobi, for providing the academic support, guidance, and a conducive environment for the completion of this project.

References

1. Lewis P, Perez E, Piktus A, Retrieval-augmented generation for knowledge-intensive NLP tasks, Advances in Neural Information Processing Systems. 2020, 33(1):9459-9474. doi: 10.5555/3495724.3496517
2. Devlin J, Chang M, Lee K, Toutanova K, BERT: Pre-training of deep bidirectional transformers for language understanding, Proceedings of NAACL-HLT. 2019, 4171-4186, doi: 10.18653/v1/N19-1423
3. Karpukhin V, Oguz B, Min S, Dense passage retrieval for open-domain question answering, Proceedings of EMNLP. 2020, 6769-6781. doi: 10.18653/v1/2020.emnlp-main.550
4. Brown T, Mann B, Ryder N, Language models are few-shot learners, arXiv preprint arXiv:2005.14165. 2020. doi: 10.48550/arXiv.2 005.14165
5. Touvron H, Martin L, Stone K, LLaMA: Open and efficient foundation language models, arXiv preprint arXiv: 2302.13971. 2023. doi: 10.48550/ar Xiv.2302.139
6. Robertson S, Zaragoza H, The probabilistic relevance framework: BM25 and beyond, Foundations and Trends in Information Retrieval. 2009, 3(4):333-389. doi: 10.1561/1500000019
7. Johnson J, Douze M, Jegou H, Billion-scale similarity search with GPUs, IEEE Transactions on Big Data. 2021, 7(3):535-547. doi: 10.1109/bda ta.2019.2921573
8. Reimers N, Gurevych I, Sentence-BERT: Sentence embeddings using Siamese BERT-Networks, Proceedings of EMNLP. 2019, 3982-3992. doi: 10.18653/v1/d19-1410
9. Mikolov T, Chen K, Corrado G, Dean J, Efficient estimation of word representations in vector space, arXivpreprint arXiv:1301.3781. doi: 10.48550/arXiv.1301.3781
10. Vaswani A, Shazeer N, Parmar N, Attention is all you need, arXiv preprint arXiv:1706.03762. 2017. doi: 10.48550/arXiv.1706.03762