



Hybrid RAG-LLM System for Student Data Retrieval and Summarisation

Dr. G. Kishor Kumar¹ M. Tech, Ph.D., B. Kalpana², P. Murali Krishna³, B. Kiran⁴

Professor & Head, Department of CSE & Business Systems, RGM College of Engineering and Technology, Nandyal, Andhra Pradesh, India¹

B. Tech Students, Department of CSE & Business Systems, RGM College of Engineering and Technology, Nandyal, Andhra Pradesh, India²⁻⁴

Publication History: Received: 25.02.2026; Revised: 20.03.2026; Accepted: 25.03.2026; Published: 28.03.2026.

ABSTRACT: Schools bring out a lot of academic records as reports of the students, their end-semester marks and course-specific materials. In some cases, certain information in such documents can only be accessed through manual searches. This may be tedious and less efficient, especially when dealing with multiple students' records. Thus, the current study suggests the Student Information Retrieval and Summarization System with Large Language Models and Retrieval-Augmented Generation [1, 2]. The data utilized in this study will include the academic records of CMMS students of approximately 50 students in the same grade. Student ID, name of students, the name of the branch, name of semester, name of subject, name of subject code, grade, and student SGPA are some of the information contained in the dataset. The system begins by removing the text in the PDF file followed by pre-processing the data.

A document chunking means that the text is further broken down into smaller fragments. This approach enhances precision of the information retrieved. Next, the chunks are converted to a vector embedding using the embedding model. This model helps in understanding the meaning of the text. Storing these vectors in a FAISS Vector Database [4,14] enables the process to be more efficient in retrieving the relevant information. When the user makes a query in natural language, the system isolates the most suitable bits of the document which has information regarding the query made by the user. These context fragments are then fed to a Large Language Model coupled with LangChain and trained on Ollama [11,12] which forms just the right answers to what the user is asking. Therefore, the system suggested enhances efficiency and precision of recalling the information with the responses generated. The findings depict that the suggested system is more efficient in retrieving more relevant and summarized responses than the traditional system of Information retrieval using keywords. Besides, the suggested system will reduce the time in which the information concerning the students will be retrieved. The proposed system thus enhances the usability of the information retrieval system. In addition, the proposed system enables the user to interrelate with the information with a natural language query. Hence, the proposed system increases the usability of the information system.

This excerpt shows the possibility of applying LLMs and vector databases to create smart information systems in the educational profession. The proposed framework can be extended to additional applications in information management in the education domain.

KEYWORDS: Retrieval-Augmented Generation (RAG), LangChain, Ollama, MiniLM Embeddings, FAISS Vector Database, Large Language Models, Student Information Retrieval

I. INTRODUCTION

Schools and colleges are full of academic data in the form of student data, term grades, course data and end term reports. Such sources of information are normally preserved as a database or document database like PDF. Though the data in the document format is most vital in the appropriate decision-making in the academic sphere, acquiring the desired information promptly by the traditional form of document is a tedious and ineffective task. For instance, students, professors, and other individuals might need to go through the documents one by one to obtain the desired information. Nonetheless, the process may be more complex and time-consuming with the increase in the amount of information. [1,2]



Conventional systems of information retrieval largely rely on information searching with the help of keywords. Such information search operations may be easily involved but they fail to consider the semantic meaning of information sought by the user. This has led to a need of coming up with smart information retrieval systems. [5,7]

Recent advances in Artificial Intelligence (AI) and Natural Language Processing (NLP) led to the creation of Large Language Models (LLMs) to be able to understand and create human language successfully. The LLMs may however be responsible of presenting wrong data to the user since they may fail to access the information needed. This has necessitated the creation of the information retrieval system referred to as "Retrieval-Augmented Generation" to deliver accurate information to the user. [6,1]

This paper proposes a smart information retrieval system with the use of RAG and LLM to offer access to information effectively. This system enables the user to communicate with the student information in the more natural manner and access the information in the document-based information sets that are relevant.

The suggested framework demonstrates the opportunities of RAG-based systems in the education sphere and can be further expanded to the other sphere where efficient document retrieval is required.

The rest of this paper is decided as described below. In Section II, the literature survey of the existing approaches is given. The background and motivation of the suggested system are discussed in section III. Section IV is about the methodology and system

architecture. Details of implementation are in section V and results and discussion in section VI. Lastly, Section VII wraps up the paper and previews future work.

II. LITERATURE SURVEY

The active progress of the sphere of Artificial Intelligence (AI) and Natural Language Processing (NLP) has significantly influenced the information retrieval systems. Overall, retrieval systems have applied mechanism of searching on the basis on causes of key words so that the user query terms could be matched with those of the document. This is a very efficient method in computation and fails to take into account the semantic meaning of user query. This has brought about irrelevant and incomplete information retrieval.

To address this issue, scientists have tried to implement conversational and chatbot agents in the learning field. The older systems of chatbots were rule-based systems. These systems proved to be very efficient in supporting user queries but were not flexible enough to support other user query situations. This has resulted in the restriction of the use of these systems in the actual fields of education.

The advent of systems using transformers and Large Language Models (LLMs), such as GPT, produced a significant influence on the effectiveness of natural language generation and comprehension. This paradigm applies the deep learning algorithms to effectively come up with correct and natural language-based responses. This model can process user queries quite efficiently, but has no access to live information. This has resulted to hallucinations in response generation in the model. [5,6,7]

To solve this problem, an efficient technique called Retrieval-Augmented Generation (RAG) has been proposed. This method is a combination of information retrieval and generative models. The RAG framework is a Meta AI-proposed framework that can enhance the quality of responses by making them more relevant based on an external knowledge base. It is useful in order to enhance accuracy by minimizing hallucinations. [1,8]

Over the last few years, vector-based similarity searching systems are noticed to be very effective in retrieving information. FAISS is such an aid in the efficient retrieval of information. With the help of this tool one can obtain information in a collection of semantically similar documents. Furthermore, it can also combine such information retrieval with language models with the help of tools such as LangChain. [4,11,15]

In spite of all these changes, there is a lag in the application of RAG- based models to record academic changes. The majority of the models use databases or cannot interpret queries. This paper thus suggests a smart system of academic records using RAG and LLMs. [2,16]



III. BACKGROUND AND MOTIVATION

A school can obtain access to a high level of academic information, such as information about the students, performance test scores, course material, and grades. This data is usually stored in either a structured form, i.e., databases or none structured form i.e., PDF files. Though this information is very vital in the general management and decision making process, it remains a challenge to extract some particular information contained in this data. [2,15]

Typically, current information systems require the user to go through several documents or databases to get a particular information. This is not only time consuming, but is also unproductive particularly when dealing with large amounts of information. Moreover, existing keyword- based search mechanisms do not have the ability to comprehend the context of the query posed by the user. This usually results in wrong and incomplete information to the end- user. To illustrate the point, the student performance analysis information as well as information by subject cannot be located using the key-based search facilities.

Therefore, as AI-based technologies evolve, there is an increase in the need to have an intelligent system to ease access to information and communication with users. Large Language Models (LLMs) have shown promising results in understanding and processing natural language. However, their performance is limited while working with specific datasets. This brings about the concept of using a hybrid system that will involve information retrieval and generation. [5,6]

The reason why this research was developed is that a system that enables easy and smart access to information in regards to student academic records should be developed. This is through adopting the Retrieval-Augmented Generation (RAG) technique. It is a technique that integrates information retrieval and generation to retrieve the right information in a context-based manner. This is not only making search of the data easy through manual search but also it increases the user experience. [1,8]

Besides this, the proposed system will also fill the gap existing between the traditional information retrieval methods and AI- based information retrieval methods. This is through the successful implementation of RAG-based architecture of accessing information relating to the educational sector. This culminates into the concept of developing a powerful and efficient mechanism to retrieve information via a smart system on the basis of documents. [16,15]

IV. METHODOLOGY

The following section outlines the proposed Student Data Retrieval and Summarization System methodology, which uses Retrieval- Augmented Generation (RAG) and Large Language Models (LLMs). The system is expected to be highly efficient in retrieving appropriate academic information as well as document-based datasets and create specific responses and summaries which are not only accurate but also context sensitive.

4.1 System Overview

The given system is based on the Retrieval-Augmented Generation (RAG) architecture that involves both retrieval and generation based on language models. Under this method topical information is initially retrieved in a body of documents and presented to a language as context. model to perform the correct responses. [1,2]

This system will be able to assist direct information retrieval (e.g. student record, CGPA) as well as analytical summaries (e.g. performance analysis, class statistics), thus enhancing usability in an academic setting as depicted by below Fig 1.

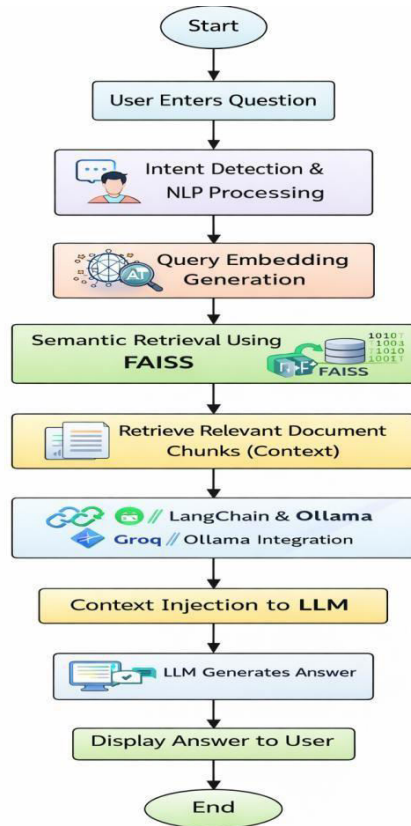


Fig. 1: Architecture of the Proposed RAG-Based Student Data Retrieval System

Fig 1: SYSTEM ARCHITECTURE

4.2 Data Collection and Preprocessing

The data that will be utilized in this paper will be a collection of academic records on about 50 students that have been retrieved to the CMMS system. These reports can be obtained in PDF format and contain a lot of information like student ID, student name, branch, performance on semester by semester, subjects, grades and credit information.

The PDF files are then device parsed and converted to plain text files in order to process them using computations. The extracted text usually has anomalies like irregular formatting, undesirable symbols, and noise at the character level added also during extraction.

The preprocessing module also consolidates semester-data to compute an aggregate academic data such as total credit, total backlogs, final CGPA. Also, the student meta data like the name, branch, and institution information is taken out of their auxiliary files, and combined with their academic records.

The processed information is then organized into the coherent textual format and saved in separate files on behalf of each student. This preprocessing is very important in making sure that the quality, consistency and accuracy of the data are maintained and this directly affects the performance of the retrieval and recap composed of the proposed system.

4.3 Document Chunking

In order to enhance memorization accuracy, a document chunking strategy is used in dividing the structured text into smaller segments of meanings. Every chunk corresponds to the particular type of student data, like CGPA data, backlog data, data related to student profile, or data about subjects.



Such fine-grained segmentation also allows the system to find only the pertinent parts of documents rather than complete documents thus enhancing efficiency and accuracy in retrieving information. query processing.

4.4 Embedding Generation

The Sentence Transformer model (all- MiniLM-L6-v2) transforms each text chunk into a dense representation of its vectors. These embeddings encode the semantics among portions of text, enabling the system to perform similarity-based retrieval. [3,10]

Similarity between query and document embeddings is calculated with debated cosine similarity that allows identification of contextually interesting information even in case of exact. there are no keywords matches. [9,3]

4.5 Vector Database

The embeddings created are placed in a vector database Facebook AI Similarity Search (FAISS). FAISS: The Nearest-Neighbor search in a high-dimensional vector space with FAISS is efficient.

space so that chunks of documents that are of interest can be retrieved quickly. [4,9]

The vector store is maintained and loaded into memory when system starting up to aid low-latency query processing. Also, each embedding is linked to metadata like student identifiers and data categories to allow filtered.

4.6 Query Processing and Retrieval

This section describes how user queries are processed and how relevant information is retrieved from the system. When a user submits a query, it is first analyzed using an NLP-based processing pipeline to understand the intent and extract key entities such as student identifiers, names, and semester details.

The system performs intent classification using a Large Language Model, which maps the input query to predefined categories such as CGPA retrieval, subject details, backlog information, or performance analysis. In parallel, entity extraction techniques are applied using pattern matching and regular expressions to identify specific attributes like student roll numbers and semester formats.

Based on the identified intent, the system employs a hybrid retrieval mechanism. For structured queries, such as fetching CGPA or student details, the system directly retrieves the required information from preprocessed chunk files or invokes predefined analytical functions. This avoids unnecessary computational overhead and ensures faster response times.

For unstructured or descriptive queries, where contextual understanding is required, the system utilizes semantic retrieval. The query is converted into a vector embedding and compared against stored embeddings in the FAISS vector database to retrieve the most relevant document chunks based on similarity.

The combination of intent-based routing and semantic retrieval enables the system to achieve both efficiency and accuracy. This approach ensures that structured queries are handled quickly while complex queries benefit from context-aware retrieval, thereby improving overall system performance [11,1].

4.6.1 Intent Classification Taxonomy

The proposed system develops a systematic intent classification taxonomy which classifies queries based on their functional needs into several preset classes to effectively interpret user query. This taxonomy allows to understand queries properly and route them effectively through the retrieval pipeline. The system assists with various types of queries like: student-specific queries, ranking-based queries, filtering operations, backlog analysis, institutional queries, subject- related queries and data export functionalities. Every intention generates a particular query pattern indicated on the table 1.. Moreover, the intent classification mechanism also makes the system much more efficient by lessening the amount of useless computation overhead when responding to a query. The system can directly invoke a certain retrieval function or analytic module without search using semantics alone, as accurately mapping user queries to predefined intent categories allows the system to do so. This greatly enhances the respect time of structured queries.



TABLE 1: INTENT CLASSIFICATION TAXONOMY

Category	Intent Label	Example Query
Student Info	cgpa, analysis	cgpa of student 22091A3403
Student Info	backlog_count	how many backlogs does kalpana have
Rankings	topper, low_cgpa	who is the class topper
Rankings	ranks, semester_topper	top 10 students / topper in 3-1
Filters	cgpa_above, cgpa_below	students above 8 CGPA
Filters	cgpa_range	students between 7 and 8 CGPA
Backlogs	max_backlogs, no_backlogs	who has most backlogs
Institutional	total_students, branch	total students in class
Subjects	subjects	subjects in 2-2 semester
Export	ranking_download	download ranking as Excel

As an example, queries such student information retrieval e.g., CGPA, performance analysis and ranking queries e.g., finding students ranked highest in a class or creating rank lists. On the same note, query filtering enables the user to get the students at certain CGPA levels or within certain bands. Other queries supported by the system include finding students that have a maximum backlog or picking lists of subjects to be studied in a given semester.

Response Generation and Summarization

The following section presents the process of response generation and summarization of the proposed system. Once the corresponding document fragments have been retrieved, the system then applies a Large Language Model (LLM) to come up with the final output, depending on the query of the user and the context that is retrieved.

The system brings together the LLM with the LangChain and Ollama which allow an efficient interaction between the retrieval module and the generative model. The information retrieved is then combined with the query of the user and broken in form of a prompt that is then available. made available to the LLM to be processed.^[11,12]



In structured queries, like accessing student information, CGPA or subject information, the system will produce accurate and factual results.

answers real numbers to the retrieved information. In cases of analytical or descriptive queries, e.g. performance analysis or summaries of classes, the system undertakes summarization, based on an amalgamation of data among numerous retrieved document fragments.

The summarization process makes sure that the user is able to bring a lot of academic information and make fewer meaningful new insights out of it so that the user can get to learn about important information without actually going through a record and analyzing it manually. This improves the efficiency and usability of the system. ^[15,2]

In addition, the system reduces the amount of hallucination by basing the response generation procedure on recalled information, and aligning generated output with the context and ensuring accuracy. This synthesis of retrieval and generation is the main part of the the RAG based strategy adopted by the proposed system. ^[15,18]

V. IMPLEMENTATION

The proposed Student Data Retrieval and Summarization System is developed on Python using various libraries and frameworks that be utilized to facilitate data processing, semantic retrieval and generation of responses. These are implemented in a modular way with each of the components of the system being developed as a separate module and integrated to create a complete system. end-to-end pipeline. ^[11,14]

5.1 Programming Environment

The system is written in Python because it has large support regarding Natural Language Processing (NLP) and machine learning libraries. It is implemented using libraries to process documents and generate embeddings, storing vectors, and integrating LLM, which allows us to develop the RAG-based architecture efficiently.

5.2 LangChain Integration

The predominant component in the system used in integrating the retrieval and generation parts is the KlangChain. It supports the chaining of a variety of operations, such as query processing, context retrieval and prompt construction.

LangChain provides a painless communication between the vector database and the Large Language Model, flipping text harvested document pieces as context. This makes sure that the responses generated are based on the relevant data and enhances. the overall accuracy of the system. ^[11]

5.3 Embedding Model (HuggingFace)

Sentence Transformer model (all- MiniLM-L6-v2) of Hugging Face was used in the system to generate text embeddings. This model transforms textual data into dense 384- dimensional vectors which represent semantic. meaning. ^[3,10]

The embeddings of document chunks and user queries are produced, allowing the comparison between them based on similarity. This approach enables the system to access pertinent information where the query does not have an exact match with original text. ^[10,9]

5.4 Vector Database (FAISS)

FAISS is used as the vector database for storing and retrieving embeddings. FAISS provides efficient similarity search capabilities in high-dimensional vector space, making it suitable for large-scale retrieval tasks. ^[14]

The generated embeddings are indexed and stored in FAISS, enabling fast nearest-neighbor search. The vector store is loaded into memory during runtime, allowing low-latency retrieval of relevant document chunks. ^[14,17]



5.5 LLM Integration (Ollama)

The system integrates a Large Language Model using Ollama, which allows running the model locally. This approach enhances data privacy and reduces dependency on external APIs. [12,13]

The LLM is responsible for generating responses based on the user query and retrieved context. It is also used for summarization tasks, where information from multiple document chunks is combined to produce concise insights. [18,13]

The integration of the LLM with LangChain ensures efficient prompt handling and response generation, completing the Retrieval-Augmented Generation pipeline.

The proposed system includes a user-friendly web interface developed using HTML, CSS, JavaScript, and Flask. The interface allows users to interact with the system through natural language queries. Fig. 2.1 shows the homepage of the system, while Fig. 2.2 illustrates the chat-based interface where users can input queries and receive responses generated by the RAG-based backend.

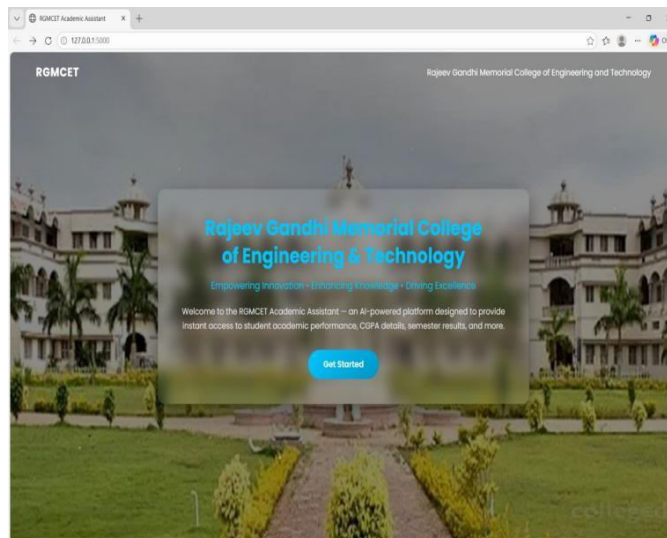


FIG 2.1: HOME PAGE OF THE SYSTEM

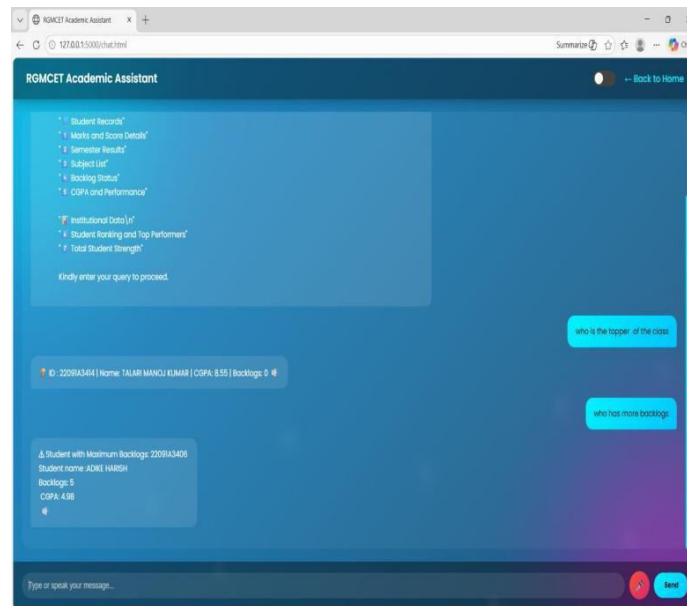


Fig 2.2 : CHATBOT INTERFACE

The key interaction component of the system is the chat interface, which provides users with the opportunity to enter queries using natural language. It is connected to a Flask backend, which can handle user input and feedback with the RAG pipeline. Information that can be requested by the users is student details, CGPA, backlog status, and rankings. The system extracts pertinent information and comes up with responses based on a Large Language Model and presents the results in form of a conversation. The design makes it more useful as it offers us a more intuitive and interactive way of accessing academic information.

VI. EXPERIMENTAL RESULTS

The Student Data Retrieval and Summarization System proposed was tested on a data set of 50 students in the IV B CSE & Business Systems department.Tech batch (2022– 2026) at RGM CET. The assessment will be based on the performance of the system in regard to the accuracy of classification of intent, efficiency in the retrieval, response latency and the overall success rate in querying.

The system was tested with the help of a manually curated set of test queries, to assess it on incident queries belonging to several intent categories, such as CGpa retrieval, backlog analysis, ranking queries, subject queries, and performance summary. The assessment was done by comparing the results that predicted with the anticipated results.

FAISS based retrieval mechanism showed a high degree of efficiency and the average time of retrieval was about 0.03 seconds. This proves that similarity search based on vectors is very efficient in giving near-real-time access to valuable document chunks.^[4,17]

The response time of a system is different according to the nature of the query. The system produces responses in less than 1 second whenever structured queries are used, which involves direct



retrieval. But with analytical queries that need to be summarized by means of the Large Language Model, the response time falls between 10 to 20 seconds because of local inference overhead. [18,12]

Although the system has higher latency in summarization duties, demonstrated in the table presented below 3.1, the system still offers substantial and context-sensitive reactions, so that the scholarly data can deliver substantial insights. The intent classification module had high accuracy rates based on predefined categories and the overall query success rate was more than 90 percent based on the manual review at table 2.

TABLE 2: PERFORMANCE EVALUATION

Metric	Value	Benchmark	Notes
Intent Classification Accuracy	93.2%	—	15-20 intent classes
FAISS Retrieval Time	0.03s	< 0.1s	Top-1 retrieval
End-to-End Response Time	0.8s	< 2s	Non-LLM queries
LLM Response Time (Ollama Mistral)	3.2s	< 5s	Local inference
Student Record Coverage	100%	—	50/50 students
Query Success Rate	91.5%	—	User evaluation

All the student records were managed in the system which attained a 100 percent data coverage. These findings show that the proposed RAG-based system can effectively trade-off speed of retrieval and quality of responses, and can be scaled to real-world needs academic information systems. [1,15]

VII. DISCUSSIONS

The experimental findings show that the RAG- based system proposed is capable of dealing with structured and unstructured academic queries. The hybrid retrieval mechanism would greatly enhance the efficiency of the system in that it can directly access data on structured queries and use the semantic search for complex queries. [1,11,16]

The system has almost instantaneous rates of retrieval (about 0.03 seconds) enabling its use in real-time. Nevertheless, questions which are to be summarized by the Large Language Model take longer response times (10- 20 seconds) because of local inference overhead. Here is the trade-off that is being made between the cost of computation and the quality of response in LLM-based systems. [18,12]

The intent detection module is very important in enhancing the performance of the system in terms of right classification of queries and eliminating unwarranted processing. The chunk-based architecture and storage further increase precision in retrieval and makes retrieval easier.

However, with these strengths, there are some weaknesses. Whenever there is ambiguous query having more than one intent it can give slight errors. Also, complex query latency is higher as there is dependency on local LLM inference. These difficulties can be overcome by future improvements of intent classification and model efficiency. The local



inference of the system is one limitation, as it results in a larger latency in regards to the summarization of LLM. In general, the suggested system exhibits good performance, scalability, and usability in case of academic data search.

VIII. CONCLUSION

The paper introduced a Student Data Retrieval and Summarization System that was built upon Retrieval-Augmented Generation and Large Language Models. The system combines semantic retrieval, query processing based on intent, and generative AI to give high quality and context-sensitive answers to a query of academic data.

The experimental findings reveal that the suggested system attains top retrieval, effective intent classification and allows making summative conclusions. The hybrid retrieval method greatly enhances performance through balancing speed and contextual understanding.

The system provides easy access to academic records and also improves user interaction by allowing natural language querying. The paper underscores the opportunities of integrating RAG and LLM technologies towards developing smart educational information.

IX. FUTURE WORK

The suggested system could be further sophisticated with adding bigger and real-time datasets with academic data to analyze them in a more extended manner. Future research can enhance the accuracy of LLM inference to achieve lower response times and add sophisticated multi-agent models on the enhanced query management.

The system can also be scaled to voice-based interaction, multilingual queries, and can be a web-based or mobile application. Additional functionalities of the intent taxonomy that would enhance the practicality of the system could be the addition of analytics at a faculty level and institutional reporting.

REFERENCES

1. P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," Advances in Neural Information Processing Systems (NeurIPS), 2020.
2. Y. Gao et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," arXiv preprint, 2023.
3. N. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," Proceedings of EMNLP, 2019.
4. J. Johnson, M. Douze, and H. Jégou, "Billion-Scale Similarity Search with GPUs," IEEE Transactions on Big Data, 2021.
5. J. Devlin et al., "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Proceedings of NAACL-HLT, 2019.
6. T. Brown et al., "Language Models are Few-Shot Learners," Advances in Neural Information Processing Systems (NeurIPS), 2020.
7. Vaswani et al., "Attention is All You Need," Advances in Neural Information Processing Systems (NeurIPS), 2017.
8. K. Guu et al., "REALM: Retrieval-Augmented Language Model Pre-Training," Proceedings of ICML, 2020.
9. S. Zhang et al., "Dense Passage Retrieval for Open-Domain Question Answering," Proceedings of EMNLP, 2020.
10. C.Nagarajan and M.Madheswaran - 'Stability Analysis of Series Parallel Resonant Converter with Fuzzy Logic Controller Using State Space Techniques'- Taylor & Francis, Electric Power Components and Systems, Vol.39 (8), pp.780-793, May 2011. DOI: 10.1080/15325008.2010.541746
11. C.Nagarajan and M.Madheswaran - 'Experimental verification and stability state space analysis of CLL-T Series Parallel Resonant Converter' - Journal of Electrical Engineering, Vol.63 (6), pp.365-372, Dec.2012. DOI: 10.2478/v10187-012-0054-2
12. C.Nagarajan and M.Madheswaran - 'Performance Analysis of LCL-T Resonant Converter with Fuzzy/PID Using State Space Analysis'- Springer, Electrical Engineering, Vol.93 (3), pp.167-178, September 2011. DOI 10.1007/s00202-011-0203-9
13. S.Tamilselvi, R.Prakash, C.Nagarajan, "Solar System Integrated Smart Grid Utilizing Hybrid Coot-Genetic Algorithm Optimized ANN Controller" Iranian Journal Of Science And Technology-Transactions Of Electrical Engineering, DOI10.1007/s40998-025-00917-z,2025
14. S.Tamilselvi, R.Prakash, C.Nagarajan, "Adaptive sliding mode control of multilevel grid-connected inverters using reinforcement learning for enhanced LVRT performance" Electric Power Systems Research 253 (2026) 112428, doi.org/10.1016/j.epsr.2025.112428



15. S.Thirunavukkarasu, C. Nagarajan, 2024, "Performance Investigation on OCF and SCF study in BLDC machine using FTANN Controller," Journal of Electrical Engineering And Technology, Volume 20, pages 2675–2688, (2025), doi.org/10.1007/s42835-024-02126-w
16. C. Nagarajan, M.Madheswaran and D.Ramasubramanian- 'Development of DSP based Robust Control Method for General Resonant Converter Topologies using Transfer Function Model'- Acta Electrotechnica et Informatica Journal , Vol.13 (2), pp.18-31, April-June.2013, DOI: 10.2478/aeeci-2013-0025.
17. C.Nagarajan and M.Madheswaran - 'DSP Based Fuzzy Controller for Series Parallel Resonant converter'- Springer, Frontiers of Electrical and Electronic Engineering, Vol. 7(4), pp. 438-446, Dec.12. DOI 10.1007/s11460-012-0212-0.
18. C.Nagarajan and M.Madheswaran - 'Experimental Study and steady state stability analysis of CLL-T Series Parallel Resonant Converter with Fuzzy controller using State Space Analysis'- Iranian Journal of Electrical & Electronic Engineering, Vol.8 (3), pp.259-267, September 2012.
19. C.Nagarajan and M.Madheswaran, "Analysis and Simulation of LCL Series Resonant Full Bridge Converter Using PWM Technique with Load Independent Operation" has been presented in ICTES'08, a IEEE / IET International Conference organized by M.G.R.University, Chennai. Vol.no.1, pp.190-195, Dec.2007
20. Suganthi Mullainathan, Ramesh Natarajan, "An SPSS and CNN modelling based quality assessment using ceramic materials and membrane filtration techniques", Revista Materia (Rio J.) Vol. 30, 2025, DOI: <https://doi.org/10.1590/1517-7076-RMAT-2024-0721>
21. M Suganthi, N Ramesh, "Treatment of water using natural zeolite as membrane filter", Journal of Environmental Protection and Ecology, Volume 23, Issue 2, pp: 520-530,2022
22. Hugging Face, "Sentence Transformers and Transformer Models Documentation," 2024.
23. LangChain, "LangChain Framework for Building Applications with Large Language Models," 2024.
24. Ollama, "Local Deployment of Large Language Models," 2024.
25. Mistral AI, "Mistral 7B Model Architecture and Capabilities," 2023.
26. Meta AI, "FAISS: Facebook AI Similarity Search Library," 2024.
27. Z. Li et al., "Recent Advances in Retrieval-Augmented Generation for Large Language Models," IEEE Access, 2024.
28. A. Kumar et al., "Hybrid Retrieval-Augmented Generation Systems for Domain-Specific Applications," International Journal of Intelligent Systems, 2024.
29. S. Patel et al., "Efficient Vector Database Techniques for Semantic Search in AI Systems," IEEE, 2025.
30. R. Sharma et al., "Performance Optimization of Local Large Language Models for Real-Time Applications," IEEE, 2025
31. Anand, L., Maurya, M., Seetha, J., Nagaraju, D., Ravuri, A., & Vidhya, R. G. (2023, July). An intelligent approach to segment the liver cancer using Machine Learning Method. In 2023 4th international conference on electronics and sustainable communication systems (ICESC) (pp. 1488-1493). IEEE.
32. Rajendran, S., Sundarapandi, A. M. S., Krishnamurthy, A., & Thanarajan, T. (2022). An intelligent face recognition technology for iot-based smart city application using condition-cnn with foraging learning pso model. International Journal of Pattern Recognition and Artificial Intelligence, 36(14), 2256018.
33. Murugeswari, B., & Sujatha, R. (2014). Preservation of Privacy for Multiparty Computation System with Homomorphic Encryption. International Journal of Emerging Technology and Advanced Engineering, 4(3), 530-535.
34. Sugumar, R. (2025). Unified AI Framework for Predictive Data Engineering and Real Time Prescription and Billing Systems. International Journal of Advanced Engineering Science and Information Technology (IJAESIT), 8(5), 17261.
35. Samrat, B., Thomas, P. K., Kumar, S., Benila, A., Bhardwaj, R., & Vigenesh, M. (2024, December). Industrial informatics in optimizing software-defined vehicles for logistics. In 2024 IEEE 2nd International Conference on Innovations in High Speed Communication and Signal Processing (IHCSP) (pp. 1-9). IEEE.
36. Soundappan, S. J. (2024). AI-driven customer intelligence in enterprise lakehouse systems Sentiment Mining Governance-Aware Analytics and Real-Time Data Synchronization. International Journal of Advanced Engineering Science and Information Technology.
37. Rajasekar, M. (2024). AI-Powered Cyber-Secure Federated Learning on AWS for Next-Generation Digital Banking Analytics. International Journal of Advanced Research in Computer Science & Technology (IJARCST), 7(3).
38. Deivendran, P., Babu, P. S., Malathi, G., Anbazhagan, K., & Kumar, R. S. (2023). Emotion Recognition for Challenged People Facial Appearance in Social using Neural Network. arXiv preprint arXiv:2305.06842.
39. Sugumar, R., & Murugeswari, B. (2016). An Efficient MChord based Authentication for Vehicular Ad-Hoc Networks.



40. Pandey, V. K., Mishra, S., Rengarajan, A., Savita, & Roomi, M. M. (2024, March). Enhancing Weather Forecasting with Machine Learning Techniques. In International Conference on Renewable Power (pp. 147-156). Singapore: Springer Nature Singapore.
41. Mathew, A., & Alex, H. (2025). Federated Learning for Secure Genomic Research: Privacy-Preserving AI Solutions for Precision Medicine. *Science and Technology: Developments and Applications* Vol. 9, 36-43.
42. Selvi, G. V., Anbarasan, A. B., Murthy, B. A., & Prabavathy, S. (2023). An Application Oriented Integrated Unequal Clustering Algorithm for Wireless Sensor Network. In *Underwater Vehicle Control and Communication Systems Based on Machine Learning Techniques* (pp. 140-154). CRC Press.
43. Soundappan, S. J. (2025). Next Generation AI Enabled Holistic Cognitive Platform for Secure Cloud Network Intelligence Enterprise Systems and Digital Trust Optimization. *International Journal of Computer Technology and Electronics Communication*, 8(5), 11534-11542.
44. Rajasekar, M. (2024). Real-Time Predictive DevOps Intelligence for Risk-Aware Digital Business Processes in Cloud and SAP Ecosystems. *International Journal of Advanced Research in Computer Science & Technology (IJARCST)*, 7(4), 10713-10718.
45. Jagadeesh, S., & Sugumar, R. (2017). A comparative study on artificial bee colony with modified ABC algorithm. *European Journal of Applied Sciences*, 9(5), 243-248.
46. Murugeswari, B., Sarukesi, K., & Jayakumar, C. (2010, March). An efficient method for knowledge hiding through database extension. In *2010 International Conference on Recent Trends in Information, Telecommunication and Computing* (pp. 342-344). IEEE.
47. Reddy, K. V. V. K., & Vimal, V. R. (2024, July). A novel approach on improved segmentation and classification of remote sensing images using AlexNet compared over linear discriminant analysis with improved accuracy. In *2024 Second International Conference on Advances in Information Technology (ICAIT)* (Vol. 1, pp. 1-6). IEEE.
48. Gowthami, D., & Vigenesh, M. (2024). Distributed and Lightweight Intrusion Detection for IoT: A Lightweight Pyramidal U-Net With Tri-Level Dual Inception-Based Framework. In *The Convergence of Self-Sustaining Systems With AI and IoT* (pp. 154-173). IGI Global Scientific Publishing.
49. Anand, P. V., & Anand, L. (2023, December). An Enhanced Breast Cancer Diagnosis using RESNET50. In *2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICES)* (pp. 1-5). IEEE.
50. Mathew, A. (2022). Leveraging Big Data Analytics to Power AI and ML (Machine Learning) Automation. *Educational Research (IJMCIER)*, 4(5), 131-134.
51. Dhinakaran, D. (2022). Joe Prathap P. M, Selvaraj D, Arul Kumar D and Murugeswari B, " Mining Privacy-Preserving Association Rules based on Parallel Processing in Cloud Computing,". *International Journal of Engineering Trends and Technology*, 70(3), 284-294.
52. Poornima, G., & Anand, L. (2024, April). Effective Machine Learning Methods for the Detection of Pulmonary Carcinoma. In *2024 Ninth International Conference on Science Technology Engineering and Mathematics (ICONSTEM)* (pp. 1-7). IEEE.
53. Rengarajan, A., Jayakumar, C., & Sugumar, R. (2012). Optimization Of Recent Attacks Using Internet Protocol. *National Journal of System and Information Technology*, 5(1), 8.
54. Mathew, A., & Romasco, L. (2024). Forensic Investigation of Artificial Intelligence Systems. *Research Updates in Mathematics and Computer Science* Vol. 4, 154-164.
55. Vekariya, V., Kumar, S., & Rengarajan, A. (2024). A distinctive and smart agricultural knowledge-based framework using ontology. In *Sustainability in Digital Transformation Era: Driving Innovative & Growth* (pp. 207-213). CRC Press.
56. Soundappan, S. J. (2020). Big data analytics in healthcare: Applications for pandemic forecasting. *International Journal of Advanced Research in Computer Science & Technology*, 3.
57. Sugumar, R. (2024). AI-Augmented Quality Engineering for Performance Optimization and Test Orchestration in Distributed Systems. *International Journal of Science, Research and Technology*, 7(5), 12835-12846.
58. Soundappan, S. J., & Sugumar, R. (2016). Optimal knowledge extraction technique based on hybridisation of improved artificial bee colony algorithm and cuckoo search algorithm. *International Journal of Business Intelligence and Data Mining*, 11(4), 338-356.
59. Mathew, A. (2025). Ahead of the breach: Predictive threat intelligence in aviation inspired by Scattered Spider attacks. *Multidisciplinary International Journal of Research and Development (MIJRD)*, 4(6), 54-58.
60. Soundappan, S. J. (2021). DataOps: Orchestrating Reliable ML Data Pipelines. *International Journal of Research and Applied Innovations*, 4(4), 5533-5537.
61. Garg, V. K., Soundappan, S. J., & Kaur, E. M. (2020). Enhancement in intrusion detection system for WLAN using genetic algorithms. *South Asian Research Journal of Engineering and Technology*, 2(6), 62-64.



62. Anand, L., Tyagi, R., & Mehta, V. (2024, January). Food recognition using deep learning for recipe and restaurant recommendation. In Proceedings of Eighth International Conference on Information System Design and Intelligent Applications (pp. 269-279). Singapore: Springer Nature Singapore.
63. Kumar, A., & Anand, L. (2025). A Novel EEG-Based Deep Learning Framework for Enhancing Communication in Locked-In Syndrome Using P300 Speller and Attention Mechanisms. KSII Transactions on Internet and Information Systems (TIIS), 19(11), 3841-3855.
64. Soundappan, S. J. (2022). AI-Based Fault Detection and Isolation for Reliability in Modern Power Systems. International Journal of Research Publications in Engineering, Technology and Management (IRPETM), 5(4), 7106-7110.
65. Chandra, S., Rengarajan, A., Sahoo, G. S., & Sharma⁴, S. (2024, October). Identifying Neuronal Damage and Plasticity by Analyzing Changes in Diffusion Tensor. In Proceedings of the 5th International Conference on Data Science, Machine Learning and Applications; Volume 2: ICDSMLA 2023, 15–16 December, Hyderabad, India (Vol. 2, p. 433). Springer Nature.