APPLICATIONS OF RETRIEVAL-AUGMENTED GENERATION SYSTEMS IN HIGHER EDUCATION: A REVIEW

[1]KHADIJA ALMOHSEN, [2]JAFLA AL AMMARI

American University of Bahrain, University of Bahrain
E-mail: [1]Khadija.almohsen@aubh.edu.bh, [2]jalammari@uob.edu.bh

Abstract - This paper presents a thorough review of the emerging field of Retrieval Augmented Generations (RAGs), with focus on its applications in higher education domain. RAGs' exceptional properties, inherited from the strengths of Large Language Models (LLMs) with Information Retrieval (IR) models, bring novelty and creativity to its applications in different activities in higher education including teaching and learning, student services and administration, and quality and institutional effectiveness. This survey to the literature reveals the increasing utilization of RAG in education, driven by to its pivotal role in advancing the provided services and enhancing students learning experience. It identifies critical gaps in the current research and outlines promising future directions with transformative potential for higher education, offering valuable guidance for researchers aiming to advance this emerging field.

Keywords - Higher education, Large Language Model, Retrieval Augmented Generation, Information Retrieval.

I. INTRODUCTION

Continuous advancement in computational power, Artificial Intelligence (AI) and big data is driving the parallel progression of Generative AI (GenAI) technologies [1], [2]. As GenAI gains popularity and demonstrates exceptional capabilities in producing realistic yet creative content across different media, wide range of fields were motivated to explore and adopt GenAI technologies [1], [3], [4].

Large Language Models (LLMs) represent one GenAI technology that is capable of simulating human linguistic abilities in understanding or generating text-based content [1]. They represent the backbone of many conversational agents, question-answering systems, and other language-related tasks such as: summarization, translation, and sentiment analysis [2], [5]. As LLMs are pre-trained on a large amount of natural language corpus, they capture a vast amount of knowledge about the world in their model's parameters (i.e. their internal memory). This enables them to generate responses and retrieve answers from their internal memories without accessing any external sources [6]-[8]. Consequently, their generation performance was found to be improved by increasing the volume of their parameters and enlarging their memories capacity [7].

Despite their remarkable success, they have some limitations such as the knowledge update issue. This refers to their tendency to generate factual errors due to their outdated knowledge as they are limited to their training data and lack access to real-time knowledge [9], [10]. Moreover, LLMs suffer from hallucination as they might produce responses that deviate from the established knowledge when presented with novel queries that are beyond their training data or require current information [10], [11]. Lack of domain knowledge is another limitation that would require considerable manpower for dataset collection toaidin developing domain specific model [7].

To overcome these limitations and improve the accuracy of the results, researchers started exploring the integration of LLMs with information retrieval systems, leading to the development of a new type of models called Retrieval Augmented Generation (RAG) [3], [7], [9]-[11]. These models aim to combine the strengths of having updatable index in information retrieval systems with the unprecedented capabilities of LLMs, enhancing the accuracy and relevance of the generated responses [10].

Together, LLMs and RAG models are transforming how organizations across various domains access and make use of their available complex data, enabling processes streamlining and fostering a more efficient workforce [3], [11], [12]. Higher Education Institutions are among these organizations, benefiting from the adoption of these technologies to transform operations, automate various institutional practices, and support their stakeholders [3], [5], [8], [13].

The main aim of this research paper is to critically review and synthesize the existing body of literature on the adaptation of RAG in higher education, offering insights into how these systems can be utilized to enhance educational services and provide students with an improved learning experience. Additionally, this review seeks to identify commonalities and differences in the utilization of RAG within the higher education domain while providing a comprehensive understanding of the concepts and theories underpinning these systems.

This study holds significant value as it represents, to our knowledge, the first dedicated review of RAG applications in the domain of education. Given the rapid advancements in RAG technologies and their

growing popularity, this review serves as a valuable resource for researchers entering the field, supporting the development of innovative methods and models for applying RAG in higher education.

The remainder of this paper is structured as follows: Section II provides an overview of LLMs, followed by an exploration of RAG models in Section III. Section IV details the employed methodology, while Section V summarizes the reviewed articles. Section VI presents a comparative analysis of the findings and discusses their implications. Finally, Section VII concludes the paper.

II. LLMS OVERVIEW

LLMs are GenAI models that are trained on very large-scale, diverse, data sets, aiming to mimic human linguistic capabilities in text generation and understanding [1] . They represent the backbone of many conversational agents, question-answering systems, and other language-related tasks such as: summarization, translation, and sentiment analysis [2].

As LLMs are pre-trained on large natural language corpora, they represent huge repositories of knowledge, capturing a vast amount of knowledge about the world in their model's parameters (i.e. their internal memory). This enables them to generate responses and retrieve answers from their memories without accessing any external sources [6]-[8]. Consequently, their generation performance was found to be improved by increasing the volume of their parameters [7].

These LLMs could be classified into three categories based on their architectures, which are:

1. Encoder-only Models: They process the input and represent the relationship between its words and context in an encoder component only. One of their early developed models is Bidirectional Encoder Representations from Transformers (BERT) by Google. It was used as the basis for many other models such as: RoBERTa from Meta, and DeBERTa from Microsoft. These models excel in tasks that require understanding and comprehension rather than generation[1].
2. Encoder – Decoder Models: They use both encoder and decoder components. The encoder processes the input, captures its underlying structure and semantics, and represents them it into a hidden space. This hidden space will be used by the decoder to generate the desired output as human-readable text. Examples of such LLMs models are: T5, PLBART and AlphaCode. They showed notable performance in wide range of tasks, those requiring understanding as well as others requiring generation, such as: summarization, translation, and question-answering[1].
3. Decoder – only Models: These models produce the output without making use of an encoder component, but by relying on sequential token production. Among some of these models are: GPT series, ChatGPT, Claude, LLaMA, Vicuna, Gemini, Codex, and Copilot. These models are superior in generation tasks rather than comprehension [1], [2].

LLMs could also be classified as open source and closed-source. While closed-source models often outperform open-source alternatives in terms of performance, open-source models can compete them in specific applications due to their flexibility and adaptability. Open-source models offer notable advantages in terms of transparency, customizability, and cost-effectiveness, especially for researchers and organizations with domain-specific needs. Contrarily, closed-source models excel in providing pre-trained capabilities and high accuracy but are limited by licensing restrictions and lack of customization options [14], [15].

III. RETRIEVAL AUGMENTED GENERATION MODEL OVERVIEW

Although Large Language Models (LLMs) have been widely adopted by prominent conversational chatbots, such as ChatGPT, Gemini, and Copilot [10], demonstrating proficiency in various natural language tasks, such as: text summarization, question-answering, and code generation [2], [5] , they have certain limitations that might reduce accuracy of their responses. These limitations include: (1) reliance on pre-trained model with static knowledge base that might become outdated (2) hallucination when queried a about information beyond their knowledge scope, and (3) a lack of transparency in training, decision-making and response generation processes, which can reduce user trust [5], [9] - [11]. In response to these challenges, researchers - the computational linguistics community in particular- were motivated to dedicate more effort toward addressing these challenges. Retrieval-Augmented Generation (RAG) technology has since emerged to enhance LLMs by retrieving information from external sources and incorporating it into the prompts of LLMs [11].

RAG models could be classified into three main approaches [5], which are:

Naive RAG is the basic RAG paradigm which starts by splitting text into vectorized chunks stored in a database. When the user enters a query, it gets converted to a vector and used to retrieve similar chunks from the database, which are then passed to an LLM for response generation.

The advanced RAG is built on top of naive RAG by adding pre-retrieval steps to enhance indexing and the user query, as well as re-ranking retrieved contexts to ensure relevance. The focus of advanced RAG is posting the accuracy of the generated response.

- Modular RAG focus on enhancing the system flexibility and adaptability by having modular architecture that breaks down the system into re-usable and re-configurable modules.

RAG systems have three main modules that work together to achieve their objectives, which are the retriever, generator, and retrieval fusions [7], [10], [16]:

- Retriever: is the module which operates in two stages: indexing and retrieval. In the indexing stage, the external data sources are encoded into embeddings using an embedding model. These embeddings are efficiently indexed and stored in a vector database, forming the external knowledge base that facilitates domain adaptation and knowledge updates. In the retrieval stage, the user query is also encoded into an embedding using an embedding model. The retriever then performs a similarity search between the query embedding and the pre-indexed embeddings in the vector database to extract the most relevant pieces of data corresponding to the user query.
- Retrieval fusions: is the means of augmenting the data retrieved from the knowledge base by the retriever with the original user query. This combined information will be fed into the generator module to create the system's final response. There are three main fusion techniques which are: query-based fusion, latent fusion, and logits-based fusion. The query-based fusion combines the original user query with the retriever's results before passing them into the generator module. The logits-based fusion focuses on synthesizing the output logits from the generator with those of the retriever's results to pose the robustness of the combined logits. The latent fusion integrates the retriever's result representations into the latent representations of the generator module to implicitly improve the models' performance.
- Generator: is the module utilizing the LLM to generate the responses or makes predictions based on the user's query and the retriever's results.

Consequently, the typical workflow for a RAG system involves three steps: (1) retrieving the relevant information from external knowledge base based on given user query, (2) Integrating the retrieved information with original user query or intermediate states based on the fusion techniques employed; and (3) generating the system response based on the output of the previous step [7].

IV. METHODOLOGY

1) Exploratory Review of Literature:

A broad review of several research papers was conducted, aiming to gain breadth knowledge and comprehensive understanding of the adoption of RAG in the higher education domain. The sources included journal articles and conference papers accessed via Google Scholar and IEEE Xplore databases. The focus was on identifying the use of RAG systems in various areas in higher education, such as: teaching and learning, students' services and administration, and quality and institutional effectiveness.

1) Comparative Analysis:

The reviewed articles were compared and categorized based on several factors: the focus area (e.g. teaching and learning, students' services and administration, and quality and institutional effectiveness), target user groups (e.g., students, instructors, administrators), data sources / knowledge bases (e.g., institutional repositories, online course materials, student records, and external knowledge bases), and the LLM models utilized (e.g. GPT, T5, or other domain-specific models). This analysis provided valuable insights into the patterns and trends of RAG adoption in higher education, such as the common use cases, the types of data sources utilized to enhance retrieval, and the type of LLMs integrated in the system. By identifying commonalities and differences across studies, this step facilitated the identification of research gaps, that would encourage more research effort to be put in utilizing RAG systems in advancing higher education.

V. APPLICATIONS OF RAGS IN HIGHER EDUCATION

A thorough review of the literature reveals a growing research interest in RAG and the diverse avenues of exploration within the field. One of the active areas of research is the potential applications of RAG models in different domains, such as healthcare [11], finance [12], and education[3], [5], [8], [13], [15], [19]-[23].

Given that education is among the fields impacted by RAG technologies, and considering the context of this research work, this section will be devoted to review recent research works in the applications of RAG models in higher education, thereby providing a comprehensive overview of the current state-of the art for researchers interested in this evolving domain.

Several efforts were devoted to the development of custom chatbots as AI assisted tutors for specific subjects, tailored to meet the needs of specific students' population. These chatbot - based tutors aim to provide personalized, on-demand, guidance and up-to-date information relevant to the course context, as well as being 24/7 available [13], [19]. Among these tutors was Professor Leodar RAG-based chatbot [13] which aimed to enhance students' understanding of course content, their engagement, and their exam preparedness. It embedded in-depth subject knowledge by creating its knowledge base from the course materials, including lecture notes, Jupyter notebooks, and the recommended additional readings.

This knowledge base was split into chunks and grouped by their semantic similarity into vectors to ensure efficient retrieval. Whenever a student asks Prof. leader a question, it will look for relevant material in its knowledge base and will use the most relevant ones as the context of the prompt to be fed to the LLM, which is Claude 3 model developed by Anthropic. It then generates human-like, relevant, response to the student. This custom chatbot was evaluated over a semester on an introductory course in Data Science and AI at Nanyang Technological University, Singapore. Having Prof. Leodar using casual language increased the students' acceptance of the support provided, with many of them reporting a positive experience.

Star tutor is another notable example, that was developed as a specialized tutor in statistics to support students in carrying quantitative analysis [19]. The tutor adopted RAG model to mitigate the hallucination issue associated with LLMs. It used ChatGPT API, developed by Open AI, to comprehend the students' questions provided in natural language and generate accurate responses by retrieving relevant information from the course notes. These course notes, covering t-test and ANOVA, were segmented into chunks, considering ChatGPT's limitation of the size of contextual information it can handle.

These chunks were stored in vector database (DB), serving as searchable index for semantic retrieval. When a student enters a query, the most relevant chunks get retrieved, combined with the input query and chat history, and provided to ChatGPT 3.5 LLM as the prompt. This enabled Star tutor to reduce the likelihood of hallucination and enhance the accuracy of the result. Star Tutor was evaluated in a course taught by two instructors who assessed its responses to 20 questions using a predefined rubric. The rubric measured not only the accuracy and relevance of the responses but also the wealth of information provided. The evaluation showed that Star Tutor successfully answered all questions and accurately generated R code when requested.

Another research work investigated the use of RAG system in providing detailed, constructive, feedback for students on their assessments. Kai was developed as a special-purpose tutor, integrated in the Learning Management System of an introductory programming course, to simply provide feedback on programming assessments. Its knowledge base consisted of lectures' recording transcript texts with timestamps, pinpointing coverage times. This RAG-based tutor differs from naive RAG models as it does not receive a specific query from the user. On the other hand, the system shall generate the query by combining the student code solution together with the compiler output, the relevant passage from the lecture recording transcript, and the assessment descriptions. This gets sent to OpenAI GPT-4 LLM to generate feedback that includes relevant technical terms and video links for review. An interesting feature about Kai tutor is that it does not provide knowledge of the correct result but encourage independent learning by watching the lecture recording and knowing how to correct the mistake. This tutor was tested on 15 students who received two types of feedback-one with video links and one without. Feedback with video links was found helpful for complex errors, whereas simpler feedback was faster for straightforward mistakes [24].

Some RAG-based chatbots were designed not to serve as tutors but to improve information access and retrieval in academia through Question/ Answer (Q/A) [15], [22]. In [22], the focus was on overcoming the limitation of traditional search systems in providing contextual understanding. The implemented RAG-based chatbot indexed web software development course material in a Weaviate vector database that got searched whenever the user enters a query. This RAG model, powered by OpenAI GPT-3.5-Turbo, was compared with a traditional search bar, to assess its utility, user preference, and task performance. The results showed that while the RAG chatbot excelled in summarizing large amounts of information, the search bar provided faster results, suggesting integrating both functionalities for optimal outcomes in academic settings.

Authors of [15] intended to address traditional chatbots' limitations in providing accurate and contextually relevant answers due to their reliance on predefined datasets or lack of external knowledge integration. Their RAG model combined OpenAI GPT-3.5-Turbo with chromaDB vector database that stored 111 academic documents collected from faculty resources and university websites. The evaluation results demonstrated that their RAG system outperformed other LLMs such as Google Gemini Pro in generating accurate and contextually appropriate answers.

RAG-based chatbots were also developed to facilitate administrative tasks in academia, such as managing students' admission [20]. Such chatbots aim to ease providing prospective students with personalized consultations, saving them time by eliminating the need to browse and explore the institution's websites or visit campus for in-person discussions. This would benefit both students and institutions, enhancing the student experience while increasing the likelihood of attracting qualified applicants and sustaining institution's overall operations. The chatbot in [20] utilized ChatGPT 3.5 model and was evaluated on undergraduate admission data from three universities (one in the Unites State, one in Kenya and one in Hong Kong). The positive feedback received from students who used the admission chatbot indicated the potential role of RAG-based chatbots in transforming professional services in higher education.

HICON is another RAG-based consultation chatbot providing personalized guidance for students pursuing higher education, reducing reliance on costly and monopolized counseling services [23]. It used a categorization mechanism to classify students into groups based on their academic and professional backgrounds, providing customized prompts and advice tailored to individual needs. Added to that, it has resume screener feature that streamlined the process of evaluating student qualifications, enhancing the overall user experience. HICON integrated Llama 2, as the LLM, in its RAG solution to enable efficient data retrieval. The evaluation of HICON constituted evaluating both the accuracy of the categorizer, by measuring Precision, Recall, and F1 Score, and the truthfulness of the LLM, using TruthfulQA benchmark. The result of the evaluation indicated a satisfactory performance, revealing HICON potential in revolutionizing the student experience.

Another research effort [5] used a hybrid context RAG Pipeline, integrating LLM-augmented knowledge graphs and vector database, in supporting quality assurance practices and generating accreditation reports. Higher education institutions usually undertake a rigorous and time-consuming reporting process, to demonstrate proper alignment with the accreditation body standards. [5] integrated their institutional documents into a database that was prepopulated with the accreditor's data, and used ChatGPT 3.5 to query the database, as well as evaluate the alignment of their institutional data with the accreditation standards. The correctness and relevancy of the responses provided by the RAG model were reported to be optimal, promoting the effectiveness of such systems in automating the manual process of writing the accreditation reports.

Other researchers investigated the use of RAG models in

Ref.	Area	Application	User Group	Data Source
[13]	Teaching and learning	Tutor	Students	course materials (lecture notes, Jupyter notebooks) and the additional readings books
[19]	Teaching and learning	Tutor	Students	course notes related to two quantitative analysis methods: t-test and ANOVA
[24]	Teaching and learning	Tutor	Students	lectures' recording transcript texts with timestamps
[21]	Teaching and learning	Assessment Grading	Instructors	SemEval-2013 joint student response dataset
[25]	Teaching and learning	Question Generation	Instructors	Course textbook
[22]	Teaching and learning	Information Retrieval	Students	A web software development course materials
[15]	Student Services and Administration	Information Retrieval	Students	111 documents from faculty and university websites
[20]	Student Services and Administration	Admission consultation chatbot	students	admission data for undergraduate from three universities
[23]	Student Services and Administration	Consultation chatbot	Students	Students' info (resumes and filled forms)
[5]	Quality and Institutional Effectiveness	Accreditation	Administrators	Institutional data and accreditation standards

Table 1Comparative Analysis of Reviewed Papers [by Focus Application, User Group, and Data Source]

course assessments' auto-grading. [21] proposed a pipeline that stored training responses, from SemEval-2013 joint student response dataset, in a vector database, which got augmented with transformer-based encoders and Claude 3 LLMs. Their approach retrieved semantically similar responses from the vector database and use LLMs to analyze these responses as well as determine appropriate scores. It was further enhanced by fine-tuning the retrieval process along with applying prompt engineering. The evaluation results showed the superiority of RAG-based scoring approach over existing rule-based or deep learning approaches.

A RAG model was utilized in another research paper in automating the creation of question banks from course materials, ensuring the questions were diverse and subjectively relevant to the course domain. The knowledge base for this model was built from the course textbook (i.e. an operating systems textbook). The RAG system tested four leading LLMs—Falcon, Orca, Llama-2, and Google Gemini— for their effectiveness. A Linux expert evaluated 1,400 questions generated by these models, concluding that Google Gemini outperformed the other LLMs in generating high-quality, relevant questions [25].

To conclude this section, it worths to acknowledge that not all institutions welcome the use of GenAI,

LLMs, and RAG systems; some continue to prohibit their usage despite the potential benefits [13].

VI. COMPARATIVE ANALYSIS AND DISCUSSION

Table 1 presents a comparative analysis of the reviewed literature, comparing and categorizing the articles based on several factors: the focus area, specific application within the focus area, target user groups, and external data sources / knowledge bases.

The table indicates that the advent of RAG systems has driven notable efforts toward advancing the capabilities of chatbots, particularly tutoring systems, with the goal of supporting students' learning experiences and improving their interaction and access to information. This highlights a predominantly student-centric focus, with fewer applications targeting instructors and administrators. Such imbalance suggests an opportunity to explore innovative use cases that cater to these underserved stakeholders.

Furthermore, the analysis reveals a lack of research into the applications of RAG systems in areas such as institutional effectiveness, governance, and compliance. This represents a notable gap in the literature that is essential to be addressed to fully leverage the transformative potential of this technology. A promising future direction could involve proposing a framework that utilizes a RAG system for enhancing accessibility, raising awareness level, and posing the adherence level to policies, procedures, and guidelines in higher education.

Regarding the knowledge bases integrated into RAG systems, it is notable that the majority are constructed from relatively small and context-specific data sources—such as course notes, textbooks, lecture transcripts, and institutional data—rather than large and aggregated datasets, highlighting a focus on context-relevant content rather than extensive data aggregation.

Considering that the effectiveness of RAG systems relies heavily on the design and integration of their core components—retrieval, generator, and retrieval fusion, an

Ref.	RAG Component							
	Retrieval			Generator - LLM		Retrieval Fusions		
	Embedding	Vector DB	Similarity search	Open source	Closed source	Query based	Latent	Logits-based
[13]	OpenAI's embedding model	-	Semantic similarity		Anthropic's Claude 3 model	✓		
[19]	OpenAI's text-embedding-ada-003	ChromaDB	Semantic similarity		Open AI's ChatGPT 3.5	✓		
[24]	OpenAI's text-embedding-ada-002	Postgres database with the pgvector extension	-		Open AI's GPT 4	✓		
[21]	-	-	Cosine similarity		Anthropic's Claude 3 model	✓		
[25]	-	-	Semantic similarity	Falcon and Llama-2 by Meta AI	Google Gemini and Orca by Microsoft	✓		
[22]	OpenAI's text-embedding-3-large	Weaviate	-		OpenAI GPT-3.5-Turbo	✓		
[15]	OpenAI Ada	ChromaDB	Cosine similarity		OpenAI GPT-3.5-Turbo	✓		
[20]	-	-	Cosine similarity		Open AI's ChatGPT 3.5	✓		

[23]	-	-	-	Llama-2 by Meta AI		✓		
[5]	OpenAI's text-embedding-ada-002	Neo4j with Apache Lucene	Cosine similarity		Open AI's ChatGPT 3.5	✓		

Table 1Comparative Analysis of Reviewed Papers [by Implementation Details and Technical Aspects]

extensive examination and comparison of the technical aspects and implementation details of these systems was conducted and summarized in the following table. Table2 shed the light on the basic building blocks of a retrieval module, detailing the embedding models employed, vector databases used, and similarity search measures adopted to retrieve relevant content. Additionally, the generator module is analyzed by identifying the LLMs utilized and categorizing them as open-source or closed-source. Lastly, the table explores the retrieval fusion techniques implemented, distinguishing between query-based, latent, and logits-based approaches.

The analysis of the table reveals key insights into the technical aspects and implementations driving the performance and adaptability of RAG systems in higher education domain. One notable finding is that all reviewed systems employed query-based retrieval fusion techniques, where the retrieved content is combined with the original user query before being passed to the generator. This leaves the other two retrieval fusion techniques, i.e. latent and logits-based fusion techniques, unexplored in this domain.

Another key finding is that most reported RAG systems relied on closed-source LLMs, with models such as OpenAI's GPT-3.5, GPT-4, and Anthropic's Claude 3 being prominently used. In contrast, open-source LLMs, such as Falcon and Llama-2, are less commonly used, appearing in only a few studies. This indicates a preference for closed-source models in higher education applications, likely due to their superior pre-trained capabilities, robust performance, and ease of integration, despite the growing potential of open-source alternatives for cost-effective and customizable implementations.

Unfortunately, many research papers lacked implementation details, particularly regarding the specific embedding models, vector databases, and similarity measures employed. For example, while some studies explicitly utilized OpenAI's embedding models (e.g., text-embedding-ada-002 or ada-003) and vector databases like ChromaDB or Weaviate, others did not specify these components. Similarly, many studies simply mentioned "semantic similarity" as the similarity search measure, with few specifying cosine similarity as the specific retrieval measure in use. The absence of such details highlights a gap in transparency and reproducibility within the literature.
In addition to the findings reported in table2, it is worth to mention that all the studies used the same embedding model for both the vector database and the query encoding. This suggests that all the reviewed implementations followed a naive RAG approach. The lack of exploration of other RAG approaches, i.e. advanced RAG and modular RAG, in higher education applications presents an opportunity for future research to enhance the adaptability and efficiency of RAG systems.

These findings highlight the need for more detailed technical reporting in future studies and the potential for innovation by exploring alternative RAG implementations beyond the naive approach currently dominating the field.

VII. CONCLUSION

This work provides comprehensive review of the wide range of applications of Retrieval-Augmented Generation (RAG) systems in the higher education domain. By thoroughly analyzing existing literature, the review sheds light on how RAG systems are being utilized to address challenges in teaching and learning, student services, and institutional effectiveness. The findings indicate remarkable advancements in the field such as enhancing student engagement and improving access to educational resources. Additionally, the findings highlight notable gaps in research, such as the limited focus on governance, compliance, and administrative applications.
The significance of this study lies in its pioneering effort to consolidate knowledge of the adoption of this newly emerging field in the higher education sector, offering a foundation for future research and development. As higher education institutions increasingly adopt generative AI technologies, understanding the technical aspects of RAG systems becomes critical for maximizing their transformative potential.

Future research should address the identified gaps by exploring innovative applications that go beyond the current student-centric focus. Additionally, advancing the technical implementation of RAG systems by experimenting with alternative retrieval fusion techniques, exploring alternative RAG approaches, and leveraging open-source LLMs could unlock new

possibilities for adaptability and cost-effectiveness in the field. This review not only highlights the current state of RAG adoption but also provides a roadmap for utilizing this technology to drive innovation and excellence in higher education.

REFERENCES

[1] X. Hou et al., "Large Language Models for Software Engineering: A Systematic Literature Review," ACM Transactions on Software Engineering and Methodology, Sep. 2024, doi: 10.1145/3695988.

[2] A. Fan et al., "Large Language Models for Software Engineering: Survey and Open Problems," in Proceedings - 2023 IEEE/ACM International Conference on Software Engineering: Future of Software Engineering, ICSE-FoSE 2023, Institute of Electrical and Electronics Engineers Inc., 2023, pp. 31–53. doi: 10.1109/ICSE-FoSE59343.2023.00008.

[3] A. Asbai, "Mitigating Hallucination in Large Language Model Code Generation for Higher Education: An Evaluation of Retrieval Augmented Generation; Mitigating Hallucination in Large Language Model Code Generation for Higher Education: An Evaluation of Retrieval Augmented Generation; Mitigering av Hallucination i Stora Språkmodellers KodgenereringinomHögreUtbildning – En Utvärdering av Retrieval Augmented Generation."

[4] S. S. Sengar, A. Bin Hasan, S. Kumar, and F. Carroll, "Generative artificial intelligence: a systematic review and applications," Multimed Tools Appl, 2024, doi: 10.1007/s11042-024-20016-1.

[5] C. Edwards, "Hybrid Context Retrieval Augmented Generation Pipeline: LLM-Augmented Knowledge Graphs and Vector Database for Accreditation Reporting Assistance," 2024. [Online]. Available: https://github.com/CS-Edwards/advRAG

[6] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAS: Automated Evaluation of Retrieval Augmented Generation," Sep. 2023, [Online]. Available: http://arxiv.org/abs/2309.15217

[7] S. Wu et al., "Retrieval-Augmented Generation for Natural Language Processing: A Survey," Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.13193

[8] L. Yan et al., "Practical and ethical challenges of large language models in education: A systematic scoping review," Jan. 01, 2024, John Wiley and Sons Inc. doi: 10.1111/bjet.13370.

[9] Y. Hu, Z. Lei, Z. Zhang, B. Pan, C. Ling, and L. Zhao, "GRAG: Graph Retrieval-Augmented Generation."

[10] Y. Gao et al., "Retrieval-Augmented Generation for Large Language Models: A Survey," Dec. 2023, [Online]. Available: http://arxiv.org/abs/2312.10997

[11] C. Long, Y. Liu, C. Ouyang, and Y. Yu, "Bailicai: A Domain-Optimized Retrieval-Augmented Generation Framework for Medical Applications," Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.21055

[12] B. Zhang, H. Yang, T. Zhou, M. Ali Babar, and X. Y. Liu, "Enhancing Financial Sentiment Analysis via Retrieval Augmented Large Language Models," ICAIF 2023 - 4th ACM International Conference on AI in Finance, pp. 349–356, Nov. 2023, doi: 10.1145/3604237.3626866.

[13] M. Thway, J. Recatala-Gomez, S. Lim, K. Hippalgaonkar, and L. W. T. Ng, "Battling Botpoop using GenAI for Higher Education: A Study of a Retrieval Augmented Generation Chatbot's Impact on Learning."

[14] H. Yu, Z. Yang, K. Pelrine, J. F. Godbout, and R. Rabbany, "Open, Closed, or Small Language Models for Text Classification?", Accessed: Dec. 18, 2024. [Online]. Available: https://the-decoder.com/gpt-4-is-1-76-trillion-

[15] M. Maryamah, M. M. Irfani, E. B. Tri Raharjo, N. A. Rahmi, M. Ghani, and I. K. Raharjana, "Chatbots in Academia: A Retrieval-Augmented Generation Approach for Improved Efficient Information Access," in KST 2024 - 16th International Conference on Knowledge and Smart Technology, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 259–264. doi: 10.1109/KST61284.2024.10499652.

[16] P. Zhao et al., "Retrieval-Augmented Generation for AI-Generated Content: A Survey," Feb. 2024, [Online]. Available: http://arxiv.org/abs/2402.19473

[17] S.-Q. Yan, J.-C. Gu, Y. Zhu, and Z.-H. Ling, "Corrective Retrieval Augmented Generation," Jan. 2024, [Online]. Available: http://arxiv.org/abs/2401.15884

[18] H. Yu, A. Gan, K. Zhang, S. Tong, Q. Liu, and Z. Liu, "Evaluation of Retrieval-Augmented Generation: A Survey," May 2024, [Online]. Available: http://arxiv.org/abs/2405.07437

[19] Y. Lee, "Developing a computer-based tutor utilizing Generative Artificial Intelligence (GAI) and Retrieval-Augmented Generation (RAG)," Educ Inf Technol (Dordr), 2024, doi: 10.1007/s10639-024-13129-5.

[20] Z. Chen et al., "Facilitating university admission using a chatbot based on large language models with retrieval-augmented generation," Educational Technology & Society, vol. 27, no. 4, pp. 454–470, 2024, doi: 10.2307/48791566.

[21] Z. Wang and C. Ormerod, "Generative Language Models with Retrieval Augmented Generation for Automated Short Answer Scoring," Aug. 2024, [Online]. Available: http://arxiv.org/abs/2408.03811

[22] L. Pasquarelli, C. Koutcheme, and A. Hellas, "Comparing the Utility, Preference, and Performance of Course Material Search Functionality and Retrieval-Augmented Generation Large Language Model (RAG-LLM) AI Chatbots in Information-Seeking Tasks," Oct. 2024, [Online]. Available: http://arxiv.org/abs/2410.13326

[23] A. D. Singla, S. Tripathi, and A. H. Victoria, "HICON AI: Higher Education Counseling Bot," in Proceedings - 2024 4th International Conference on Pervasive Computing and Social Networking, ICPCSN 2024, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 779–784. doi: 10.1109/ICPCSN62568.2024.00131.

[24] S. Jacobs and S. Jaschke, "Leveraging Lecture Content for Improved Feedback: Explorations with GPT-4 and Retrieval Augmented Generation," May 2024, doi: 10.1109/CSEET62301.2024.10663001.

[25] Sayed, M. Kanojia, and S. Nabajja, "Advanced Subjective Question Bank Generation Using Retrieval Augmented Generation Architecture," International Journal of Computer Information Systems and Industrial Management Applications, vol. 16, pp. 264–277, 2024.

★★★