LearnRAG: Implementing Retrieval-Augmented Generation for Adaptive Learning Systems

Richard Shan
Department of Data Science
North Carolina School of Science and Mathematics
Durham, NC, USA
m@richardshan.com

***Abstract*—The rapid advancements in large language models have revolutionized natural language processing, yet their static knowledge bases limit their applicability in dynamic, domain-specific, and personalized contexts. Retrieval-Augmented Generation systems address this challenge by integrating retrieval mechanisms with generative models to deliver real-time, contextually enriched responses. This paper implements LearnRAG, an open-source RAG framework for personalized learning that is modular in architecture, hybrid in retrieval, and fine-tuned for generation to produce adaptive educational content. A holistic case study of LearnRAG showed scalability, efficiency, increasing learner engagement, and reducing educators' workload. Issues such as multimodal integration, content accuracy, and learning styles are discussed, and strategies for ethical deployment are developed. LearnRAG offers a robust, scalable, and adaptive platform to meet the evolving needs of learners and educators worldwide, representing a paradigm shift in GenAI-driven education.**

Keywords—Retrieval-Augmented Generation (RAG), large language models (LLMs), Natural Language Processing (NLP), open source, architecture, system design, vector database, orchestration, retrieval, implementation, personalized learning.

I. Introduction

The exponential growth in the capabilities of large language models (LLMs) has marked a transformative era in natural language processing (NLP). These models exhibit exceptional fluency in generating human-like text, revolutionizing tasks ranging from conversational AI to complex decision support systems. However, despite their prowess, LLMs are inherently constrained by the fixed nature of their pretraining knowledge base, which is static and limited to the information available at the time of training [1]. Consequently, LLMs struggle to deliver up-to-date, domain-specific, or contextualized responses in dynamic or specialized applications. Addressing these challenges has given rise to Retrieval-Augmented Generation (RAG) systems, which synergize the strengths of information retrieval and generative modeling [2].

Among applicable domains for RAG, personalized learning stands out as a field ripe for innovation. Tailoring educational content to individual learners' profiles—factoring in their unique needs, interests, and progression—has demonstrated significant potential to improve engagement, comprehension, and academic outcomes. RAG systems enable such customization by dynamically generating educational materials that align with a learner's specific context, creating more meaningful and effective learning experiences [3].

Despite the promise of RAG systems in personalized learning, the landscape of available solutions reveals critical gaps. While proprietary systems have demonstrated success in integrating retrieval and generation capabilities, comprehensive open-source architectures tailored for educational applications remain scarce. Existing solutions often lack the modularity and scalability necessary for robust adaptation [4], and the absence of detailed documentation poses a barrier for developers and researchers seeking to customize these systems. Furthermore, achieving seamless integration between retrieval mechanisms and generative models presents a significant technical hurdle, particularly for resource-constrained environments [5].

This paper seeks to address these limitations by introducing LearnRAG, an open-source RAG framework specifically designed to enhance personalized learning. The key contributions of this work include:

1. Architecture Design: A modular and scalable framework integrating state-of-the-art retrieval and generation techniques.
2. Technical Components: Comprehensive documentation of each system module, including the retriever, generator, indexing subsystem, user interface, and orchestration layer.
3. Practical Implementation: A case study illustrating the deployment of LearnRAG in an online learning platform, demonstrating its capacity to deliver adaptive and engaging educational content.
4. Impact Analysis: Evaluation of LearnRAG's effectiveness in improving learner engagement and outcomes.

II. Background and Related Work

A. Overview of Retrieval-Augmented Generation

The field of natural language processing (NLP) has been transformed by the advent of pre-trained large language models such as BERT [6], GPT-4, and T5. These models have demonstrated remarkable abilities in understanding and generating human-like text across a variety of tasks. However, a significant limitation of these models is their fixed knowledge base; they can only generate information present in their training

data up to a certain cutoff date. This constraint poses challenges when dealing with real-time information, domain-specific knowledge, or personalized content requirements.

Retrieval-Augmented Generation addresses this limitation by integrating information retrieval mechanisms with generative models. In a RAG system, the model retrieves relevant external information from a knowledge base or document corpus in response to a query or input context. This retrieved information is then used to condition the generative model, allowing it to produce responses that are both contextually appropriate and enriched with up-to-date or domain-specific knowledge. The RAG framework operates in two main stages:

Retrieval Stage: Given an input query, the retriever searches a large corpus to find documents or passages that are relevant to the query. This can involve traditional sparse retrieval methods like TF-IDF and BM25, or dense retrieval techniques using neural embeddings.

Generation Stage: The generator, typically a sequence-to-sequence model, takes the retrieved documents along with the original query to produce a final response. The generator leverages the external knowledge to enhance the quality and relevance of the output.

By combining retrieval with generation, RAG systems can produce responses that are more informative, accurate, and contextually appropriate, making them suitable for applications such as question answering, dialogue systems, and personalized content delivery.

B. Existing RAG Systems

Several RAG systems have been proposed and implemented, each contributing to the evolution of this field:

RAG Model: This pioneering work [1] introduced the Retrieval-Augmented Generation model, which combines a neural retriever with a sequence-to-sequence generator. The retriever uses dense embeddings to fetch relevant documents, and the generator produces answers conditioned on these documents.

REALM (Retrieval-Augmented Language Model Pre-Training): REALM [7] integrates retrieval into the pre-training process of language models. It enables models to retrieve and incorporate factual knowledge dynamically during both training and inference.

FIT-RAG: FIT-RAG utilizes the factual information by constructing a bi-label document scorer. It reduces the tokens by introducing a self-knowledge recognizer and a sub-document-level token reducer. FIT-RAG achieves both superior effectiveness and efficiency, which is validated by extensive experiments across three open-domain question-answering datasets: TriviaQA, NQ and PopQA [8].

Blended RAG: Blended RAG leverages semantic search techniques, such as Dense Vector indexes and Sparse Encoder indexes, blended with hybrid query strategies. It achieves better retrieval results and sets new benchmarks for IR (Information Retrieval) datasets like NQ and TREC-COVID datasets [9].

RQ-RAG: RQ-RAG attempts to enhance the RAG model by equipping it with capabilities for explicit rewriting, decomposition, and disambiguation, looking into the nuances of ambiguous or complex queries that necessitate further clarification or decomposition for accurate responses [10].

RAGSys: RAGSys explores In-Context Learning (ICL) retrieval that resembles item-cold-start recommender systems, prioritizing discovery and maximizing information gain over strict relevance [11].

IM-RAG: IM-RAG integrates information retrieval systems with LLMs to support multi-round RAG through learning Inner Monologues (IM, i.e., the human inner voice that narrates one's thoughts), achieving state-of-the-art (SOTA) performance with the HotPotQA dataset [12].

Speculative RAG: Speculative RAG is a framework that leverages a larger generalist language model to efficiently verify multiple RAG drafts produced in parallel by a smaller, distilled specialist language model. It achieves state-of-the-art performance with reduced latency on TriviaQA, MuSiQue, PubHealth, and ARC-Challenge benchmarks [13].

Stochastic RAG: Stochastic RAG casts the retrieval process in RAG as a stochastic sampling without replacement process. Through this formulation, it employs straight-through Gumbel-top-k that provides a differentiable approximation for sampling without replacement and enables effective end-to-end optimization for RAG [14].

KG-RAG: The Knowledge Graph-Retrieval Augmented Generation pipeline enhances the knowledge capabilities of language model agents by integrating structured Knowledge Graphs with the functionalities of LLMs, thereby significantly reducing the reliance on the latent knowledge of LLMs. It demonstrates notable improvements in the reduction of hallucinated content, for the ComplexWebQuestions dataset [15].

TurboRAG: TurboRAG redesigns the inference paradigm of the current RAG system by first pre-computing and storing the key-value (KV) caches of documents offline, and then directly retrieving the saved KV cache for prefill [16].

MMed-RAG: MMed-RAG is a versatile multimodal RAG system designed to enhance the factuality of Med-LVLMs, introducing a domain-aware retrieval mechanism, an adaptive retrieved contexts selection method, and a provable RAG-based preference fine-tuning strategy [17].

Despite these advancements, existing RAG systems face several limitations: Complexity and Scalability: Many systems require extensive computational resources, making them challenging to scale for large datasets or real-time applications; Integration Challenges: Combining retrieval and generation components often involves intricate engineering, and changes in one component may necessitate adjustments in others; Domain Adaptation: Adapting existing RAG systems to specialized domains like personalized learning requires significant customization, including retraining models and adjusting retrieval strategies; Open-source Accessibility: While some implementations are available, comprehensive open-source RAG architectures with detailed documentation and modular design are limited, hindering widespread adoption and collaborative development.

C. Open-source Initiatives

Open-source projects play a crucial role in advancing RAG research by providing accessible implementations for experimentation and extension. Notable open-source initiatives include:

Hugging Face's Transformers Library: Hugging Face provides an implementation of RAG models within their Transformers library [18]. This includes pre-trained models and tools for fine-tuning on custom datasets. However, the integration between retrieval and generation can be complex for newcomers.

Haystack: Haystack [19] is an open-source NLP framework that supports building end-to-end RAG pipelines. It offers modular components for retrieval, including sparse and dense retrievers, and supports various generators. Haystack is designed for question answering and search applications but can be adapted for other use cases.

Open-RAG: Open-RAG is designed to enhance reasoning capabilities in RAG with open-source LLMs, transforming an arbitrary dense LLM into a parameter-efficient sparse mixture of experts (MoE) model capable of handling complex reasoning tasks, including both single- and multi-hop queries [20].

Knowledge-Enabled Language Generation: Some projects explore integrating external knowledge bases into language generation, such as using Wikidata or ConceptNet. These efforts contribute to the broader field of knowledge-enhanced NLP but may not provide complete RAG architectures.

III. System Architecture

The limitations of previous initiatives often stem from a lack of comprehensive, domain-specific solutions that are easily adaptable. There is a need for an open-source RAG architecture: Modular: Allowing individual components (retriever, generator, indexer) to be developed, replaced, or improved independently; Scalable: Capable of handling large datasets and real-time processing requirements; Well-documented: Providing detailed instructions and examples to facilitate adoption by the research and development community; Domain-adaptable: Designed to be easily customized for specialized applications, such as personalized learning, without extensive re-engineering.

This study seeks to address these gaps by presenting an open-source RAG system architecture tailored for personalized learning applications, complete with implementation details, evaluation results, and insights from a real-world case study.

The proposed Retrieval-Augmented Generation system, LearnRAG, is designed to address the challenges of delivering personalized educational content at scale. This architecture is built around modularity, scalability, and adaptability, enabling efficient integration with various learning platforms. The LearnRAG system comprises five core components: Generator, User Interface, Indexing Module, Retriever, and Orchestration Layer. Each module is designed to work independently yet cohesively, ensuring a robust and flexible framework for personalized learning.

The User Interface serves as the primary interaction point for learners and educators, providing a seamless and engaging experience. It enables learners to input queries, receive personalized content, and track their progress. The UI is designed to support multiple platforms, including web and mobile applications, and is equipped with accessibility features such as screen reader compatibility and adjustable text sizes.

The orchestration layer facilitates communication between the various modules, ensuring smooth data flow and system operation. This layer acts as the backbone of the architecture, managing the sequence of operations, error handling, and performance optimization. It employs microservices principles, enabling independent scaling and maintenance of individual components.

The retriever module is responsible for fetching relevant educational materials based on learner queries and profiles. It employs a hybrid approach that combines sparse retrieval methods (e.g., BM25, TF-IDF) with dense retrieval techniques using neural embeddings. This combination ensures both efficiency and contextual accuracy. The module also integrates personalization strategies, such as re-ranking results based on learner profiles and preferences.

Efficient indexing is crucial for managing and retrieving vast amounts of educational content. The indexing module utilizes both inverted indexes for sparse retrieval and vector-based indexes for dense retrieval. These indexes are built using scalable solutions like FAISS and are optimized for handling high-dimensional embeddings. The module also supports dynamic updates, ensuring that the content repository remains current and relevant.

The generator module leverages state-of-the-art large language models (e.g., GPT-4, T5) to produce personalized educational content. By incorporating retrieved documents and learner-specific attributes, the generator creates contextually accurate and pedagogically appropriate outputs. Fine-tuned on domain-specific educational datasets, this module ensures that the generated content aligns with curriculum standards and individual learning objectives. A high-level schematic of the LearnRAG architecture is illustrated in Figure 1.

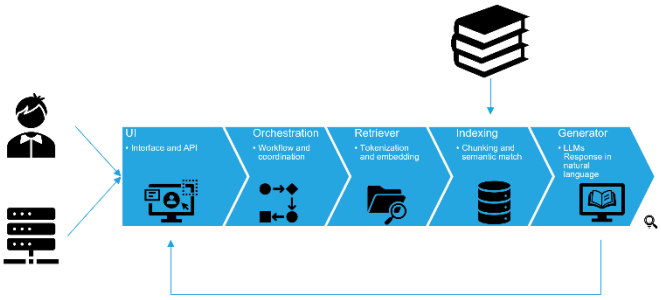

Figure 1. LearnRAG Architecture

IV. Technical Components

The LearnRAG system's technical components form the backbone of its ability to deliver personalized educational content efficiently and accurately. These components encompass advanced techniques for retrieval, generation, indexing, and scalability, each designed to address the unique challenges of integrating dynamic and adaptive learning systems.

The retrieval module is central to the system's ability to fetch relevant educational content. It combines sparse and dense retrieval techniques to balance efficiency and contextual understanding. Sparse retrieval methods, such as Term Frequency-Inverse Document Frequency (TF-IDF) and BM25, rely on keyword matching to rank documents by relevance. These algorithms are computationally efficient, making them well-suited for processing large corpora. To enhance semantic understanding, the system employs dense retrieval, which uses neural embeddings to map both queries and documents into a shared vector space. Models like Sentence-BERT (SBERT) and Dense Passage Retrieval (DPR) are fine-tuned on educational datasets to ensure that retrieval results align with the system's instructional goals. Additionally, approximate nearest neighbor (ANN) search, implemented through tools such as FAISS, ensures efficient similarity searches in high-dimensional embedding spaces. The retriever also incorporates learner-specific personalization by integrating user profiles into the retrieval process, thereby tailoring content to individual needs and preferences.

The generation module is responsible for transforming retrieved content into coherent and contextually appropriate educational material. This module leverages state-of-the-art pre-trained language models, including GPT-4 and T5, for their advanced natural language understanding and generation capabilities. To adapt these models for educational purposes, fine-tuning is performed using domain-specific datasets that include explanations, problem sets, and common misconceptions. Personalization is further achieved by conditioning generation on learner attributes, such as reading levels, interests, and prior knowledge. The generator is designed to interact seamlessly with retrieved documents, ensuring that the outputs are both accurate and grounded in factual information. To minimize errors and hallucinations, the system incorporates automated validation and quality assurance mechanisms, further enhancing the reliability of the generated content.

Efficient indexing and storage solutions are crucial for maintaining the performance of the retrieval system. The LearnRAG architecture employs a dual indexing strategy, utilizing inverted indexes for sparse retrieval and vector-based indexes for dense retrieval. These indexes, built using scalable tools like Chroma and FAISS, enable the system to handle high-dimensional embeddings and facilitate rapid lookups. The content repository undergoes extensive preprocessing, including text normalization, tokenization, and metadata enrichment. Metadata annotations—such as subject area, difficulty level, and curriculum alignment—ensure that content is categorized and retrievable with precision. The system also supports dynamic updates, allowing new content to be seamlessly integrated into the repository without disrupting ongoing operations.

Scalability and performance optimization are key design principles in LearnRAG. The system adopts a distributed architecture based on microservices, enabling independent scaling of retrieval, generation, and indexing components. Containerization technologies, such as Docker and Kubernetes, further enhance resource management and deployment flexibility. To handle high user volumes and large datasets, the system employs caching layers using tools like Redis to store frequently accessed data and intermediate results. Load balancers distribute incoming requests evenly across servers, preventing bottlenecks and ensuring consistent performance. Additionally, asynchronous processing mechanisms, including message queues like Apache Kafka, improve the system's ability to manage tasks efficiently, particularly during peak usage periods. Model optimization techniques, such as quantization, pruning, and knowledge distillation, further enhance scalability by reducing computational overhead without compromising output quality.

Ensuring data privacy and compliance with regulations is paramount in LearnRAG's design. The system employs robust data protection measures, including anonymization techniques to remove personally identifiable information and encryption protocols like SSL/TLS for secure data transmission. Consent management features are implemented to ensure user compliance with data collection and processing. The system adheres to regulatory standards, such as GDPR and COPPA, through rigorous data handling policies and regular security audits. By integrating these measures, LearnRAG safeguards learner data while maintaining ethical and legal accountability.

In general, the LearnRAG system's technical components work cohesively to provide a robust, scalable, and secure platform for personalized learning. Its integration of advanced retrieval and generation methods, combined with efficient indexing, scalability strategies, and stringent data privacy measures, ensures a seamless and impactful educational experience for learners and educators alike. These components position LearnRAG as a state-of-the-art solution capable of addressing the evolving demands of personalized learning environments.

V. Case Study: Implementation in Personalized Learning

To demonstrate the capabilities of LearnRAG, a case study was conducted by integrating the system into an online personalized learning platform. This implementation aimed to assess the system's ability to deliver customized educational content, optimize user engagement, and meet the specific requirements of a diverse learner base. The deployment involved collaboration between technical experts, educators, and domain specialists to ensure the system aligned with both technical and pedagogical objectives.

The integration process began with a thorough analysis of the platform's existing infrastructure and its compatibility with LearnRAG's architecture. The modular nature of LearnRAG facilitated a seamless integration, as its components could be independently deployed and connected to the platform's backend services. The retriever module was configured to access the platform's educational repository, which consisted of over one million documents spanning various subjects and difficulty levels. These documents were preprocessed to create both sparse and dense indexes, enabling efficient and accurate retrieval. Metadata enrichment during preprocessing ensured alignment with curriculum standards and learner profiles.

The generator module was fine-tuned using a combination of publicly available and proprietary educational datasets to create content aligned with the platform's instructional goals.

The fine-tuning process focused on adapting the language model to produce outputs suitable for learners of varying ages and competencies. For instance, the generator was trained to simplify complex topics for younger students while maintaining depth and rigor for advanced learners. Moreover, the personalization capabilities of LearnRAG were utilized to tailor responses based on individual learner profiles, which included data on prior knowledge, learning objectives, and engagement patterns. This personalized approach enhanced the relevance and quality of the generated content, fostering a more engaging learning experience.

A critical aspect of the implementation was the system's ability to operate efficiently under real-world conditions. To evaluate this, LearnRAG was tested in a live environment with thousands of concurrent users accessing the platform. Scalability was ensured by deploying the system using containerized microservices, allowing individual components to scale independently based on demand. Caching mechanisms reduced latency by storing frequently retrieved data, while load balancers ensured an even distribution of requests across servers. These optimizations enabled the system to deliver responses in under two seconds, even during peak usage periods.

The effectiveness of LearnRAG was evaluated through both quantitative metrics and qualitative feedback from users. Key performance indicators included response accuracy, relevance of content, and user engagement metrics such as session duration and completion rates. Learners reported high levels of satisfaction with the system's ability to provide clear, concise, and personalized explanations. Educators noted that the system's contextual understanding and adaptability significantly reduced the time required to create tailored learning materials, allowing them to focus more on direct interaction with students.

A significant challenge encountered during implementation was ensuring the factual accuracy of generated content. To address this, an automated fact-checking pipeline was integrated into the system. This pipeline cross-referenced generated outputs with authoritative sources, flagging inaccuracies for review. Additionally, iterative feedback loops involving educators and subject matter experts were established to refine the system's performance. These measures ensured that the content delivered by LearnRAG met high standards of accuracy and reliability.

The case study highlighted several advantages of LearnRAG over traditional learning systems. The combination of advanced retrieval techniques and adaptive generation allowed the system to deliver highly relevant content tailored to individual learners. Its scalability and efficiency made it suitable for large-scale deployments, while its modular design provided flexibility for future enhancements. However, the study also revealed areas for improvement, such as the need for enhanced multilingual support and better integration with non-textual educational resources like videos and simulations.

Essentially, the implementation of LearnRAG demonstrated its potential to transform personalized learning through innovative use of retrieval and generation technologies. By addressing both technical and pedagogical challenges, the system proved to be a scalable, efficient, and impactful solution for modern educational platforms. This case study serves as a foundation for further exploration and development of LearnRAG in diverse learning contexts.

VI. Discussion

The LearnRAG system represents a significant advancement in the application of Retrieval-Augmented Generation technologies to personalized learning. By addressing the limitations of static language models and integrating dynamic retrieval with adaptive generation, the system delivers a solution that is not only technically robust but also pedagogically transformative. However, the development and deployment of LearnRAG revealed both strengths and areas for improvement, underscoring the complexity of balancing technical innovation with educational impact.

One of the most notable strengths of LearnRAG lies in its ability to seamlessly combine retrieval and generation processes. The hybrid retrieval approach, which integrates sparse and dense retrieval methods, ensures that the system can efficiently access relevant and contextually accurate information from vast repositories. Coupled with a fine-tuned generator, this capability allows LearnRAG to produce highly customized educational content that meets the unique needs of individual learners. The system's ability to incorporate learner profiles into both retrieval and generation processes further enhances its effectiveness, fostering a level of personalization that is difficult to achieve with traditional learning systems.

The modular and scalable architecture of LearnRAG also proved to be a key advantage. By employing containerized microservices, the system can adapt to varying levels of demand, making it suitable for both small-scale and large-scale deployments. This flexibility is particularly valuable in educational contexts, where user volumes can fluctuate significantly based on factors such as time of year or specific course offerings. The system's design also facilitates ongoing innovation, as individual components can be updated or replaced without disrupting the overall functionality. This ensures that LearnRAG remains adaptable to future advancements in retrieval and generation technologies.

Despite its strengths, the implementation of LearnRAG also highlighted several challenges and limitations. Ensuring the factual accuracy of generated content emerged as a critical issue, particularly given the high stakes associated with educational materials. While the integration of an automated fact-checking pipeline and iterative feedback loops helped mitigate this problem, achieving perfect accuracy remains an ongoing challenge. Additionally, the system's reliance on high-quality, domain-specific datasets for fine-tuning introduces potential bottlenecks in terms of scalability and accessibility. Developing and curating these datasets require significant time and resources, which may limit the ability to deploy LearnRAG in less resourced educational contexts.

Another area for improvement is the system's handling of non-textual educational resources. While LearnRAG excels at processing and generating textual content, its integration with multimedia resources such as videos, images, and simulations is currently limited. Given the increasing importance of multimodal learning in modern education, enhancing the system's capabilities in this area would significantly broaden its

applicability and impact. Similarly, while the system supports multiple languages, its performance in non-English contexts lags behind due to the limited availability of high-quality multilingual datasets. Addressing this gap would enable LearnRAG to better serve diverse learner populations worldwide.

The ethical implications of deploying a system like LearnRAG in educational settings also warrant careful consideration. Issues such as data privacy, bias in content generation, and the potential for over-reliance on automated systems must be addressed to ensure that the system is used responsibly and equitably. By incorporating robust privacy protections, rigorous bias mitigation strategies, and clear guidelines for educators, these risks can be minimized, enabling LearnRAG to serve as a positive force in education.

In short, the LearnRAG system demonstrates immense potential to transform personalized learning by leveraging the strengths of Retrieval-Augmented Generation technologies. Its ability to deliver tailored, contextually accurate educational content makes it a powerful tool for enhancing learner engagement and outcomes. However, realizing its full potential will require ongoing innovation and refinement, particularly in areas such as accuracy, multimodal integration, and ethical implementation. By addressing these challenges, LearnRAG can continue to advance the field of personalized learning, setting a new standard for the use of artificial intelligence in education.

VII. Conclusion

The development of LearnRAG highlights the transformative potential of Retrieval-Augmented Generation systems in advancing personalized learning. By seamlessly integrating retrieval and generation processes, LearnRAG addresses key limitations of static language models, delivering tailored, accurate, and contextually relevant educational content to learners. Its modular and scalable architecture ensures that the system can adapt to the diverse and dynamic needs of modern educational platforms, making it a robust solution for large-scale deployments.

LearnRAG's implementation demonstrated its ability to enhance learner engagement, improve instructional delivery, and reduce the workload for educators. Through fine-tuning on domain-specific datasets and the incorporation of learner profiles, the system achieved a high level of personalization that aligns with individual learning goals. These strengths position LearnRAG as a powerful tool for transforming traditional educational approaches, enabling more dynamic and interactive learning experiences.

In a nutshell, LearnRAG represents a significant step forward in the application of artificial intelligence to education. Its innovative architecture, advanced technical capabilities, and focus on personalization make it a powerful tool for enhancing learning experiences. While challenges remain, the ongoing development of LearnRAG holds the promise of shaping the future of education, making learning more accessible, engaging, and effective for diverse populations around the world. By continuing to refine and expand the system, LearnRAG can play a pivotal role in redefining the possibilities of personalized education.

References

[1] Raiaan, M. A. K., Mukta, M. S. H., Fatema, K., Fahad, N. M., Sakib, S., Mim, M. M. J., ... & Azam, S. (2024). A review on large Language Models: Architectures, applications, taxonomies, open issues and challenges. *IEEE Access*.

[2] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., ... & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33, 9459-9474.

[3] Guo, K., & Li, D. (2024). Understanding EFL students' use of self-made AI chatbots as personalized writing assistance tools: A mixed methods study. *System*, 124, 103362.

[4] David, I., Latifaj, M., Pietron, J., Zhang, W., Ciccozzi, F., Malavolta, I., ... & Hebig, R. (2023). Blended modeling in commercial and open-source model-driven software engineering tools: A systematic study. *Software and Systems Modeling*, 22(1), 415-447.

[5] Hadi, M. U., Al Tashi, Q., Shah, A., Qureshi, R., Muneer, A., Irfan, M., ... & Shah, M. (2024). Large language models: a comprehensive survey of its applications, challenges, limitations, and future prospects. TechRxiv. August 12, 2024. doi:10.36227/techrxiv.23589741.v6

[6] Wang, H., Li, J., Wu, H., Hovy, E., & Sun, Y. (2023). Pre-trained language models and their applications. Engineering, 25, 51-65.

[7] Guu, K., Lee, K., Tung, Z., Pasupat, P., & Chang, M. (2020, November). Retrieval augmented language model pre-training. In *International conference on machine learning* (pp. 3929-3938). PMLR.

[8] Mao, Y., Dong, X., Xu, W., Gao, Y., Wei, B., & Zhang, Y. (2024). FIT-RAG: Black-Box RAG with Factual Information and Token Reduction. arXiv:2403.14374.

[9] Sawarkar, K., Mangal, A., & Solanki, S. R. (2024). Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. arXiv:2404.07220.

[10] Chan, C. M., Xu, C., Yuan, R., Luo, H., Xue, W., Guo, Y., & Fu, J. (2024). Rq-rag: Learning to refine queries for retrieval augmented generation. arXiv:2404.00610.

[11] Contal, E., & McGoldrick, G. (2024). RAGSys: Item-Cold-Start Recommender as RAG System. arXiv:2405.17587.

[12] Yang, D., Rao, J., Chen, K., Guo, X., Zhang, Y., Yang, J., & Zhang, Y. (2024). Im-rag: Multi-round retrieval-augmented generation through learning inner monologues. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 730-740).

[13] Wang, Z., Wang, Z., Le, L., Zheng, H. S., Mishra, S., Perot, V., ... & Pfister, T. (2024). Speculative RAG: Enhancing Retrieval Augmented Generation through Drafting. arXiv:2407.08223.

[14] Zamani, H., & Bendersky, M. (2024). Stochastic RAG: End-to-End Retrieval-Augmented Generation through Expected Utility Maximization. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval* (pp. 2641-2646).

[15] Sanmartin, D. (2024). KG-RAG: Bridging the Gap Between Knowledge and Creativity. arXiv:2405.12035.

[16] Lu, S., Wang, H., Rong, Y., Chen, Z., & Tang, Y. (2024). TurboRAG: Accelerating Retrieval-Augmented Generation with Precomputed KV Caches for Chunked Text. arXiv:2410.07590.

[17] Xia, P., Zhu, K., Li, H., Wang, T., Shi, W., Wang, S., ... & Yao, H. (2024). MMed-RAG: Versatile Multimodal RAG System for Medical Vision Language Models. arXiv:2410.13085.

[18] Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., ... & Rush, A. M. (2020). Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 conference on empirical methods in natural language processing*: system demonstrations (pp. 38-45).

[19] Deepset (2024), Haystack: An Open Source NLP Framework. Available: https://haystack.deepset.ai/. Accessed: Oct. 1, 2024.

[20] Islam, S. B., Rahman, M. A., Hossain, K. S. M., Hoque, E., Joty, S., & Parvez, M. R. (2024). OPEN-RAG: Enhanced Retrieval-Augmented Reasoning with Open-Source Large Language Models. arXiv:2410.01782.