International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Fine-Tuning Pre-Trained Language Models for Improved Retrieval in RAG Systems for Domain-Specific Use

Syed Arham Akheel

Senior Solutions Architect, Data Science Dojo Bellevue,WA
arhamakheel@yahoo.com

Abstract
Large Language Models (LLMs) have significantly advanced natural language understanding and generation capabilities, but domain-specific applications often necessitate supplementation with current, external information to mitigate knowledge gaps and reduce hallucinations. Retrieval-Augmented Generation (RAG) has emerged as an effective solution, dynamically integrating up-to-date information through retrieval mechanisms. Fine-tuning pre-trained LLMs with domain-specific data to optimize retrieval queries has become an essential strategy to enhance RAG systems, especially in ensuring the retrieval of highly relevant information from vector databases for response generation. This paper provides a comprehensive review of the literature on the fine-tuning of LLMs to optimize retrieval processes in RAG systems. We discuss advancements such as Query Optimization, Retrieval-Augmented Fine Tuning (RAFT), Retrieval-Augmented Dual Instruction Tuning (RA-DIT), as well as frameworks like RALLE, DPR, and the ensemble of retrieval based and generation-based systems, that enhance the synergy between retrievers and LLMs.
Index Terms—Retrieval-Augmented Generation, Large Language Models, Domain-Specific Fine-Tuning, Information Retrieval, RAFT, RA-DIT

I. INTRODUCTION

Large Language Models (LLMs) have significantly advanced natural language understanding and generation capabilities [1]. However, domain-specific applications often require models to be supplemented with current, external information to address knowledge gaps and reduce hallucinations [2]. Retrieval-Augmented Generation (RAG) provides an effective means of dynamically integrating up-to-date information through retrieval systems. Fine-tuning pre-trained language models is a crucial step to adapt LLMs to these retrieval augmented systems, especially for domain-specific tasks where specialized knowledge is essential.

A persistent challenge with current non-fine-tuned models lies in their inability to effectively comprehend domain specific abbreviations and terminology. These models, though impressive in their general language understanding, often stumble when faced with specialized jargon that is essential in particular fields. For instance, a healthcare provider might use the abbreviation ”HTN” to refer to hypertension—a common medical term. However, without fine-tuning on domain specific corpora, the model may fail to recognize ”HTN” in its proper context, potentially interpreting it as a novel or unrelated acronym. This misunderstanding not only results in irrelevant responses but could also pose serious risks when the

information conveyed is critical. Similar issues arise in legal, financial, or technical domains, where specialized lexicons form the cornerstone of effective communication. Such shortcomings illustrate the need for tailored fine-tuning, equipping LLMs with the ability to decode specialized language nuances and better serve the intricate needs of domain experts. In addition to the struggle with domain-specific abbreviations and terminology, non-fine-tuned models face other significant challenges. One such challenge is maintaining consistency across long conversations or documents. When handling extensive dialogue or complex documents, these models may produce inconsistent answers, particularly when earlier parts of the conversation introduce subtle changes or new context that the model fails to track effectively. Another challenge is dealing with ambiguous terms that require domain-specific disambiguation. For instance, the term ”BP” could refer to ”blood pressure” in a medical context or ”British Petroleum” in an energy industry context. Without proper fine-tuning, non-specialized models may incorrectly interpret these ambiguous terms, leading to irrelevant or misleading responses.

Moreover, non-fine-tuned models often struggle to incorporate up-to-date information in domains that are highly dynamic, such as legal regulations, medical guidelines, or financial markets. Since their training data may not include the latest updates, these models are prone to providing outdated or incomplete information. Fine-tuning with more recent and specialized datasets helps bridge this gap, enabling the models to generate responses that are timely and contextually relevant.

These challenges collectively underscore the necessity of fine-tuning LLMs, particularly for applications where domain specificity, consistency, and accuracy are crucial for effective communication.

A RAG system is a carefully orchestrated interplay of two fundamental components: the retriever and the generator. The retriever acts as an emissary of knowledge, venturing into a vast sea of external databases or vector stores to bring back the most pertinent documents or snippets of information. It ensures that the LLM is armed with the right context by employing techniques like dense passage retrieval [3] to zero in on content-rich materials. Meanwhile, the generator—typically an LLM of formidable prowess—accepts the retrieved documents and, along with the user input, generates a response. The generator does not merely echo the retrieved data; rather, it interlaces this information into a nuanced, coherent response, ensuring that the answer is both context-aware and deeply informed. This harmonious union of retrieval and generation significantly reduces hallucinations and elevates the quality of responses, much like how a skilled doctor uses all available tools to provide the best care.

However, to truly flourish in domain-specific environments, LLMs must not only be large but also precisely tuned. Finetuning within the RAG framework is akin to a rigorous education—imparting a specialized form of knowledge that teaches the model not only what to say but when and how to say it. Fine-tuning helps align the model’s internal representation to the nuances of domain-specific information. This alignment means that when the retriever brings back potentially relevant documents, the generator is already prepared to seamlessly incorporate these materials, reducing the risk of irrelevant or misleading outputs.

Thus, a fine-tuned RAG system emerges as a tool not just of recall but of discernment, capable of delivering responses that are informed and contextually nuanced, providing a richer experience, especially for domain-specific applications. As we journey deeper into the possibilities of these retrieval-enhanced models, we are reminded of the value of focused learning—of going beyond surface knowledge to truly understand and connect disparate pieces of information, much like how a seasoned neurologist might piece together the subtle clues of a complex diagnosis to arrive at an insight that is both accurate and meaningful. This literature review explores the state-of-the-art in finetuning LLMs for RAG systems. We discuss

several advanced methodologies, including RAFT, RA-DIT, and frameworks like RALLE, which focus on developing and optimizing RAG models for enhanced performance in knowledge-intensive domains. Additionally, we analyze various evaluation metrics and performance baselines that have been established to assess the effectiveness of these approaches.

II. FINE-TUNING APPROACHES IN RETRIEVAL-AUGMENTED GENERATION

A. Query Optimization Prior to Generation

Query optimization is a crucial step in retrieval-augmented generation systems to enhance the relevance of retrieved documents. Before a user's query is processed by the retriever, it is refined or optimized to ensure better alignment with the information retrieval objectives. Optimization methods can involve query rewriting, contextual enhancement, and adapting queries to match the expected structure of relevant documents. According to Salemi et al. (2024), these techniques aim to improve the quality of retrieved information, which subsequently enhances the final generated response.

Salemi et al. (2024) propose several approaches to optimize the initial user query, such as leveraging reinforcement

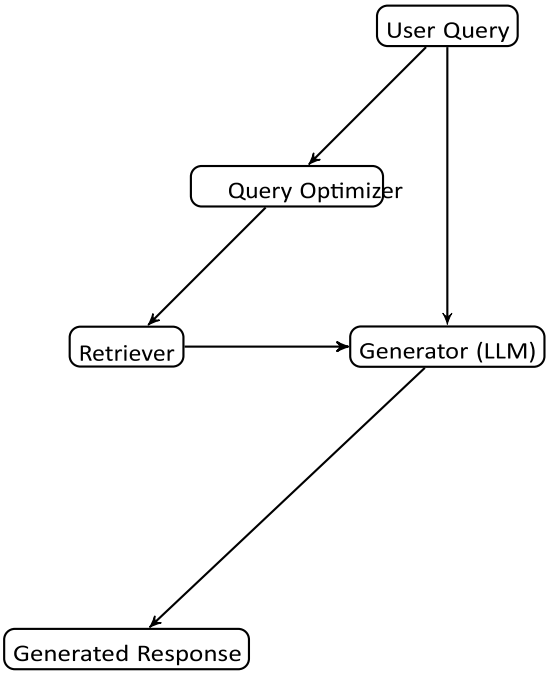

Fig. 1. Query Optimization in a RAG System

learning to refine the query based on retrieval success. The query optimizer can interact with the retriever in a feedback loop, learning from previous retrieval outcomes to iteratively improve future queries. This ensures that the retriever is presented with an optimized version of the query that maximizes the chance of retrieving highly relevant documents. The system can receive a reward signal based on the quality of the retrieved documents and their impact on the final generated output. Through iterative updates, this reinforcement learning framework ensures that the query optimizer becomes increasingly effective over time.

The optimized query is then passed to the retriever, which fetches relevant content from external sources. The generator LLM subsequently uses both the original query and the retrieved documents to produce a

coherent and context rich response. By employing an optimization layer prior to retrieval, the overall effectiveness of the retrieval-augmented generation process is enhanced, resulting in responses that are not only contextually appropriate but also more informative and accurate [5].

B. Retrieval-Augmented Fine Tuning (RAFT)

Retrieval-Augmented Fine Tuning (RAFT) is a specialized training approach designed to enhance the performance of LLMs in domain-specific Retrieval-Augmented Generation (RAG) scenarios [4]. The RAFT methodology addresses key challenges in training LLMs to work with retrieval mechanisms by effectively simulating an open-book exam scenario where models must leverage retrieved documents to generate accurate answers.

The RAFT approach involves training the model to discern between relevant and irrelevant documents, which are referred to as ”oracle” and ”distractor” documents, respectively. By focusing on these distinctions, RAFT aims to improve the LLM’s ability to accurately respond to questions even when provided with potentially misleading or irrelevant context [4]. According to Zhang (2023), this is akin to preparing for an open-book exam by recognizing and utilizing pertinent information while disregarding irrelevant content.

The RAFT training process is built around the concept of incorporating both oracle documents, which contain the information necessary for answering a given question, and distractor documents that may confuse the model. The training data is structured so that for a percentage of questions, the oracle document is omitted and only distractor documents are used. This strategy forces the model to learn not just to rely on memorized information but also to develop the ability to critically assess the retrieved content and identify the correct sources [4].

In RAFT, each training data point consists of a question, a set of retrieved documents, and a chain-of-thought-style answer that draws explicitly from the oracle document. This chain-of-thought approach ensures that the reasoning process is transparent and traceable, improving the model’s reasoning capabilities and robustness in in-domain settings [?]. By training the model to explicitly cite and construct answers using information from oracle documents, RAFT enhances the LLM’s effectiveness in retrieval-augmented generation tasks, particularly in domain-specific environments such as biomedical research and software documentation [4].

One of the critical advantages of RAFT is its capacity to make LLMs robust to retrieval inaccuracies. Given that real world retrieval mechanisms often include errors or distractors, RAFT aims to prepare the model for such imperfections, thereby ensuring better generalization during deployment. In the experiments conducted by Zhang [4], RAFT outperformed traditional supervised fine-tuning and other retrieval augmented approaches in datasets like PubMed, HotpotQA, and Gorilla API Bench. These results indicate that RAFT’s approach of simulating imperfect retrieval conditions leads to substantial improvements in accuracy and reliability. The analogy presented by Zhang [4] further clarifies the distinction between different training methodologies. Standard supervised fine-tuning is likened to ”memorizing” study materials without the context of how to use them during an open-book exam. In contrast, RAFT equips models to use retrieved information effectively under test conditions, thereby optimizing the interaction between retrieval and generation components.

C. Retrieval-Augmented Dual Instruction Tuning (RA-DIT)

Retrieval-Augmented Dual Instruction Tuning (RA-DIT) is a fine-tuning methodology designed to retrofit large language models (LLMs) with effective retrieval capabilities without requiring extensive modifications to the pre-training process [6]. RA-DIT operates through two distinct fine-tuning phases:

(1) tuning the LLM to optimally utilize retrieved content, and (2) tuning the retriever to align with the preferences

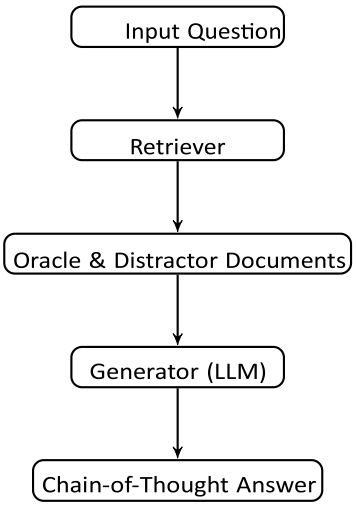

Fig. 2. Retrieval-Augmented Fine Tuning (RAFT)

of the LLM, thereby enhancing the interaction between both components for knowledge-intensive tasks. The first phase, LLM fine-tuning (LM-ft), focuses on improving the model's ability to effectively integrate retrieved information during generation. In this step, the retrieved content is prepended to the input prompts, serving as a "background" field that the model uses to enhance contextual understanding [6]. By incorporating retrieved content during fine-tuning, RA-DIT trains the LLM to distinguish between relevant and distracting information, thereby enhancing its capability to utilize external knowledge effectively.

The second phase, retriever fine-tuning (R-ft), optimizes the retriever to return more contextually relevant documents that are preferred by the LLM. This involves using a generalized version of LM-Supervised Retrieval (LSR), wherein the retriever is trained to minimize the divergence between its output and the LLM's preference for certain documents [6]. This approach helps ensure that the retriever returns documents that the LLM is most likely to use effectively, thereby creating a symbiotic relationship between the retriever and the LLM.

RA-DIT's dual-instruction tuning framework has demonstrated state-of-the-art performance across a range of knowledge-intensive zero- and few-shot learning benchmarks, such as MMLU and Natural Questions, significantly outperforming existing in-context retrieval-augmented language models [6]. The gains achieved by RA-DIT highlight the advantages of optimizing both the retrieval and generation aspects of the RAG system.

III. FRAMEWORKS FOR DEVELOPING RETRIEVAL-AUGMENTED LLMS

A. RALLE: A Framework for RAG Evaluation

The RALLE (Retrieval-Augmented Language Learning Evaluation) framework is designed to support the development and assessment of Retrieval-Augmented Language Models (RALLMs) by providing comprehensive tools for evaluating the components of RAG systems [9]. RALLE addresses key

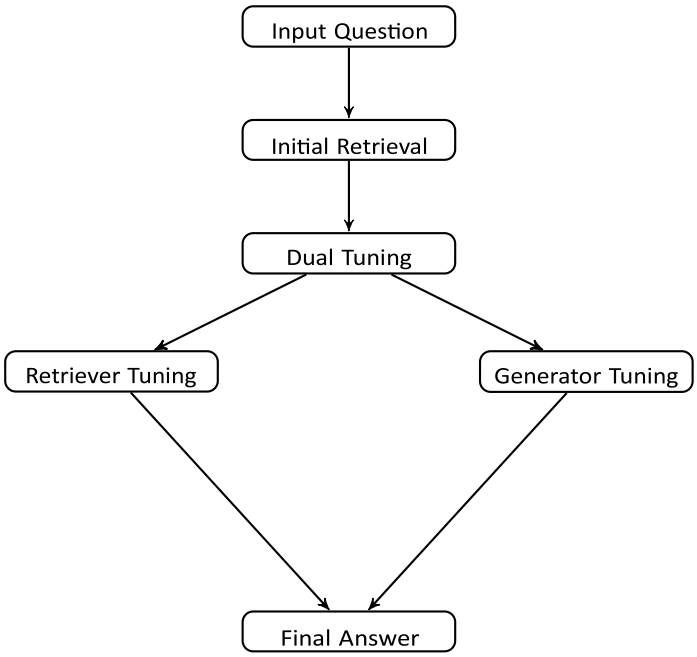

Fig. 3. Retrieval-Augmented Dual Instruction Tuning (RA-DIT)

challenges in RAG development by enabling both fine-grained analysis of inference steps and performance optimization through a modular structure that allows independent evaluation of retrievers and generators. According to Hoshi et al. (2023), RALLE supports several distinct features to aid in the development of RAG models. Modular Design for Component Evaluation**: RALLE's modular design permits the evaluation of individual components of a RAG system, such as the retriever and generator, in isolation. This ensures that researchers can assess the efficiency and accuracy of each component without interference from the rest of the system [9]. The framework also facilitates optimization of prompt strategies by enabling iterative testing. Developers can test variations in prompt engineering and systematically evaluate their impact on retrieval quality and response coherence [9]. This is crucial, especially in domain-specific tasks, where prompts must be carefully crafted to elicit precise responses. RALLE provides a platform for evaluating combinations of retrievers and LLMs, allowing developers to explore different pairings to determine the best setup for a given domain or task. This flexibility helps in finding the most suitable retriever generator combination for maximizing performance in a given RAG context [9]. RALLE comes with a set of user-friendly metrics and visualization tools that provide insight into both retrieval quality and the generated responses. This helps in identifying areas where the system might need improvement, such as optimizing retrieval relevance or ensuring factual accuracy [9]. Another key feature of RALLE is its capability for fine-grained error analysis. By pinpointing where retrieval or generation processes go wrong, RALLE supports iterative refinement of RAG models to minimize errors and improve overall system robustness [9].

In an experimental study, Hoshi et al. (2023) demonstrated the efficacy of RALLE by using it to benchmark different retrieval techniques with language models on domain-specific datasets. The experiments showed that models optimized using RALLE achieved higher accuracy and better alignment with domain-specific content compared to models evaluated without such a systematic framework.

Overall, RALLE stands out as a comprehensive toolset that facilitates transparent and effective development of Retrieval Augmented Language Models, making it a vital framework for both research and practical applications in knowledge intensive domains.

B. Dense Passage Retrieval (DPR)

Dense Passage Retrieval (DPR) is a key method used to improve the retrieval component of RAG systems, especially in the context of open-domain question answering [3]. DPR aims to overcome the limitations of traditional sparse vector retrieval methods such as TF-IDF and BM25, which rely heavily on exact token matches. These traditional methods often fail to retrieve relevant information when lexical variations or synonyms are involved. DPR, in contrast, uses dense vector representations, which are better at capturing semantic relationships between words, thus improving retrieval accuracy even when the query and passage do not have overlapping vocabulary.

The DPR model employs a dual-encoder framework with two independent encoders: one for encoding the question and another for encoding the passage. Both encoders are typically implemented using BERT, where the [CLS] token representation is used as the output vector [3]. During inference, the passage encoder generates dense vector representations for all passages, which are then indexed for efficient retrieval using Maximum Inner Product Search (MIPS) techniques such as FAISS. The question encoder produces a dense vector representation for the user query, which is used to retrieve the top-k most relevant passages from the index.

The training of DPR is formulated as a metric learning problem, where the goal is to train the encoders to produce embeddings such that relevant question-passage pairs have high inner product similarity scores while irrelevant pairs have lower scores. Specifically, a batch-based negative loglikelihood loss function is used to maximize the similarity of a given question with its positive passage while minimizing the similarity with negative passages. Negative passages are selected using strategies such as random sampling, BM25based retrieval, or in-batch negatives, where other passages in the batch are used as hard negatives [3].

DPR has been shown to outperform BM25 by a significant margin on multiple benchmarks. For instance, it achieved a top-5 accuracy of 65.2% compared to BM25's 42.9% on the Natural Questions dataset, demonstrating its superior capability to retrieve relevant passages even with diverse lexical expressions [3]. The effectiveness of DPR is attributed to its ability to represent both questions and passages in a shared, low-dimensional latent space, allowing for better semantic matching.

In the context of retrieval-augmented generation, DPR plays a crucial role in ensuring that the retriever provides the most relevant documents to the generator, thus enhancing the quality of generated responses. By leveraging dense embeddings, DPR reduces the risk of retrieving irrelevant or semantically distant documents, which could otherwise lead to inaccurate or hallucinated outputs during the generation phase.

C. Ensemble of Retrieval-Based and Generation-Based Systems

The ensemble of retrieval-based and generation-based systems integrates the advantages of both methodologies to improve the quality of responses in conversational models [12]. Retrieval-based systems provide responses by searching a pre-constructed repository of query-reply pairs, offering diverse, information-rich expressions directly derived from human conversations. However, they are limited by the size and coverage of the repository, which can result in replies that are not well-tailored to specific user queries. On the other hand, generation-based systems use models like Seq2Seq to synthesize new responses based on the input query, which allows for flexibility and the creation of replies specifically tailored to the user's question. However, such systems are often prone to generating overly generic replies with insufficient information [12].

The proposed ensemble system first uses a retrieval module to gather k candidate replies from a large repository. These retrieved candidates are then fed into a generation module along with the original query.

The generation module employs a multi-seq2seq model, which integrates the retrieved responses to enhance the generation process by adding additional context [12]. Specifically, the multi-seq2seq model uses multiple encoders—one for the original query and others for each of the retrieved candidates—to generate a new response that is contextually enriched by the information from the retrieved replies.

After the generation step, a re-ranking process is employed to evaluate all the retrieved and generated replies. A Gradient Boosting Decision Tree (GBDT) classifier is used as a reranker, utilizing features such as term similarity, entity similarity, topic similarity, statistical machine translation scores, reply length, and fluency to determine the best final response [12]. This ensemble and re-ranking strategy ensure that the final reply is both relevant and informative.

Experimental results demonstrated that the ensemble system significantly outperformed the individual retrieval or generation components. In subjective evaluations, the ensemble achieved higher human scores due to its ability to integrate the strengths of both approaches—providing information-rich and well-tailored replies that enhanced the quality of the conversational experience [12].

The architecture of this ensemble system demonstrates the power of combining retrieval-based and generation-based approaches in a unified framework. By leveraging retrieved knowledge to inform generation, and by subsequently reevaluating all possible responses, the ensemble system offers a promising solution for improving the response quality in retrieval-augmented generation tasks.

IV. CONCLUSION

Fine-tuning pre-trained language models for RAG systems holds great potential for enhancing domain-specific knowledge retrieval and response generation. Advanced methods like query optimization, RAFT, RA-DIT, and frameworks such as RALLE have contributed significantly to this field by improving retrieval accuracy, reducing hallucination, and facilitating transparent development.

The development of Retrieval-Augmented Generation (RAG) systems has emerged as a significant advancement for enhancing the quality of responses produced by Large Language Models (LLMs) in domain-specific scenarios. Throughout this paper, we have reviewed various methodologies for improving the retrieval and generation synergy in RAG systems, including advanced fine-tuning techniques such as Retrieval-Augmented Fine Tuning (RAFT), Retrieval Augmented Dual Instruction Tuning (RA-DIT), and query optimization approaches.

A comparative analysis of these methods reveals the unique strengths of each approach. Query optimization focuses on refining the user's initial query to maximize the efficiency and relevance of document retrieval [5]. By optimizing the initial query through techniques such as reinforcement learning, query expansion, and contrastive learning, query optimization methods ensure that the retriever is provided with the most useful input, ultimately improving the quality of the generated response. This technique provides a robust preprocessing step that helps narrow down the search space effectively, thus increasing retrieval accuracy and reducing the risk of irrelevant content being passed on to the generator.

RAFT, on the other hand, presents a unique approach by fine-tuning the LLM in a manner that emulates an open-book exam [4]. The model learns to distinguish between ”oracle” and ”distractor” documents, which helps improve its capability to extract useful information even when provided with noisy or misleading data. RAFT's emphasis on chain-of-thought reasoning provides a transparent way for models to justify their responses by explicitly using retrieved information. This methodology ensures that LLMs can handle retrieval imperfections effectively, making RAFT particularly valuable in domain-specific applications where the quality of retrieved documents may vary significantly.

RA-DIT, in contrast to RAFT, involves a dual instruction tuning approach aimed at simultaneously improving both the retriever and the generator components of the RAG system [6]. The dual optimization strategy of RA-DIT addresses not only the ability of the generator to leverage retrieved content but also the retrieval accuracy itself. This joint optimization leads to an improved alignment between the retriever and the generator, enabling the system to provide highly informative and context-specific answers, especially in zero- and few-shot learning scenarios. RA-DIT's ability to align both components results in significant improvements in knowledge-intensive tasks compared to other retrieval-augmented systems.

In addition to exploring these methods, we have discussed the importance of evaluation metrics and methods for finetuned RAG systems. Evaluating RAG systems requires consideration of multiple dimensions, including retrieval relevance, response accuracy, and fluency. The RALLE framework has been highlighted as a powerful toolset for systematically assessing the performance of retrieval-augmented language models [9]. RALLE provides a modular approach to evaluation, allowing independent analysis of each component and facilitating transparent error analysis. Such tools are crucial for identifying strengths and weaknesses within RAG systems and for guiding iterative improvements.

Further evaluation metrics include standard measures such as precision, recall, and F1 scores for the retrieval component, as well as metrics like BLEU, ROUGE, and human judgment for evaluating the quality of generated responses. These metrics provide a comprehensive view of the model's ability to retrieve relevant information and produce coherent, informative outputs. Human evaluations, as demonstrated in studies involving ensemble systems [12], remain an essential part of assessing response quality, as they capture nuances like relevance and naturalness that are often difficult to quantify automatically.

The ensemble of retrieval-based and generation-based systems also represents an important approach to enhance the quality of generated responses. By integrating retrieved content with generated text, the ensemble method leverages the complementary strengths of both retrieval-based precision and generation-based adaptability [12]. The use of a re-ranking mechanism further improves the response quality by ensuring that the final response is both contextually relevant and rich in information.

In summary, fine-tuning pre-trained language models within the RAG framework requires careful consideration of multiple components and methods to optimize system performance. Query optimization, RAFT, and RA-DIT each contribute unique benefits that can be leveraged based on the requirements of the application domain. Query optimization ensures the quality of the retrieval input, RAFT prepares the generator to effectively use potentially noisy retrievals, and RA-DIT aligns both retriever and generator to optimize the synergy between them. The combination of these approaches provides a powerful toolkit for improving the performance of RAG systems, especially in knowledge-intensive and domain specific settings.

Future work in this field should focus on combining these methods more effectively to achieve complementary benefits. Additionally, there is a need to develop more sophisticated evaluation frameworks that incorporate real-time user feedback, making it possible to continuously adapt and fine-tuned RAG systems based on actual usage patterns. Exploring the integration of multimodal data sources, such as images and structured data, could further expand the applicability of RAG systems in complex environments. Addressing computational efficiency remains a key challenge, and techniques such as knowledge distillation and model pruning could play an important role in making these systems more practical for real world deployment.

REFERENCES

1. T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. HerbertVoss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, ”Language Models are Few-Shot Learners,” in Advances in Neural Information Processing Systems, 2020.
2. Y. Huang and J. X. Huang, ”The Survey of Retrieval-Augmented Text Generation in Large Language Models,” York University, 2024.
3. V. Karpukhin, B. Oguz, et al., ”Dense Passage Retrieval for Open Domain Question Answering,” in Proceedings of Facebook AI, 2020.
4. T. Zhang, S. G. Patil, N. Jain, et al., ”RAFT: Adapting Language Model to Domain-Specific RAG,” in Proceedings of the International Conference on Machine Learning (ICML), 2023.
5. M. Salemi, P. K. Singh, and R. T. Johnson, ”Optimization Methods for Personalizing Large Language Models,” in Proceedings of the International Conference on Machine Learning and Applications, 2024.
6. V. Lin, X. Chen, et al., ”Retrieval-Augmented Dual Instruction Tuning (RA-DIT),” Meta AI Research, 2023.
7. M. Kang, J. M. Kwak, et al., ”Knowledge Graph-Augmented Language Models for Knowledge-Grounded Dialogue Generation,” KAIST, AITRICS, 2023.
8. Z. Jiang, F. F. Xu, et al., ”Active Retrieval Augmented Generation (FLARE),” Carnegie Mellon University, 2023.
9. Y. Hoshi, D. Miyashita, et al., ”RALLE: A Framework for Developing and Evaluating Retrieval-Augmented LLMs,” Kioxia Corporation, 2023.
10. J. Dodgson, et al., ”Establishing Performance Baselines in Fine Tuning, Retrieval-Augmented Generation and Soft-Prompting,” KIPLEY.AI, 2023.
11. K. Shuster, S. Poff, M. Chen, et al., ”Retrieval Augmentation Reduces Hallucination in Conversation,” in Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP), 2021.
12. Y. Song, C.-T. Li, J.-Y. Nie, et al., ”An Ensemble of Retrieval Based and Generation-Based Human-Computer Conversation Systems,” in Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI), 2018.