Federated Learning and RAG Integration: A Scalable Approach for Medical Large Language Models

Jincheol Jung, Hongju Jeong, and Eui-Nam Huh
College of Software Convergence, Kyung Hee University
Yongin-si, 17104, Republic of Korea
Email: {bik1111, sub06038, johnhuh}@khu.ac.kr

***Abstract*—This study analyzes the performance of domain-specific Large Language Models (LLMs) for the medical field by integrating Retrieval-Augmented Generation (RAG) systems within a federated learning framework. Leveraging the inherent advantages of federated learning, such as preserving data privacy and enabling distributed computation, this research explores the integration of RAG systems with models trained under varying client configurations to optimize performance. Experimental results demonstrate that the federated learning-based models integrated with RAG systems consistently outperform their non-integrated counterparts across all evaluation metrics. This study highlights the potential of combining federated learning and RAG systems for developing domain-specific LLMs in the medical field, providing a scalable and privacy-preserving solution for enhancing text generation capabilities.**

***Index Terms*—Federated learning, Large language models, RAG, Fine-tuning**

I. Introduction

The advancement of large language models (LLMs) [1] has significantly expanded the scope of natural language processing (NLP) tasks such as text comprehension, reasoning, and generation. These technologies are particularly impactful in domain-specific applications like medical, where generating contextually accurate and relevant information is critical. However, the centralized paradigm of LLM training and deployment, which consolidates data onto a single site, faces significant challenges in sensitive domains. The inherent sensitivity of medical data and stringent regulatory requirements amplify concerns regarding data privacy, security, and scalability, limiting the applicability of LLMs in medical.

Federated Learning (FL) offers a promising alternative by enabling collaborative model training across decentralized data sources while ensuring that data remains on local devices. FL provides a robust framework for safeguarding data privacy and achieving scalability, making it particularly effective for sensitive data environments.

Meanwhile, Retrieval-Augmented Generation (RAG) [2] systems enhance both information retrieval and text generation performance. RAG systems retrieve relevant information from external knowledge bases and utilize it to generate contextually enriched and accurate responses. This capability is particularly valuable in domain-specific applications such as medical, where integrating up-to-date knowledge and context is critical. However, most existing RAG systems are designed for centralized architectures, and their application in decentralized FL environments remains underexplored.

This study compares four approaches to integrating LLMs with RAG systems: centralized LLM [3], centralized LLM with RAG [4], federated LLM [5], and federated LLM with RAG. The study proposes a federated LLM framework that leverages client-specific RAG systems to enable decentralized retrieval and generation optimized for local datasets. This integration adheres to the privacy-preserving principles of FL while ensuring effective performance in heterogeneous client environments.

The experiments were conducted using the Medical Meadow Flashcards dataset and the federated learning framework Flower. Client-specific RAG systems were integrated using subsets of the PubMed Central® (PMC) Open Access Subset [6]. While the study utilized open datasets due to the constraints of accessing real-world medical data, the framework can be extended to real-world applications using institutional datasets. This approach protects sensitive medical data while enabling the generation of contextually appropriate information tailored to the characteristics of individual client datasets.

The performance evaluation was based on metrics such as Context Recall, Factual Correctness, Faithfulness, Semantic Similarity, and Answer Relevancy [7]. The results show that federated LLMs integrated with RAG systems achieved performance comparable to or exceeding centralized architectures and outperformed models without RAG integration across all metrics.

The primary objective of this study is to analyze the impact of RAG system integration on different learning paradigms, with a particular focus on maximizing the synergy between LLMs and RAG systems in federated learning environments. By comparing centralized and federated approaches, this research aims to empirically demonstrate whether the integration of RAG systems into federated learning frameworks can enhance performance and scalability while ensuring data privacy.

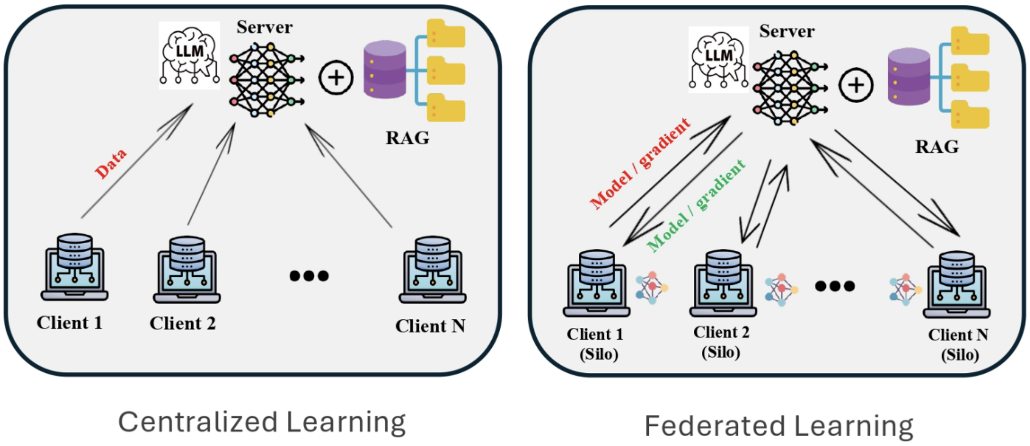

Figure 1: A comparison of centralized learning, federated learning in RAG system integration. The arrows indicate the data flow through the model training process.

II. Background

A. Federated Learning

Federated Learning is a distributed machine learning paradigm where a global model is collaboratively trained across multiple devices or clients without sharing their local data.

This privacy-preserving approach relies on aggregating locally computed updates to refine the global model. FL enables access to previously inaccessible private datasets, thereby promoting scalability and model generalization.

We propose employing FL to fine-tune LLMs tailored to specific domains. The federated training pipeline consists of three stages: (1) a global model is distributed to selected clients, (2) local training is performed on each client with private data, and (3) locally updated models are aggregated to produce an improved global model. This process is repeated over multiple communication rounds to achieve convergence.

B. Flower Framework

Flower is a server-client-based federated learning (FL) framework that facilitates decentralized training by enabling clients to perform local model updates [8], which are then aggregated by the server to refine a global model. The communication between clients and the server leverages well-established algorithms such as FedAvg [9], with the flexibility to implement custom aggregation strategies tailored to specific research needs. In this study, Flower was employed to orchestrate the training of a global medical domain model across distributed client environments. Its robust support for heterogeneous clients and versatile deployment scenarios made Flower an ideal choice for addressing the diverse requirements of this research, particularly in maintaining scalability and adaptability across varying system configurations.

III. RAG System Integration: Methods And Workflow

This section describes the learning methodologies of centralized and federated learning environments and outlines the processes and methodologies for integrating RAG systems into the trained models. Figure 1 visually compares the architectures of centralized learning, federated learning, and their respective strategies for integrating RAG systems.

In the centralized learning paradigm, all client data are aggregated onto a single site to train domain-specific LLMs. While centralized learning offers the advantage of simplified data integration and model management, it c relies on centralized data storage and processing, leading to significant concerns regarding data privacy and security. These issues are particularly critical in sensitive domains such as medical, where stringent regulatory requirements and the sensitive nature of the data exacerbate these challenges.

In contrast, federated learning distributes data across clients, enabling model training to run locally within each client's environment. Clients independently update their local models using private datasets and periodically communicate these updates to a central server, which aggregates them to produce a global model. This iterative process continues across multiple communication rounds until convergence is achieved. Federated learning preserves data privacy by keeping data localized while enhancing scalability in distributed environments.

This study systematically compares four approaches by integrating RAG systems into LLMs trained under each learning paradigm: (1) centralized LLM, (2) centralized LLM with integrated RAG systems, (3) federated LLM, and (4) federated LLM with integrated RAG systems.

The integration process of RAG systems into centralized and federated learning environments involves the following stages:

1) Document Processing: To provide context for the RAG system, 85 PDF files related to the fields of medicine and life sciences were utilized. These files were sourced from PMC, a free full-text archive of biomedical and life sciences journal literature maintained by the U.S. National Institutes of Health (NIH) and the National Library of Medicine (NLM) [10]. The PDF files were processed using LangChain's `PyPDFLoader` for content extraction. The extracted content was then segmented into 1000-character chunks with an overlap of 50 characters using the `RecursiveCharacterTextSplitter` utility to ensure continuity across the divided text.

2) Search Mechanism: Two retrieval methods were utilized to identify relevant documents:

- **BM25:** A traditional text-based retrieval method that ranks documents based on term frequency and inverse document frequency [11].
- **FAISS:** A dense embedding-based retrieval method that leverages `neuml/pubmedbert-base-embeddings` model to retrieve semantically similar documents [12].
- **Ensemble Retrieval:** To combine the strengths of BM25 and FAISS, an ensemble retriever was configured, assigning 80% weight to BM25 and 20% to FAISS.

3) LLM Integration: The fine-tuned LLM was integrated into the RAG pipeline using HuggingFace's text generation pipeline to manage response generation. Responses were generated with a maximum length of 512 tokens, and the temperature was set to 0, ensuring deterministic outputs based on retrieved contexts for reliable responses.

IV. Experimetal Setup

This section provides a systematic explanation of the experimental design for both centralized and federated learning approaches, aimed at constructing a domain-specific model for the medical domain. Additionally, it details the method for integrating the RAG system into each learning paradigm to evaluate performance.

All experiments were conducted in an NVIDIA GeForce RTX 3090 GPU environment, with both paradigms utilizing the Mistral 7B model as the base model. To enhance model efficiency, 4-bit quantization was applied, and the Low-Rank Adaptation (LoRA) [13] technique was employed to enable efficient fine-tuning. LoRA was configured with an r-value of 16 and an alpha of 64.

In the federated learning approach, the Flower framework was used for fine-tuning the model. A total of 20 virtual clients were generated, and for each training round, a predefined number of clients (2, 4, or 6) were randomly selected to participate in the training process. The dataset distribution was set to Non-Independent and Identically Distributed(Non-IID), with approximately 3.4k Medical Meadow Flashcards [14] unevenly allocated among clients as follows: `[900, 926, 1052, 1064, 1136, 1250, 1319, 1328, 1448, 1524, 1659, 1675, 1877, 1924, 2089, 2144, 2350, 2515, 2627, 3148]`.

The learning rate was dynamically adjusted using a cosine annealing function, with hyperparameters set to $lrate\_max = 5 \times 10^{-5}$ and $lrate\_min = 1 \times 10^{-4}$. The batch size was fixed at 16, and cross-entropy loss was employed as the loss function to minimize the discrepancy between the model's output and the actual labels. Training was conducted for a total of 100 rounds.

In the centralized learning approach, the concept of rounds used in federated learning was not applicable. Instead, the model was trained on the entire dataset using a single server. The batch size was set to 16, and training was conducted over 3 epochs, resulting in a total of 6369 steps. The learning rate was configured at 5×10^{-5}, and a cosine annealing scheduler was applied for learning rate adjustment.

For both learning paradigms, the fine-tuned models were integrated with RAG systems. To evaluate the performance of the RAG system, the toolkit ragas [15] was employed, which is specifically designed for evaluating LLMs applications. Key performance metrics included Context Recall, Factual Correctness, Faithfulness, Semantic Similarity, and Answer Relevancy.

V. Experimental Results

This section presents the results of integrating RAG systems into fine-tuned models under each learning paradigm and provides a comparative analysis of their performance. In the federated learning paradigm, models trained with 2, 4, and 6 clients were evaluated with RAG system integration. The evaluation metrics employed were Context Recall, Factual Correctness, Faithfulness, Semantic Similarity, and Answer Relevancy, each of which is defined as follows:

- **Context Recall**: Measures how successfully relevant documents were retrieved from the provided context. This metric evaluates whether critical information was missed, with higher values indicating that more relevant context was included. Context Recall is always computed with reference to the ground truth data.
- **Factual Correctness**: Evaluates the factual accuracy of the generated response by comparing it with the ground truth data. This metric quantifies the alignment between the response and the ground truth using Natural Language Inference (NLI) to decompose both into claims and assess factual overlap. Scores range from 0 to 1, where higher scores indicate better factual correctness.
- **Faithfulness**: Assesses how consistent and factual the generated response is with respect to the retrieved context. Responses receive high scores if all claims can be inferred from the given context. Scores range from 0 to 1, with higher values reflecting greater reliability of the response.
- **Semantic Similarity**: Measures the semantic alignment between the generated response and the ground truth. This metric evaluates the degree of semantic consistency using a cross-encoder model to calculate scores. Scores range from 0 to 1, with higher values indicating superior semantic coherence.
- **Answer Relevancy**: Evaluates how relevant the generated response is to the given question. This metric involves generating a reverse query from the answer and assessing its cosine similarity with the original question. Higher

TABLE I: Comparison of Settings with and without RAG For models without RAG, Context Recall and Faithfulness are blank as no context is retrieved.

Experiment Scenario	Setting	Context Recall	Factual Correctness	Faithfulness	Semantic Similarity	Answer Relevancy
2	w/o RAG	-	0.1160	-	0.8205	0.9357
	w/ RAG	**0.5**	**0.158**	**0.4364**	**0.8631**	**0.9374**
4	w/o RAG	-	0.1260	-	0.8348	0.9366
	w/ RAG	**0.5**	**0.2000**	**0.4077**	**0.8736**	**0.9449**
6	w/o RAG	-	0.1020	-	0.8177	0.9200
	w/ RAG	**0.5**	**0.243**	**0.5160**	**0.8760**	**0.9370**
Centralized Learning	w/o RAG	-	0.096	-	0.8206	0.7449
	w/ RAG	**0.5**	**0.137**	**0.3368**	**0.8629**	**0.9508**

scores reflect stronger alignment between the question and the response.

The performance analysis of RAG system integration demonstrates consistently superior results across all evaluation metrics in both centralized and federated learning paradigms (with 2, 4, and 6 clients). Notably, in the federated learning paradigm, the Semantic Similarity metric exhibited the most significant performance improvement across all experimental scenarios.

When comparing learning paradigms, federated learning without RAG integration outperformed centralized learning in Factual Correctness and Answer Relevancy metrics across all client configurations (2, 4, and 6 clients). With RAG integration, federated learning continued to outperform centralized learning, particularly in Factual Correctness and Semantic Similarity metrics, underscoring its effectiveness in distributed environments.

Within the federated learning paradigm, the configuration with 6 clients and RAG integration achieved the highest performance in Factual Correctness, Faithfulness, and Semantic Similarity metrics compared to configurations with 2 or 4 clients. Moreover, when compared to its counterpart without RAG integration, the 6-client configuration recorded the largest performance gains. This trend was especially pronounced in Factual Correctness and Semantic Similarity metrics, which showed a positive correlation with the number of participating clients, demonstrating improved performance as the client count increased.

VI. Conclusions

This study presents an empirical analysis of the potential for integrating federated learning with RAG systems to develop domain-specific LLMs in the medical domain. The proposed framework demonstrates its capability to deliver robust and scalable performance while preserving data privacy, a critical requirement in distributed and heterogeneous client environments.

Experimental results reveal that models integrating FL with RAG systems consistently outperform centralized learning approaches across all evaluation metrics, with notable improvements in Semantic Similarity and Factual Correctness. Additionally, the study highlights a positive correlation between the number of participating clients and model performance, further validating the scalability and effectiveness of FL in distributed settings.

This work underscores the viability of integrating FL and RAG systems as a practical solution for privacy-preserving and high-performance text generation in sensitive domains such as medical. The demonstrated framework not only addresses key challenges in applying LLMs to privacy-sensitive environments but also establishes a foundation for extending this approach to other domain-specific applications requiring robust data privacy and scalability.

Acknowledgment

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2024-RS-2023-00258649) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation) Professor Eui-Nam Huh is the corresponding author.

References

[1] S. Minaee, T. Mikolov, N. Nikzad, M. Chenaghlu, R. Socher, X. Amatriain, and J. Gao, ”Large language models: A survey,” arXiv preprint arXiv:2402.06196, 2024.

[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] Hu, J., Wang, D., Wang, Z., Pang, X., Xu, H., Ren, J., & Ren, K. (2024). Federated Large Language Model: Solutions, Challenges and Future Directions. IEEE Wireless Communications.

[4] Rangan, K., & Yin, Y. (2024). A fine-tuning enhanced RAG system with quantized influence measure as AI judge. Scientific Reports, 14(1), 27446.

[5] Kuang, W., Qian, B., Li, Z., Chen, D., Gao, D., Pan, X., ... & Zhou, J. (2024, August). Federatedscope-llm: A comprehensive package for fine-tuning large language models in federated learning. In Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 5260-5271).

[6] National Center for Biotechnology Information, ”PubMed Central,” Available: https://pmc.ncbi.nlm.nih.gov/, [Accessed: Dec. 10, 2024].

[7] Ragas Documentation. ”Available Metrics.” [Online]. Available: https://docs.ragas.io/en/stable/concepts/metrics/available_metrics/, [Accessed: Dec. 10, 2024].

[8] Beutel, D. J., Topal, T., Mathur, A., Qiu, X., Fernandez-Marques, J., Gao, Y., ... & Lane, N. D. (2020). Flower: A friendly federated learning research framework. arXiv preprint arXiv:2007.14390.

[9] Li, X., Huang, K., Yang, W., Wang, S., & Zhang, Z. (2019). On the convergence of fedavg on non-iid data. arXiv preprint arXiv:1907.02189.

[10] National Center for Biotechnology Information. ”PubMed Central Open Access PDF Archive.” [Online]. Available: https://ftp.ncbi.nlm.nih.gov/pub/pmc/oa_pdf/. [Accessed: Dec. 10, 2024].

[11] Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. Foundations and Trends® in Information Retrieval, 3(4), 333-389.

[12] Douze, M., Guzhva, A., Deng, C., Johnson, J., Szilvasy, G., Mazaré, P. E., ... & Jégou, H. (2024). The faiss library. arXiv preprint arXiv:2401.08281.

[13] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2021). Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685.

[14] Han, T., Adams, L. C., Papaioannou, J. M., Grundmann, P., Oberhauser, T., Löser, A., ... & Bressem, K. K. (2023). MedAlpaca–an open-source collection of medical conversational AI models and training data. arXiv preprint arXiv:2304.08247.

[15] Es, S., James, J., Espinosa-Anke, L., & Schockaert, S. (2023). Ragas: Automated evaluation of retrieval augmented generation. arXiv preprint arXiv:2309.15217.