COSE474-2024F: Final Project Report
Enhancing Domain-Specific Question Answering Accuracy via Prompt Optimization

2021170220 / Kang Seok Jeon

1. Introduction

1.1. Motivation

In recent years, the demand for conversational AI has grown significantly across various industries, driven by its ability to enhance efficiency and accessibility. The global conversational AI market reached approximately $1.386 billion in 2024, with a projected compound annual growth rate (CAGR) of 21.95%, reflecting widespread adoption in sectors such as healthcare, finance, and information technology (Markets & Markets, 2024)(Insights, 2024). These systems are increasingly being used to provide automated customer service, streamline operations, and improve user experiences in professional and everyday applications.

Despite this rapid adoption, general-purpose AI models like ChatGPT often fail to address domain-specific queries with sufficient depth and accuracy. Their reliance on generalized training data results in limitations when handling specialized topics such as medical advice, legal consultations, or technical troubleshooting. This highlights the need to optimize AI models for domain-specific applications to meet the growing demand for precise and contextually accurate responses in professional settings.

This research focuses on enhancing the accuracy of domain-specific question answering by incorporating targeted knowledge retrieval and optimizing prompt design. The goal is to enable AI systems to deliver more reliable and contextually relevant answers in specialized domains.

1.2. Problem Definition

General-purpose large language models (LLMs) like ChatGPT struggle with domain-specific question answering due to their dependence on broad, generalized training data. Consequently, their responses often lack the precision and depth required for specialized fields such as healthcare, law, and technology.

Retrieval-Augmented Generation (RAG) addresses these limitations by integrating external knowledge retrieval with generative models(Lewis et al., 2020). RAG retrieves relevant documents from a knowledge base and uses them as context to generate responses, enabling systems to dynamically incorporate domain-specific information. Furthermore, various extensions of RAG, such as Blended RAG(Sawarkar et al., 2024) and RAFT (Retrieval-Augmented Fine-Tuning)(Zhang et al., 2024), have been proposed to further improve its performance.

However, RAG's performance can be highly dependent on prompt design for generator. Poorly crafted prompts can lead to irrelevant document retrieval or misaligned responses, limiting the system's effectiveness. This project seeks to improve RAG's performance by focusing on prompt optimization, refining the interaction between retrieval and generation processes. Through improved prompt design, the aim is to maximize RAG's domain-specific accuracy while maintaining architectural simplicity.

1.3. Concise Description of Contribution

This project aims to enhance domain-specific question answering accuracy by leveraging Retrieval-Augmented Generation (RAG) and optimizing prompt design. The research will evaluate the impact of prompt refinement on the performance of RAG systems, focusing on improving retrieval relevance and response coherence in specialized fields. By comparing baseline RAG systems with those incorporating optimized prompts, the study will assess the potential for achieving precise and contextually relevant answers in professional applications.

2. Methods

2.1. Overview

This project combines Retrieval-Augmented Generation (RAG) with learnable vector-based prompts to improve domain-specific question answering. The learnable prompts act as a foundation of domain knowledge, enhancing the system's ability to understand queries and generate contextually relevant responses. As shown in Figure 1, the method dynamically integrates retrieval results with optimized prompts, ensuring adaptability and scalability across domains.

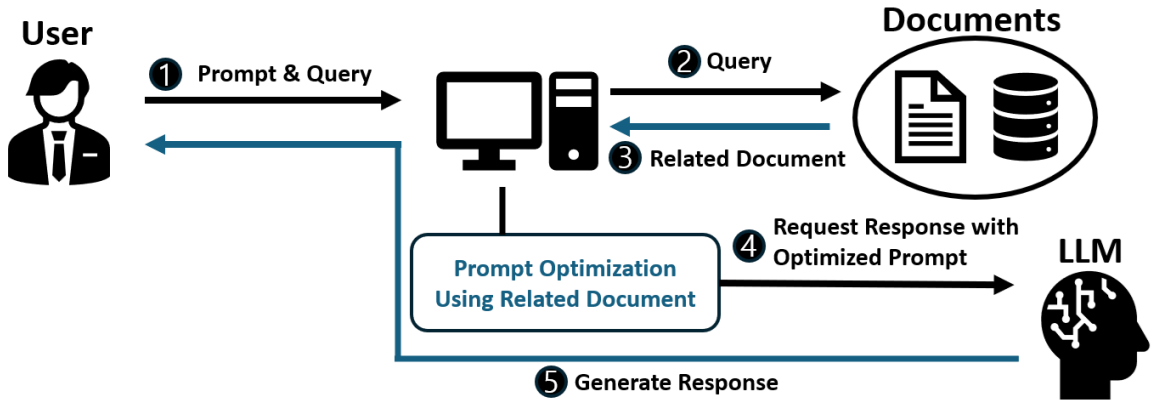

Figure 1. Workflow of Prompt Optimization into RAG Pipeline

2.2. Methodology

1. **Prompt and Query Generation:** The user provides a query and an initial prompt. The prompt, enhanced by the learnable embeddings, serves as the input to the retrieval-augmented pipeline.

2. **Document Retrieval:** The retriever fetches relevant documents from a knowledge base using Maximum Inner Product Search (MIPS), creating a dynamic context for the query.

3. **Prompt Optimization:** Retrieved documents refine the initial prompt via learnable vector-based embeddings. These embeddings encapsulate domain-specific knowledge and dynamically adapt to the context of the query.

4. **Response Generation:** The optimized prompt, combined with the query and retrieved documents, is passed to LLAMA 3.1 Instruct 8B to generate the final response.

5. **Feedback and Update:** Generated responses are evaluated against ground truth. The feedback is used to update the learnable prompt embeddings, improving both retrieval relevance and response accuracy.

All code and pipelines will be open-sourced for reproducibility.

2.3. Significance and Novelty

The method introduces a novel combination of RAG and learnable prompts: First, learnable prompts provide a foundation of domain-specific knowledge, improving contextual understanding even with limited retrieval accuracy. Second, prompts adapt dynamically to the query and retrieved documents, ensuring relevance and consistency. Lastly, eliminates static prompt engineering by allowing end-to-end optimization of prompt embeddings.

2.4. Algorithm Description

1. Input: User query q, initial prompt p_0, and knowledge base D.
2. Retrieve top-k documents $\{d_1, d_2, \ldots, d_k\}$ using query embeddings E_q.
3. Optimize prompt embeddings E_p with the retrieved documents.
4. Combine E_p and E_q, then generate response r with the LLM.
5. Output: Response r.

Training Process

1. **Initialization:** Prompt embeddings $\mathbf{P}$ are initialized randomly with dimensions $k \times d$, where k is the length and d is the embedding size.
2. **Loss Function:** Cross-entropy loss minimizes the gap between the generated response $\hat{Y}$ and the ground truth Y:
3. **Gradient Update:** Using backpropagation, embeddings $\mathbf{P}$ and model weights are updated iteratively:

$$\mathbf{P} \leftarrow \mathbf{P} - \eta \nabla_{\mathbf{P}} \mathcal{L}.$$

The learnable prompt embeddings act as a foundation of domain knowledge, offering several key benefits.

Testing Process

1. **Learned Prompt Usage:** The trained embeddings $\mathbf{P}$ are loaded and combined with new query embeddings $\mathbf{Q}$.

2. **Dynamic Context Creation:** Retrieved documents are integrated with $\mathbf{P}$ and $\mathbf{Q}$ to generate a dynamic prompt.

3. **Answer Generation:** LLAMA 3.1 generates responses using the dynamic prompt.

3. Experiments

3.1. Dataset

The experiments were conducted using the **MedQA** dataset, which contains medical questions labeled with three possible answers: *Yes*, *No*, and *Maybe*. The dataset includes a training set for model learning and a separate test set for evaluation. For this study, we split the training data into 90% for training and 10% for internal validation.

3.2. Computing Resources and Experimental Design

The experiments were carried out on a single NVIDIA A100 GPU with 40GB of memory, running on a Linux-based system with PyTorch 2.0 and CUDA 12.2. The LLAMA 3.1 Instruct 8B model was used as the generative backbone. Each experimental configuration was evaluated three times to ensure consistency.

We designed the experiments to compare the following methods:

1. **Basic LLAMA**: Directly generating answers without additional retrieval or optimization techniques.

2. **LLAMA with RAG**: Integrating Retrieval-Augmented Generation (RAG) by retrieving the top 3 most similar contexts using cosine similarity.

3. **LLAMA with Optimized RAG**: Extending RAG with a learnable prompt that optimizes the interaction between the retrieved contexts and the generative model.

3.3. Quantitative Results

The accuracy of each method, averaged across three independent runs, is summarized in Table 1. The results indicate that while the basic LLAMA model achieves the highest accuracy, the performance degrades with the introduction of RAG and optimized RAG.

Table 1. Accuracy (%) for MedQA across methods

Method	Run 1	Run 2	Run 3
Basic LLAMA	68.0	69.0	68.0
LLAMA with RAG	64.0	64.0	59.0
LLAMA with Optimized RAG	55.0	55.0	56.0

3.4. Qualitative Results

The qualitative analysis revealed that the **Basic LLAMA** model provided consistent and straightforward answers, albeit occasionally lacking depth in context-sensitive questions. In contrast, the **LLAMA with RAG** method introduced relevant contexts but sometimes retrieved noisy or unrelated information, leading to inconsistencies in generated answers. The **LLAMA with Optimized RAG** approach, while aiming to refine context interaction through a learnable prompt, underperformed due to overfitting and difficulty in generalization. These findings highlight the challenges of effectively integrating retrieval-based methods with generative models in domain-specific tasks like MedQA, where excessive or irrelevant context can detract from model performance.

3.5. Discussion

The basic LLAMA model outperformed the other methods in terms of accuracy, suggesting that retrieval and prompt optimization may not be as effective for this dataset. The drop in accuracy for RAG can be attributed to the retrieved contexts introducing noise or irrelevant information. Optimized RAG further reduced performance, likely due to overfitting during the prompt training phase.

The experiment also highlights the challenges of integrating retrieval-based methods with generative models in highly domain-specific tasks like MedQA. A more robust retrieval mechanism or dataset-specific fine-tuning may be required to unlock the potential of RAG and prompt optimization.

4. Future Directions

- **Improved Retrieval**: Employ advanced retrieval methods, such as dense retrieval with FAISS or DPR, to improve the relevance of retrieved contexts.
- **Dataset Augmentation**: Expand the dataset with additional high-quality examples to enhance generalization.
- **Hybrid Architectures**: Explore hybrid approaches that combine retrieval-based methods with end-to-end fine-tuning of the generative model.
- **Interpretability**: Develop mechanisms to better understand the interaction between retrieved contexts and the generative process.

These directions can guide future research to overcome the limitations identified in this study and enhance the performance of retrieval-augmented systems for domain-specific tasks.

References

Insights, F. M. Conversational ai market share, trends forecast 2033. Technical report, Future Market Insights, London, UK, 2024. URL https://www.futuremarketinsights.com/reports/conversational-ai-market.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.-t., Rocktäschel, T., et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

Markets and Markets. Conversational ai market size, statistics, growth analysis trends. Technical report, Markets and Markets, North America, 2024.

Sawarkar, K., Mangal, A., and Solanki, S. R. Blended rag: Improving rag (retriever-augmented generation) accuracy with semantic search and hybrid query-based retrievers. *arXiv preprint arXiv:2404.07220*, 2024.

Zhang, T., Patil, S. G., Jain, N., Shen, S., Zaharia, M., Stoica, I., and Gonzalez, J. E. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*, 2024.