2024 7th International Conference on Data Science and Information Technology (DSIT) | 979-8-3503-8409-3/24/$31.00 ©2024 IEEE | DOI: 10.1109/DSIT61374.2024.10881380

An Effective Retrieval Method to Improve RAG Performance

1st Su Mengmeng*
North China Branch of State Grid Corporation of China
Beijing, China
mengmengSGCC@163.com

2nd Liu Zhibin
North China Branch of State Grid Corporation of China
Beijing, China
liu.zbin@nc.sgcc.com.cn

3rd Wang Qingwei
North China Branch of State Grid Corporation of China
Beijing, China
wang.qingwei@nc.sgcc.com.cn

4th Hu Man
North China Branch of State Grid Corporation of China
Beijing, China
hu.man@nc.sgcc.com.cn

5th Xu Feiyang
North China Branch of State Grid Corporation of China
Beijing, China
xu.feiyang@nc.sgcc.com.cn

***Abstract*—Although large language models (LLMs) have demonstrated impressive capabilities in generating coherent and fluent text, they often produce irrelevant content when tasked with specialized domains. To address these challenges, Retrieval-Augmented Generation (RAG) combines retrieval and generation processes to enhance the relevance, accuracy, and diversity of LLM responses. However, naive RAG approaches often struggle with precision and recall during the retrieval phase, leading to the selection of misaligned or irrelevant chunks, and sometimes missing critical information. In this work, we propose a novel approach that integrates both word-level and sentence-level retrieval techniques to optimize the retrieval process. By improving the alignment of retrieved information, our method addresses these precision and recall issues more effectively than traditional retrieval approaches. Extensive experiments on benchmark datasets show that our method is significantly superior to existing retrieval strategies, reducing computational overhead and improving accuracy. Our results not only highlight the effectiveness of advanced retrieval strategies in improving LLM performance but also demonstrate their practical implications for scaling NLP systems in real-world applications.**

***Index Terms*—large language model, retrieval augmented generation, recursive retrieval**

I. Introduction

Recent advancements in large language models (LLMs) have demonstrated their impressive capabilities across a wide range of natural language processing (NLP) tasks. However, traditional generative models, while capable of producing coherent text, often struggle with factual accuracy and contextual relevance, particularly in knowledge-intensive tasks. This is because LLMs are susceptible to ”hallucinations” when they require more knowledge than is available in their training data or current information to handle these tasks, resulting in inaccurate content. The conventional approach to addressing this issue is to fine-tune LLMs to a specific domain. While this technique is effective, it is resource-intensive and requires high-quality annotated data, and it is challenging to adapt flexibly to changing knowledge.

Retrieval augmentation generation (RAG) has been proposed as a more efficient method to address the aforementioned issues. RAG systems allow LLMs to access external knowledge sources during the generation process by incorporating a retrieval mechanism. This hybrid architecture enables the model to generate responses grounded in real-world information, thus improving overall output quality. Therefore, the performance of RAG retrieval directly impacts the quality of the responses generated by LLMs. As shown in figure 1, naive RAG strategy firstly splits the document into fixed-size chunks and then encodes each chunk using an embedding model. Subsequently, the semantic similarity between each chunk and the query is calculated, and the top-k chunks are selected to be provided to the LLMs. Nevertheless, conventional retrieval strategies often rely on basic semantic similarity, making them to struggle to capture fine-grained semantic knowledge, particularly when dealing with long texts. These methods frequently fall prey to noise in the data, which can lead to suboptimal retrieval results.

In this work, we propose a recursive retrieval strategy that operates at both the word and sentence levels. Specifically, our approach begins with the application of the WordPiece algorithm for tokenizing both queries and documents, followed by the removal of stop words from the tokenized results. This filtration process utilizes the tokens derived from the query to sift through documents, thereby generating a set of candidate documents. Subsequently, we implement recursive semantic

*Corresponding author

Science and Technology Project Funding of State Grid Corporation of China (Contract No.: Project No.: 52992324000C)

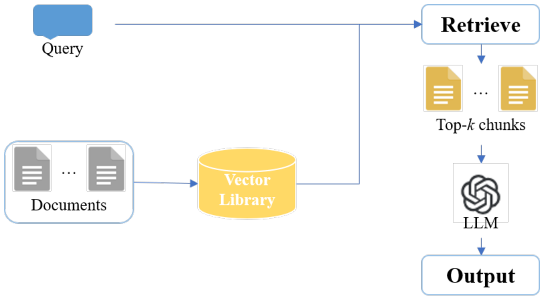

Fig. 1. Naive RAG workflows.

retrieval on these candidates to identify and recall relevant documents. We validate the effectiveness of this methodology across four publicly available datasets, demonstrating notable improvements in retrieval efficiency, particularly in scenarios involving large document corpora.

II. Related Work

The retrieval augmentation model [1] has demonstrated considerable ability in alleviating the hallucination of LLMs [2]–[5], particularly in tasks such as open-domain question answering [6], and domain-specific question answering [7], [8]. [9] investigate methods for guiding language models through the incorporation of a fixed number of search paragraphs within the input. [10] conduct two stages of fine-tuning of the retriever and language model to optimize model performance. [11] provid a summary before prompting the language model to generate the output. Furthermore, [12] employs an off-the-shelf language model to search for pertinent information for Q&A tasks. A distinct advantage of RAG over fine-tuning is its adaptable architecture, which enhances model effectiveness and reduces expenses in practical applications, thereby making it the most prevalent approach for a diverse range of proprietary domains.

While these efforts have yielded impressive results, the granularity of document segmentation is challenging to grasp in practice. The chunk size is related to the degree of fit with the query. A chunk size that is too small may fail to cover all the relevant content, on the contrary, a chunk size that is too large may result in the introduction of superfluous information, which could interfere with the answer of LLMs. Moreover, if the document is of an excessive length, vectors of fixed dimensions may prove ineffective in distinguishing subtle differences between similar semantics. In comparison to traditional RAG approaches, where the document is split into fixed-size chunks, our recursive retrieval strategy addresses these limitations by dynamically refining the granularity of retrieval. This method not only ensures that relevant content is retrieved with greater precision, but it also minimizes the inclusion of irrelevant information that could hinder model performance. By optimizing both retrieval granularity and content selection, our approach improves the overall accuracy of the model in a way that traditional methods, such as chunk-based retrieval or semantic similarity measures, may struggle to match.

III. Methodology

A. Method Overview

In this study, we employ a recursive retrieval strategy to enhance the recall rate of information retrieval. Figure 2 shows an overview of this recursive retrieval method. The process begins with tokenization of both the query and the documents, enabling a word-level retrieval approach. Initially, we filter documents that contain tokens from the query, establishing a set of candidate documents at the lexical granularity. Subsequently, we delve into a semantic retrieval phase, where we compute the semantic similarity between the query and the candidate documents derived from the initial token-based filtering. This recursive evaluation culminates in the selection of the top-k documents based on their similarity scores, thereby optimizing both the precision and recall of the information retrieval system.

Algorithm 1 Word-Level Retrieval

```
Input: input queries $Q = \{q_1, \ldots, q_N\}$, document chunks $C = \{c_1, \ldots, c_M\}$, stopwords list $\mathcal{S}$
Output: the tokenization results of the query $\mathcal{T}_q$ and chunks $\mathcal{T}_c$ ,retrieval results for word granularity $\mathcal{R}_{word}$
$\mathcal{T}_q = \emptyset, \mathcal{T}_c = \emptyset, \mathcal{R}_{word} = \emptyset$
for $m = 1 \rightarrow M$ do
    tokens $= tokenize\,(c_m)$
    tokens $= remove\_stopwords\,$(tokens)
    $\mathcal{T}_c \Leftarrow \mathcal{T}_c \cup$ tokens
end for
for $n = 1 \rightarrow N$ do
    tokens $= tokenize\,(q_n)$
    tokens $= remove\_stopwords\,$(tokens)
    $\mathcal{T}_q \Leftarrow \mathcal{T}_q \cup$ tokens
end for
for $i = 1, 2, \ldots, N$ do
    for $j = 1, 2, \ldots, M$ do
        if $\mathcal{T}_q^i \cap \mathcal{T}_c^j$ then
            $\mathcal{R}_{word}^i \Leftarrow \mathcal{R}_{word}^i \cup c_j$
        end if
    end for
end for
return $\mathcal{T}_q, \mathcal{T}_c, \mathcal{R}_{word}$
```

B. Word-Level Retrieval

Both input queries and documents undergo thorough cleaning to eliminate extraneous whitespace, punctuation, and other non-essential elements. This step is critical for facilitating accurate tokenization. The tokenization employs the WordPiece algorithm, which sequentially analyzes each word. Words present in the constructed vocabulary are retained in their entirety, while those absent are decomposed into smaller subword units and retained. Following tokenization, the method advances to the removal of stop words, which are

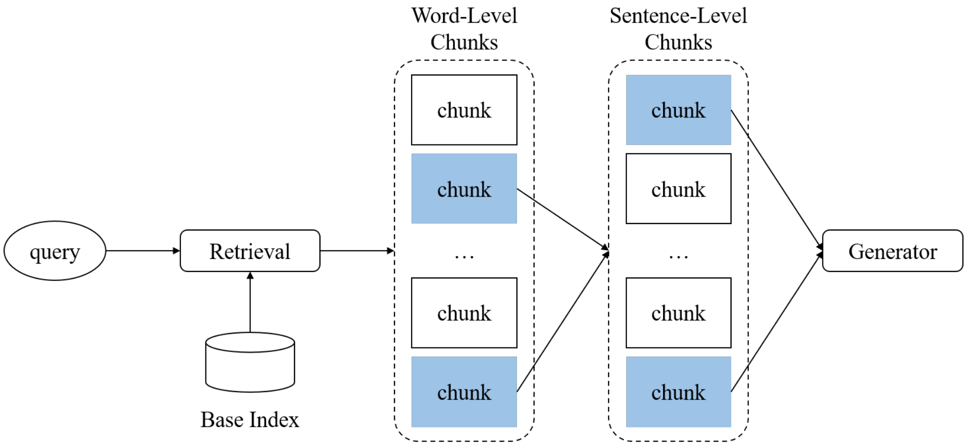

Fig. 2. Overview of the proposed method.

commonly defined as frequently occurring words that offer minimal semantic value to the overall comprehension of the text. This step results in a refined set of significant tokens that effectively represent the core meaning of both the query and the documents.

Following the tokenization phase, the next step involves filtering the document corpus based on the tokens generated from the input query. This process enhances the retrieval system's efficiency by narrowing down the pool of candidate documents. For each document in the corpus, we conduct a search for the presence of the query tokens. Documents that contain at least one of the query tokens are marked as potential matches. This is achieved by performing an intersection between the tokens of the query and the tokens of each document.

By utilizing the WordPiece tokenization method and implementing a robust document filtering strategy, our retrieval approach aims to enhance both the precision and recall of the information retrieval system, effectively responding to user queries with relevant document selections. This method not only improves retrieval efficiency but also aligns with the morphological complexities of natural language, facilitating a deeper understanding of user intent. The complete algorithm flow is shown in Algorithm 1.

C. Sentence-Level Retrieval

Various text representation methods, such as TF-IDF, Word2Vec, or BERT embeddings, can be utilized for encoding. In this method, we focus on advanced embeddings like BERT, which capture contextual relationships within the text. Each query and document is processed through the selected model, generating high-dimensional vectors that encapsulate semantic information. Once the queries and documents have been encoded into vector representations, the next step involves calculating the semantic similarity between them. The cosine similarity metric is employed to assess the degree of similarity between the encoded query vector q and each document vector d_i. It can be formalized as:

$$\text{similarity}\left(q, d_i\right) = \frac{q \cdot d_i}{\|q\| \|d_i\|} \tag{1}$$

After computing the similarity scores, the documents are sorted in descending order based on their cosine similarity scores, allowing for an intuitive prioritization of results. This sorting ensures that the documents most aligned with the query semantics are presented first. From the sorted list, the top k documents are selected as the final output.

IV. Experiment

A. Dataset

In our experimental evaluation of the proposed retrieval method, we utilized four benchmark datasets: rag-mini-wikipedia, SQuAD2.0 [13], NarrativeQA [14] and HotpotQA [15]. Each of these datasets is designed to facilitate the assessment of relation extraction tasks, providing a robust foundation for measuring the effectiveness of retrieval-augmented generation (RAG) systems. To evaluate the performance of our retrieval method across these datasets, we employed two primary metrics: Hit Rate (Hits@N) and Mean Reciprocal Rank (MRR@N).

B. Implementation Details

In our experiments, we utilized the BERT tokenizer for text preprocessing, which effectively converts raw text into subword tokens, accommodating out-of-vocabulary terms and enhancing semantic representation. For stop word removal, we incorporated a stop word list from the Natural Language Toolkit (NLTK). The semantic encoding was performed using the bge-large-en-v1.5 model. All experiments were executed on a single NVIDIA A100 GPU.

https://huggingface.co/datasets/rag-datasets/rag-mini-wikipedia

TABLE I
Comparison of Hits@N Metrics

Method	rag-mini-wikipedia			SQuAD2.0			NarrativeQA			HotpotQA		
	Hits@1	Hits@3	Hits@10	Hits@1	Hits@3	Hits@10	Hits@1	Hits@3	Hits@10	Hits@1	Hits@3	Hits@10
Naive RAG	0.69	0.86	0.92	0.70	0.86	0.95	0.59	0.64	0.74	0.80	0.95	0.98
Ours	0.70	0.89	0.95	0.72	0.89	0.96	0.62	0.72	0.80	0.80	0.95	0.98

TABLE II
Comparison of MRR@N Metrics

Method	rag-mini-wikipedia			SQuAD2.0			NarrativeQA			HotpotQA		
	MRR@1	MRR@3	MRR@10	MRR@1	MRR@3	MRR@10	MRR@1	MRR@3	MRR@10	MRR@1	MRR@3	MRR@10
Naive RAG	0.69	0.75	0.76	0.70	0.77	0.79	0.59	0.58	0.60	0.80	0.87	0.88
Ours	0.70	0.78	0.80	0.72	0.79	0.81	0.62	0.66	0.68	0.80	0.87	0.88

TABLE III
Retrieval Efficiency Comparison

	Naive RAG	Ours	Hardware Configuration
NarrativeQA	5.50 it/s	82.74 it/s	NVIDIA A100, 40GB RAM
HotpotQA	5.25 it/s	34.29 it/s	NVIDIA A100, 40GB RAM

C. Main Result

Table I illustrate the hit rate of our proposed method against the baseline in different datasets. The experimental results suggest that the recursive retrieval strategy proposed in this study significantly enhances the retrieval performance across diverse datasets. The consistent improvements in the Hits@N metrics indicate that our method not only increases the number of relevant documents retrieved but also improves the ranking of these documents, particularly at the top positions.

Table II provide a comprehensive comparison of the Mean Reciprocal Rank (MRR@N) metrics for our proposed method against a baseline across four distinct datasets. The experimental results elucidate the efficacy of our proposed retrieval method across multiple datasets, Specifically, the findings confirm that our method achieves superior retrieval accuracy compared to the baseline, where the gains in MRR metrics indicate not just enhanced retrieval but also better ranking of relevant documents. Overall, our model maintains effectiveness without introducing unnecessary complexity.

The performance gains are observed to vary across datasets, with more pronounced improvements seen in NarrativeQA compared to HotpotQA. This difference can be attributed to the unique nature of the datasets: NarrativeQA involves long-form narrative questions where relevant context is often spread across multiple paragraphs, which our recursive retrieval method handles more effectively. In contrast, HotpotQA consists of multi-hop questions that require reasoning over multiple facts. Our method performs well here too, but the gains are relatively smaller, suggesting that further refinement of the retrieval strategy could enhance performance on such tasks.

D. Efficiency

To further evaluate the efficiency of our proposed method, we conducted experiments on two large-scale datasets: NarrativeQA and HotpotQA. The measured efficiency is expressed in iterations per second (it/s), which serves as an indicator of how quickly each method can process retrieval tasks. The findings underscore the effectiveness of our proposed retrieval method in enhancing efficiency without sacrificing accuracy. Such a substantial improvement in processing efficiency suggests that our method not only optimizes the underlying algorithmic structure but also enhances the overall computational performance. This efficiency gain is particularly crucial for applications that demand real-time responses, such as interactive question-answering systems or dynamic content generation.

Such significant improvements in processing efficiency have profound practical implications. In real-time applications, such as interactive question-answering systems, these gains translate directly into faster response times, enhancing user experience by providing near-instantaneous answers to complex queries. For dynamic content generation platforms, the ability to handle large volumes of retrieval tasks more efficiently could significantly reduce latency and enable more responsive and scalable systems. Furthermore, this efficiency gain reduces the strain on hardware resources, potentially lowering computational costs, particularly in cloud-based environments where large-scale inference tasks can become costly. This improvement is especially crucial in scenarios that require high throughput or must handle massive datasets in real-time, such as search engines and personalized recommendation systems. Our method's ability to optimize retrieval speed without sacrificing accuracy not only demonstrates its algorithmic efficiency but also underscores its suitability for deployment in performance-critical applications where both speed and accuracy are essential.

V. Conclusion

This work introduces a novel recursive retrieval method that employs a two-tiered approach. The primary contribution of our method lies in its ability to leverage the granularity of word-based retrieval to filter and refine potential candidate documents before advancing to a more refined sentence-level search. This dual-layered strategy not only minimizes the computational overhead associated with processing large volumes of text but also ensures that the most relevant information is prioritized. Our experimental results demonstrate that this method significantly improves retrieval efficiency and accuracy, with specific performance gains such as an increase in retrieval speed from 5.50 iterations per second (it/s) to 82.74 it/s on the NarrativeQA dataset, as well as a 3-8% improvement in hits rate over baseline methods.

While these results are promising, there are some limitations to consider. The method's scalability in extremely large datasets, especially in the case of domain-specific retrieval tasks, may face challenges related to resource allocation and processing times for very specialized or sparse data. Additionally, the method's performance in multilingual contexts remains untested and may require further adaptation to handle language-specific nuances or cross-lingual retrieval tasks effectively.Future work could explore the applicability of this approach on more diverse datasets, including non-English languages, and investigate its potential for handling multimodal information retrieval tasks. Further research could also focus on optimizing the retrieval strategy for domain-specific applications and exploring hybrid methods that integrate external knowledge sources to improve performance across a broader range of use cases.

References

[1] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," in *Proceedings of the 34th International Conference on Neural Information Processing Systems*, ser. NIPS '20. Red Hook, NY, USA: Curran Associates Inc., 2020.

[2] S. Borgeaud, A. Mensch, J. Hoffmann, T. Cai, E. Rutherford, K. Millican, G. B. Van Den Driessche, J.-B. Lespiau, B. Damoc, A. Clark, D. De Las Casas, A. Guy, J. Menick, R. Ring, T. Hennigan, S. Huang, L. Maggiore, C. Jones, A. Cassirer, A. Brock, M. Paganini, G. Irving, O. Vinyals, S. Osindero, K. Simonyan, J. Rae, E. Elsen, and L. Sifre, "Improving language models by retrieving from trillions of tokens," in *Proceedings of the 39th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, K. Chaudhuri, S. Jegelka, L. Song, C. Szepesvari, G. Niu, and S. Sabato, Eds., vol. 162. PMLR, 17–23 Jul 2022, pp. 2206–2240. [Online]. Available: https://proceedings.mlr.press/v162/borgeaud22a.html

[3] G. Izacard, P. Lewis, M. Lomeli, L. Hosseini, F. Petroni, T. Schick, J. Dwivedi-Yu, A. Joulin, S. Riedel, and E. Grave, "Atlas: few-shot learning with retrieval augmented language models," *J. Mach. Learn. Res.*, vol. 24, no. 1, Mar. 2024.

[4] O. Ram, Y. Levine, I. Dalmedigos, D. Muhlgay, A. Shashua, K. Leyton-Brown, and Y. Shoham, "In-context retrieval-augmented language models," *Transactions of the Association for Computational Linguistics*, vol. 11, pp. 1316–1331, 2023. [Online]. Available: https://aclanthology.org/2023.tacl-1.75

[5] W. Shi, S. Min, M. Yasunaga, M. Seo, R. James, M. Lewis, L. Zettlemoyer, and W.-t. Yih, "REPLUG: Retrieval-augmented black-box language models," in *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, K. Duh, H. Gomez, and S. Bethard, Eds. Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 8371–8384. [Online]. Available: https://aclanthology.org/2024.naacl-long.463

[6] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal, "Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds. Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 10 014–10 037. [Online]. Available: https://aclanthology.org/2023.acl-long.557

[7] J. Cui, M. Ning, Z. Li, B. Chen, Y. Yan, H. Li, B. Ling, Y. Tian, and L. Yuan, "Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model," 2024. [Online]. Available: https://arxiv.org/abs/2306.16092

[8] D. Van Veen, C. Van Uden, L. Blankemeier, J.-B. Delbrouck, A. Aali, C. Bluethgen, A. Pareek, M. Polacin, E. P. Reis, A. Seehofnerova *et al.*, "Clinical text summarization: adapting large language models can outperform human experts," *Research Square*, 2023.

[9] H. Luo, Y.-S. Chuang, Y. Gong, T. Zhang, Y. Kim, X. Wu, D. Fox, H. Meng, and J. Glass, "Sail: Search-augmented instruction learning," 2023. [Online]. Available: https://arxiv.org/abs/2305.15225

[10] X. V. Lin, X. Chen, M. Chen, W. Shi, M. Lomeli, R. James, P. Rodriguez, J. Kahn, G. Szilvasy, M. Lewis, L. Zettlemoyer, and W. tau Yih, "RA-DIT: Retrieval-augmented dual instruction tuning," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=22OTbutug9

[11] F. Xu, W. Shi, and E. Choi, "RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation," in *The Twelfth International Conference on Learning Representations*, 2024. [Online]. Available: https://openreview.net/forum?id=mlJLVigNHp

[12] A. Zhou, K. Yan, M. Shlapentokh-Rothman, H. Wang, and Y.-X. Wang, "Language agent tree search unifies reasoning acting and planning in language models," 2024. [Online]. Available: https://openreview.net/forum?id=6LNTSrJjBe

[13] P. Rajpurkar, R. Jia, and P. Liang, "Know what you don't know: Unanswerable questions for SQuAD," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, I. Gurevych and Y. Miyao, Eds. Melbourne, Australia: Association for Computational Linguistics, Jul. 2018, pp. 784–789. [Online]. Available: https://aclanthology.org/P18-2124

[14] T. Kočiský, J. Schwarz, P. Blunsom, C. Dyer, K. M. Hermann, G. Melis, and E. Grefenstette, "The NarrativeQA reading comprehension challenge," *Transactions of the Association for Computational Linguistics*, vol. 6, pp. 317–328, 2018. [Online]. Available: https://aclanthology.org/Q18-1023

[15] Z. Yang, P. Qi, S. Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and C. D. Manning, "HotpotQA: A dataset for diverse, explainable multi-hop question answering," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, E. Riloff, D. Chiang, J. Hockenmaier, and J. Tsujii, Eds. Brussels, Belgium: Association for Computational Linguistics, Oct.-Nov. 2018, pp. 2369–2380. [Online]. Available: https://aclanthology.org/D18-1259