electronics

MDPI

Article

Adaptive Control of Retrieval-Augmented Generation for Large Language Models Through Reflective Tags

Chengyuan Yao and Satoshi Fujita *

Department of Information Science, Graduate School of Advanced Science and Engineering, Hiroshima University, Kagamiyama 1-4-1, Higashi-Hiroshima 739-8527, Japan
* Correspondence: fujita@hiroshima-u.ac.jp

Abstract: While retrieval-augmented generation (RAG) enhances large language models (LLMs), it also introduces challenges that can impact accuracy and performance. In practice, RAG can obscure the intrinsic strengths of LLMs. Firstly, LLMs may become too reliant on external retrieval, underutilizing their own knowledge and reasoning, which can diminish responsiveness. Secondly, RAG may introduce irrelevant or low-quality data, adding noise that disrupts generation, especially with complex tasks. This paper proposes an RAG framework that uses reflective tags to manage retrieval, evaluating documents in parallel and applying the chain-of-thought (CoT) technique for step-by-step generation. The model selects the highest quality content for final output. The key contributions are as follows: (1) reducing hallucinations by focusing on high-scoring documents; (2) improving real-time performance through efficient retrieval; and (3) mitigating negative effects by filtering out irrelevant information using parallel generation and reflective tagging. These innovations aim to optimize RAG for more reliable, high-quality results.

Keywords: retrieval-augmented generation; large language models; chain of thought; reflective tag

Citation: Yao, C.; Fujita, S. Adaptive Control of Retrieval-Augmented Generation for Large Language Models Through Reflective Tags. *Electronics* **2024**, *13*, 4643. https://doi.org/10.3390/electronics13234643

Academic Editor: Dah-Jye Lee

Received: 28 August 2024
Revised: 8 November 2024
Accepted: 11 November 2024
Published: 25 November 2024

1. Introduction

The advent of large language models (LLMs) has revolutionized natural language processing (NLP), enabling significant advancements in a wide array of applications such as text generation, translation, and question answering. However, these models still face notable challenges, particularly in handling domain-specific or highly specialized queries, where the generated answers often lack context or coherence. To address these limitations, the retrieval-augmented generation (RAG) framework was introduced [1,2], merging the generative capabilities of LLMs with the precision of external knowledge retrieval.

RAG leverages external knowledge sources by retrieving relevant documents and integrating their content into the generation process. Such a dual-step approach, i.e., comprising retrieval of pertinent information and subsequent generation of responses, enhances the relevance and accuracy of the outputs. Despite its efficacy, however, there remains substantial room for improvement in the RAG framework, particularly in refining retrieval mechanisms, optimizing document evaluation, and enhancing overall response quality.

This paper explores recent advancements in the RAG framework, focusing on methods to improve retrieval precision, integrate context-aware document evaluation, and streamline the generation process. We review notable studies and propose novel enhancements aimed at addressing the existing limitations of the RAG, ultimately contributing to more reliable and contextually appropriate outputs from LLMs.

Our Viewpoints and Contributions

While RAG provides significant benefits to LLMs, it also introduces potential issues that can impact the accuracy and performance of these models. Practical applications have revealed several limitations of RAG when applied to LLMs, which can mask the inherent

capabilities of the models themselves. Firstly, LLMs may become overly dependent on external retrieval. Excessive reliance on RAG techniques to fetch external knowledge can cause LLMs to underutilize their own knowledge and inference capabilities, potentially reducing the responsiveness and performance of the model. LLMs possess robust language understanding and generation abilities, and an overemphasis on retrieving external information can limit exploiting these intrinsic capabilities. Secondly, RAG techniques might introduce irrelevant or low-quality information, leading to a phenomenon where noise is injected into the LLM. Such irrelevant or poor-quality information can disrupt the normal generation process, decreasing the efficiency and quality of the generated content. For instance, when handling complex problems, the model may confuse significant information with trivial details, resulting in responses that lack precision and usefulness.

In this paper, we propose an RAG framework that controls the retrieval of external sources using reflective tags. This framework evaluates retrieved documents from several aspects and incorporates the chain-of-thought (CoT) technique for step-by-step content generation. The highest quality and most accurate content is then selected for final content generation. The main contributions of this paper are as follows:

- The proposed framework mitigates the hallucination in LLMs by using RAG. With the notion of tags, the model controls retrieval and self-evaluates the retrieved content, selecting only high-scoring information for generation, thus further reducing hallucinations.
- By incorporating RAG, the model can retrieve external databases for answers when it judges that it cannot answer effectively. The real-time nature of the external data enhances the model's real-time performance.
- The framework allows for parallel content generation for retrieved documents and step-by-step evaluation using reflective tags to assess relevance, validity, and accuracy. This process ensures that only the most critical and valid information is selected, filtering out irrelevant or unreliable information and generating more accurate responses.

These advancements aim to refine the integration of retrieval mechanisms with LLMs, ensuring that the strengths of both internal model capabilities and external knowledge sources are effectively harnessed to provide high-quality and reliable outputs.

The proposed method has significant potential across various domains that exhibit a strong synergy with LLMs. First, the medical and healthcare sector stands to benefit in areas such as diagnostic support, clinical data analysis, and personalized treatment pathways [3]. By enabling real-time access to the medical literature, clinical databases, and electronic health records (EHRs), the method facilitates the generation of responses grounded in the latest treatment protocols and medical knowledge. Such a capability allows healthcare professionals to formulate optimal diagnostic and treatment strategies with greater reliability. Second, in the field of finance and economics, the method can enhance financial analysis, investment report generation, and risk management processes [4]. By capturing real-time data from financial markets, corporate financial disclosures, and economic news, the method supports the generation of precise market analyses and investment strategies. Such a continuous data-driven approach provides financial institutions and investors with critical insights, enabling decisions based on up-to-date information. Third, the method offers promising applications in the advertising and marketing sectors. For instance, in the generation of marketing content and advertising copy, the use of RAG enables the incorporation of the latest market trends and consumer data, fostering innovation in content creation for the creative industries.

The remainder of the paper is organized as follows. Section 2 reviews related work on improving RAGs. Section 3 presents the details of the proposed method, outlining its architecture, implementation, and theoretical underpinnings. Section 4 provides an experimental evaluation of the proposed method's performance, including datasets, experimental setup, metrics, and results. Finally, Section 5 summarizes the paper's contributions and discusses potential directions for future work.

2. Retrieval-Augmented Generation (RAG)

LLMs have known weaknesses, such as producing answers that do not match or contradict the given context. This issue is particularly pronounced when addressing domain-specific or highly specialized queries. To mitigate these weaknesses, a technique called retrieval-augmented generation (RAG) was proposed in 2020 [1,2]. The core idea of RAG is to integrate data obtained from querying external knowledge sources into the generation process. By leveraging the accuracy and specificity of knowledge from external sources, RAG enhances the generative process, improving the ability to provide highly relevant and real-time responses to queries. This section provides an overview of RAG models and reviews existing research aimed at enhancing their performance.

2.1. Workflow of RAG

The RAG workflow comprises two main steps: corpus retrieval and content generation. In the first step, a retriever scans a large corpus to fetch documents relevant to the user's query. In the subsequent step, a reader component analyzes these retrieved documents to generate an answer to the query. Ideally, it should consider all documents containing potentially useful information. However, due to time and cost constraints, RAG utilizes the top k documents $d_1, d_2, \ldots, d_k$ from the retrieval step to calculate the conditional probabilities as follows:

$$p(y|x) = \sum_{i=1}^{k} p(y|d_i, x)p(d_i|x), \tag{1}$$

where d_i is the i^{th} retrieved document, x is the context of the input query or dialogue and y is the response generated by the model. Equation (1) is then calculated for all possible output y, and y^* satisfying the following equation is output as the response to the query:

$$p(y^*|x) = \arg\max_{y} p(y|x).$$

However, not all of $d_1, d_2, \ldots, d_k$ were relevant to the user input and generally contained duplicate or incorrect information. As a result, the responses generated by the RAG were unsatisfactory and the generated content still contained irrelevant or incorrect information.

2.2. Existing Research on Improving RAGs

Recently, Chen et al. conducted a comprehensive survey on the effects of RAG on LLMs [5]. In their study, the authors examine the performance of various LLM systems based on four essential capabilities for RAG: noise tolerance, negation filtering, information integration, and out-of-hypothesis tolerance. To support this evaluation, they developed a novel dataset called the retrieval-augmented generation benchmark (RGB) and evaluated six representative LLM systems on this benchmark. The evaluated models include ChatGLM2-6B [6], Vicuna-7B-v1.3 [7], Qwen-7B-Chat [8], and BELLE- 123 7B-2M [9].

The evaluation results demonstrate that while LLMs exhibit some degree of noise tolerance, significant challenges remain in negation handling, information integration, and dealing with false information.

Before discussing the details of the proposed method, an overview of previously proposed improvements to RAGs is provided. Recent studies have shown that context relevance significantly affects the performance of LLMs. For instance, Creswell et al. [10] found that the inclusion of random or irrelevant documents negatively impacts system performance and proposed a structured inference (SI) framework. This framework enhances inference correctness by alternating between the discovery of relevant knowledge and inference results. Similarly, Yoran et al. [11] focused on training a retrieval knowledge augmentation model that ignores irrelevant context, demonstrating that high context relevance contributes to improved performance. Although the starting point of their study is similar to ours, the proposed solutions differ substantially.

It has also been shown that fine-tuning the model for knowledge-intensive tasks can significantly enhance the performance of LLMs. For example, Ram et al. [12] considered a simple alternative called an in-context retrieval-augmented language model (RALM), which incorporates documents containing additional information or background knowledge before input without modifying the LM architecture or requiring further training. They demonstrated that performance could be further improved by tailoring document retrieval and ranking mechanisms specific to the RALM setting. Izacard et al. [13] proposed an RAG called Atlas, capable of learning knowledge-intensive tasks with few training examples. Additionally, Luo et al. [14] introduced Search-Assisted Instruction Learning (SAIL), which involves collecting search results for each training case from various search APIs and domains using an instruction-tuning corpus, and constructing a training set.

While the above-mentioned methods limit the number of queries to external resources, the recent literature has explored increasing the number of searches. Mallen et al. [15] conducted experiments on two open-domain QA datasets, confirmed that LLMs struggle with less general knowledge, and that retrieval augmenting significantly aids in such cases. They found that scaling the number of parameters improves the memorization of general knowledge but does not significantly enhance the recall of long-tail factual knowledge. Based on this finding, they devised a new retrieval augmentation method that achieves improved performance and reduced inference costs by retrieving non-parametric memories only when necessary. Jiang et al. [16] proposed a method for actively deciding what to search for and when during the generation process. They noted that continuous information gathering during generation is essential in more general scenarios involving long sentence generation. The authors proposed FLARE, a general-purpose method that iteratively predicts the next sentence to forecast future content, using it as a query to retrieve relevant documents to regenerate a sentence when it contains unreliable tokens. This method significantly impacts runtime efficiency as it iteratively and constantly searches during generation and resumes searching as soon as a sentence from the generation session contains a low-trust flag. However, this is not the case in our framework, where only high-scoring content is selected for generation.

While the aforementioned studies propose various improvements to the RAG framework, it is important to highlight complementary research that focuses on the robustness of fine-tuning methods. For example, Oh et al. [17] emphasize that improving out-of-distribution (OOD) generalization through in-distribution (ID) adaptation is a key aspect of robust fine-tuning techniques for RAG. Their work not only aims to enhance model accuracy but also proposes a method to improve calibration, ensuring more reliable predictions.

3. Proposed Method

This section outlines the proposed framework. Figure 1 illustrates the workflow of our approach, which extends the original RAG paradigm by incorporating chain-of-thought (CoT) reasoning and a novel tagging mechanism for comprehensive content evaluation. Unlike traditional RAGs, our method integrates retrieval and generation processes, allowing for dynamic retrieval based on evolving generation requirements. The subsequent subsections provide a detailed description of the external source retrieval process (Section 3.1), the document evaluation methodology (Section 3.2), and the generation of response content based on evaluated documents (Section 3.3).

3.1. Document Retrieval

Document retrieval in the proposed framework is performed using cosine similarity as a measure of closeness, where the way for vectorizing a given sentence will be described later. Note that the cosine similarity values of the vectors u and v corresponding to two sentences range from -1 to 1. A value of 1 indicates that the two vectors are perfectly matched, implying the sentences are similar, whereas a value of 0 indicates that the vectors are orthogonal and thus not similar. A value of -1 indicates that u and v are completely opposite in direction.

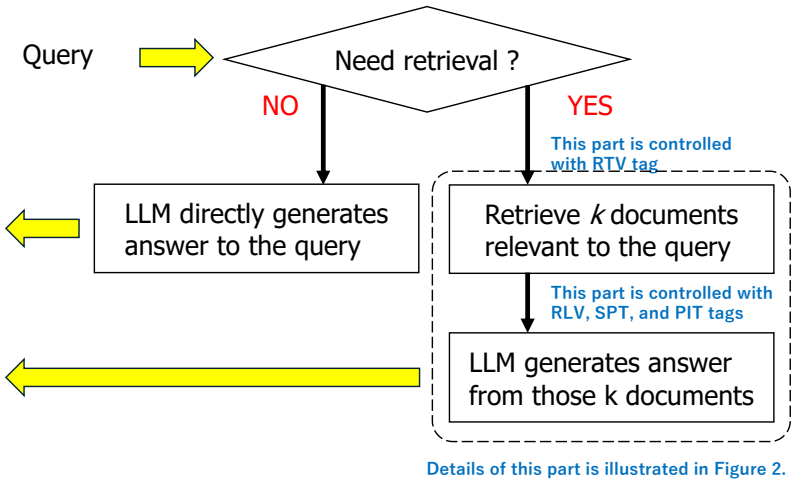

Figure 1. Proposed framework.

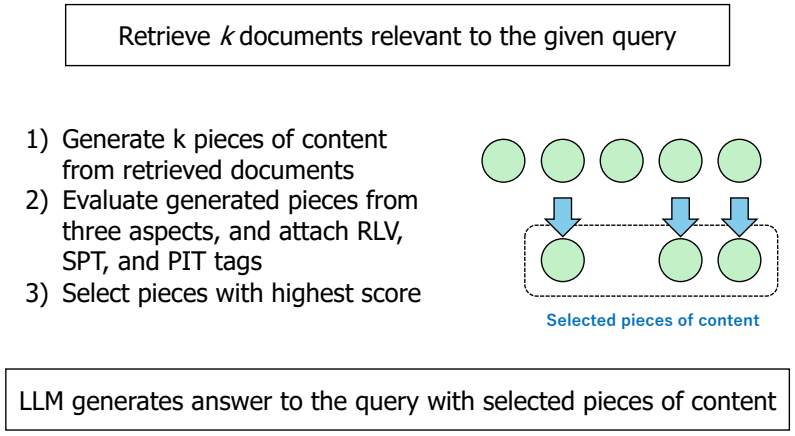

Figure 2. Workflow of content generation.

This document retrieval method offers the following advantages: Firstly, the computational complexity of cosine similarity of two vectors is $O(n)$, where n is the length of the vectors, making it suitable for large datasets where numerous large-sized vectors must be handled. Secondly, since cosine similarity depends only on the angles between vectors and not their lengths, its validity remains intact even after normalizing the vectors. This is particularly advantageous when dealing with data of different sizes and scales. Additionally, cosine similarity is influenced only by non-zero elements of vectors, making it suitable for NLP and information retrieval scenarios where vectors often have high dimensions but few non-zero elements.

The proposed framework performs sentence vectorization using Sentence-BERT. Unlike standard BERT, which produces contextual embeddings for individual tokens, Sentence-BERT generates embeddings for entire sentences, which makes it highly effective in various tasks such as semantic text similarity, paraphrase mining, and clustering. Structurally, Sentence-BERT is a BERT model with an additional pooling layer. The purpose of the pooling layer is to convert the output of BERT into a fixed-dimensional sentence vector, enhancing Sentence-BERT's performance on sentence-level tasks. Specifically, when assessing the similarity between two sentences, A and B, both sentences are input into the same BERT model with identical parameters. Through pooling, the feature vector u for sentence A and the feature vector v for sentence B are extracted. The similarity between u and v is then calculated using cosine similarity.

To optimize the model, the mean squared error (MSE) is employed as a loss function to measure the discrepancy between the predicted similarity and the actual value. Addi-

tionally, the regression objective function (ROF) method is utilized in Sentence-BERT for this purpose.

3.2. Control Document Retrieval Using Reflection Tags

Upon receiving user input, the model determines whether a search of external knowledge sources is necessary. If a search is not required, the LLM directly generates content to answer the query, but if a search is needed, it generates a retrieve (RTV) tag to initiate the search.

Specifically, the model calculates the probability distribution of all possible subsequent outputs, including RTV tags and general vocabulary, through forward propagation based on user input and context, which includes previously generated text passages and dialogue history. The model predicts whether the question can be answered using the knowledge within the LLM based on this probability distribution. If the prediction indicates that the question cannot be answered with the existing knowledge, the model generates an RTV tag.

3.3. Evaluation of Retrieved Documents

In the answer generation process, the quality of documents retrieved from external knowledge sources is progressively evaluated and filtered using three types of tags: Relevance (RLV), Support (SPT), and Point (PIT). Documents deemed relevant to the input are assigned the RLV tag. Those judged to support the answers to user questions are given the SPT tag. Finally, the PIT tag is used to score the documents for final content generation by the LLM, integrating the content of the RLV and SPT evaluations.

Let $d_1, d_2, \ldots, d_i, \ldots$ be documents obtained as a result of querying external knowledge sources. These documents are retrieved in parallel, and each document d_i is evaluated independently upon retrieval, receiving evaluation tags y_{d_i}. The specific evaluation procedure is as follows:

1. Evaluate whether document d_i is relevant to the user query, and assign one of the following RLV tags: "Relevant" or "Irrelevant", where the cosine similarity of feature vectors is used to assess the relevance. More specifically, after calculating the cosine similarity between vector $\vec{u} = [u_1, u_2, \ldots u_n]$ corresponding to the query and vector $\vec{v} = [v_1, v_2, \ldots, v_n]$ corresponding to the document as

$$cosine\_similarity(\vec{u}, \vec{v}) = \frac{\vec{u} \cdot \vec{v}}{|\vec{u}||\vec{v}|} = \frac{\sum_{i=1}^{n} u_i v_i}{\sqrt{\sum_{i=1}^{n} u_i^2} \sqrt{\sum_{i=1}^{n} v_i^2}}.$$

   The relevance to the query is determined by comparing the result to a preset threshold.
2. Assess whether the document d_i supports the input, and based on the degree of support, assign one of the following SPT tags: "Fully Supported", "Partially Supported", or "Not Supported".
3. Determine the suitability of document d_i for use in the RAG framework, and assign one of the PIT tags ranging from 1 to 5, where 5 means Highly Appropriate and 1 means Least Appropriate.

Documents with the highest score from these evaluations are selected for the content generation process described below. The reader should note that the current prototype's tagging does not fully rely on a strong theoretical background. This is because one of the main goals of this paper is to empirically demonstrate that tagging can improve the quality of responses. As discussed in the ablation study of Section 4, the presence of tags positively impacts relevance, query support, and response appropriateness, but it needs further improvements to refine the concrete tagging process.

3.4. Content Generation

Finally, we outline the content generation process used in the proposed framework. We will illustrate the process through concrete examples, with symbolic representations for brevity. Table 1 maps these symbols to their corresponding sentences.

Table 1. Mapping of symbols and sentences used in the example concerned with content generation.

Q0	How did Van Gogh create "The Starry Night"?
I0	Split the problem based on the steps in the sample below. Example: Why does the price of gold rise? Split the example question into the following: What are the main factors that affect the price of gold? How does economic uncertainty affect the price of gold? How does inflation affect the price of gold? How do supply and demand affect the price of gold?
Q1	What is the background of the creation of "The Starry Night"?
Q2	What techniques did Van Gogh use to create this painting?
Q3	What are the characteristics of the color and composition of this painting?
Q4	What is the significance of this painting in art history?
S0	"The Starry Night" is an oil painting created by Dutch post-impressionist painter Vincent van Gogh.
S1	Vincent van Gogh created "The Starry Night" in 1889 in a mental hospital in Saint-Remy, France.
S2	Now, it is in the collection of the Museum of Modern Art in New York.
S3	The inspiration came from the scenery he saw from the window of his room, combined with his memory and imagination.
S4	The detached-from-reality scene reflects Van Gogh's restless emotions and crazy hallucination world.

Upon receiving query Q0 from a user, the LLM generates an initial response S0 to the query without consulting external sources. If S0 is deemed insufficient, it generates an RTV tag to initiate a document retrieval process. Assuming four documents (S1, S2, S3, S4) are retrieved, the LLM concurrently commences content generation preparation.

Leveraging user-provided directives (I0), the LLM learns how to generate answers to complicated queries incrementally through a chain-of-thought (CoT) process (details of CoT are given in Appendix A). Suppose that successive queries (Q1, Q2, Q3, Q4) are derived from user query Q0 and directive I0. Then retrieved documents concurrently undergo evaluation and tagging based on relevance, support, and point alignment with the respective query. During the evaluation, additional document retrieval can occur as needed. Such an iterative process across Q1, Q2, Q3, and Q4 culminates in the final response to query Q0.

The above workflow is shown in Figure 2.

4. Evaluation

4.1. Setup

We conducted experiments to evaluate the performance of the proposed framework. Experiments were conducted on a Linux workstation equipped with an Intel Xeon Platinum 8358P CPU, a single RTX 3090 GPU, 30 GB of system memory, and 24 GB of video memory. The system ran Ubuntu 20.04 and Python 3.8. We employed four datasets for the evaluation: ARC-Challenge, PubHealth, PopQA, and TriviaQA, the details of which are explained as follows:

- ARC-Challenge: A fact-checking dataset comprising multiple-choice science questions from elementary to high school levels. The more challenging ARC-Challenge subset was utilized, requiring advanced reasoning. Preprocessing resulted in 1172 data points.
- PubHealth: A fact-checking dataset containing public health statements, corresponding articles, and fact-checking annotations. After preprocessing, 987 data points remained.
- PopQA: An open-domain question-answering dataset covering various domains. A long-tail subset, primarily from Wikipedia, was selected. Preprocessing yielded 1399 data points.

- TriviaQA: An open-domain question-answering dataset with 95,000 question-answer pairs. Known for its challenging long contexts and inference requirements, preprocessing reduced the dataset to 11,313 data points.

Accuracy was used as the evaluation metric for all datasets. This common metric assesses model performance by comparing predicted and ground-truth responses, and calculates the percentage of correct predictions. More formally, accuracy is calculated as

$$acc := \frac{\text{Number of correct answers}}{\text{Total number of questions in the dataset}} \times 100[\%].$$

To establish a strong baseline, the proposed method is compared against two widely adopted LLMs: GPT-3.5 and Qwen. Importantly, these baselines operate without the benefit of external knowledge sources. GPT-3.5, developed by OpenAI, represents a state-of-the-art language model characterized by optimized algorithms and architecture for enhanced text generation. Qwen, from Alibaba, is another prominent LLM demonstrating advanced capabilities in text generation and comprehension.

4.2. Comparison of Four Models

As an initial evaluation, we conducted experiments to evaluate the four models across the four datasets. The models under comparison comprised two baseline systems and their corresponding variants augmented with the proposed RAG framework. A summary of the results is presented in Table 2, where accuracy scores are reported, with percentage point improvements over the baseline models indicated in parentheses.

Table 2. Comparison of the accuracy of four models across four datasets.

Model	ARC-Challenge	PubHealth	PopQA-Long-Tail	TriviaQA
Qwen	81.83	58.42	59.97	64.64
GPT-3.5	77.99	68.70	56.90	72.86
Qwen + RAG	83.02 (+1.09)	65.51 (+7.09)	62.26 (+2.29)	66.42 (+1.78)
GPT-3.5 + RAG	81.57 (+3.58)	72.26 (+3.56)	57.68 (+0.78)	73.76 (+0.9)

Table 2 reveals a consistent enhancement in the performance of the LLM when integrated with the proposed RAG framework relative to its vanilla counterpart. This improvement is particularly pronounced on the fact-checking datasets, ARC-Challenge and PubHealth, where Qwen exhibited a 7.09 percentage point accuracy gain, rising from 58.42 to 65.51. While the Q&A tasks also recorded accuracy increases within the range of 0.7 to 2.3 points, the magnitude of improvement on TriviaQA was less substantial compared to PopQA. This disparity can be attributed to the baseline model's already high performance on TriviaQA, which includes a significantly larger dataset with more easily answerable questions compared to PopQA. The presence of challenging, inference-intensive instances within TriviaQA, despite its overall size, likely contributes to the muted accuracy gains.

4.3. Impact of the Number of Reference Passages K

The proposed RAG framework employs a hyperparameter, K, denoting the number of reference passages retrieved during the RAG process. An insufficient value of K may hinder performance due to inadequate knowledge acquisition, while excessive values can introduce redundancy and noise. To investigate the impact of K on the generated text quality, we conducted experiments with K ranging from 1 to 5. The dataset and evaluation metrics remain consistent with the previous experiment.

Figures 3 and 4 visualize the results, with the x-axis representing the value of K (e.g., rag1 for one retrieved passage, rag2 for two) and curves corresponding to different datasets. Figure 3 showcases the results for GPT-3.5 as the baseline model, while Figure 4 presents

those for Qwen. The figures indicate that the optimal number of retrieved passages for GPT-3.5 and Qwen is 2 and 4, respectively. These findings align with our hypothesis that careful selection of the number of retrieved passages is crucial for maximizing the benefits of the proposed RAG framework.

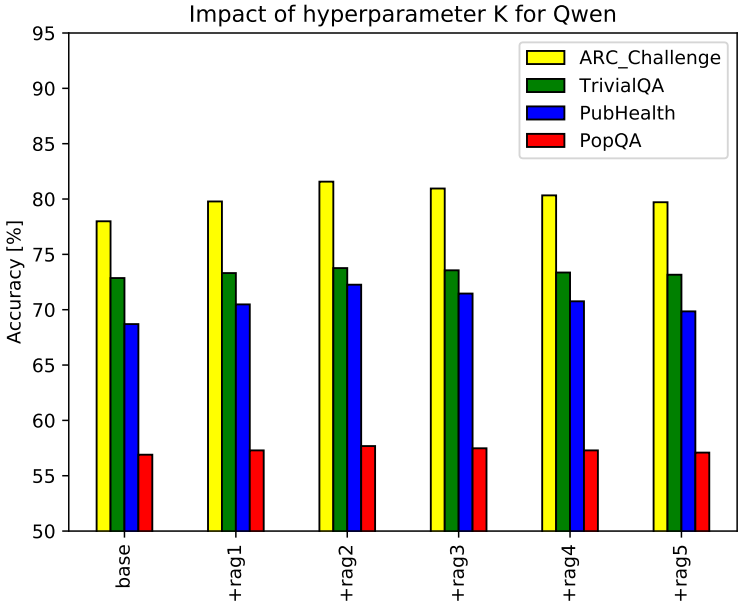

Figure 3. Impact of the number of reference passages *K* for GPT-3.5.

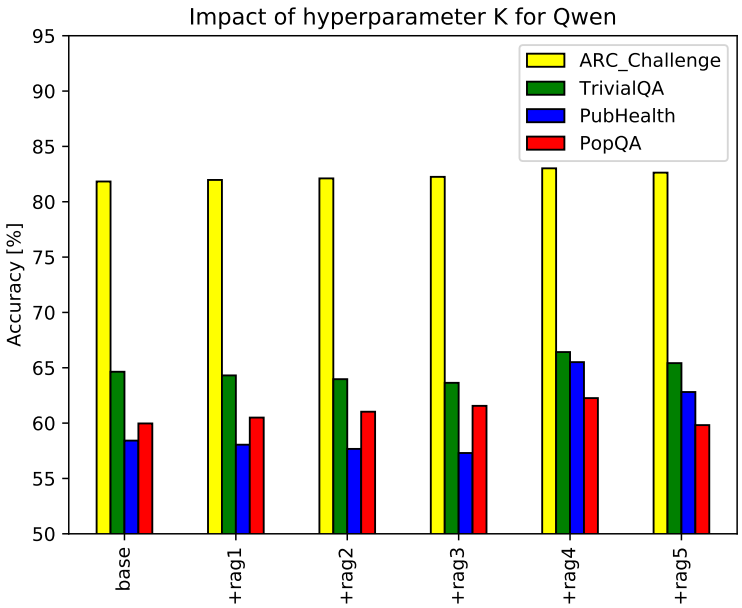

Figure 4. Impact of the number of reference passages *K* for Qwen.

4.4. Significance of Reflection Tags

The proposed framework incorporates four tags (RTV, RLV, SPT, and PIT) to regulate content retrieval and evaluation within the RAG paradigm. To assess the individual contribution of each tag, an ablation study was conducted. The GPT-3.5 model served as the baseline, with the dataset and evaluation metrics maintained from the previous experiment. Hyperparameter *K* was fixed to 2 from the results of previous experiments. Figure 5 presents the results. The horizontal axis categorizes experimental conditions by dataset, with each group comprising five bars: a full-mode baseline and four variants corresponding to the removal of each tag. Results indicate that the full-mode configuration consistently outperforms all other variants. Notably, the removal of RTV tags significantly

diminishes model performance, emphasizing the critical role of adaptive search in the proposed RAG framework.

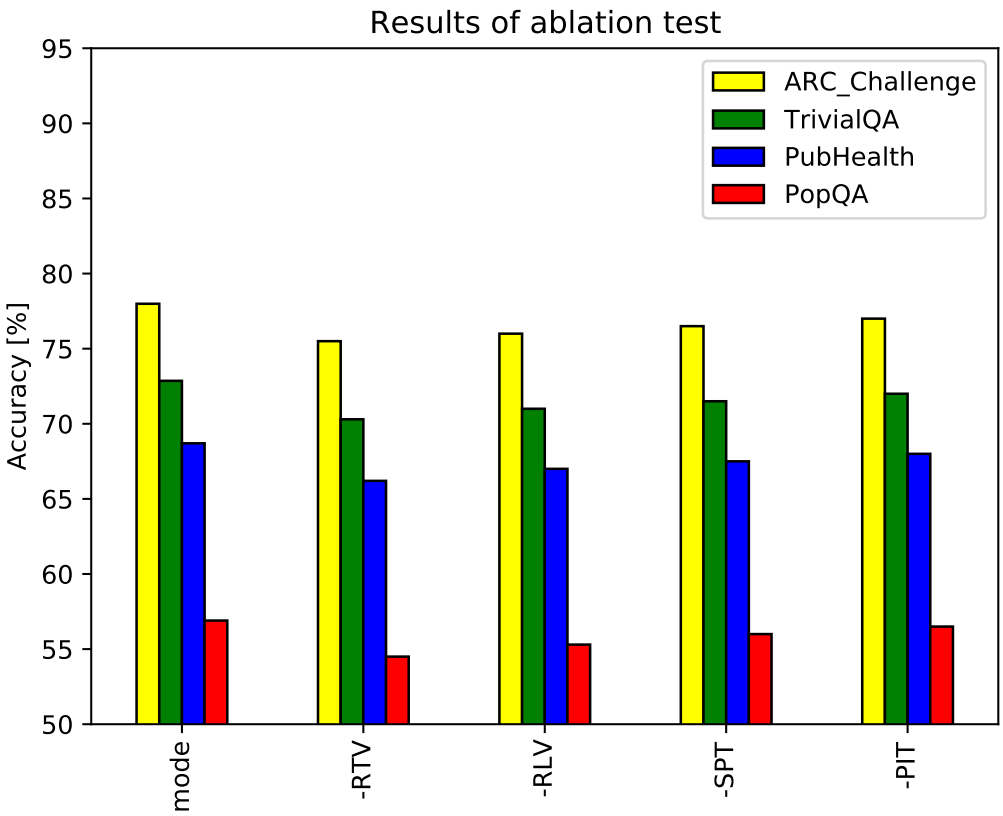

Figure 5. Results of ablation test.

4.5. Qualitative Analysis

Finally, to better understand the effectiveness of the proposed framework, we conducted a qualitative analysis of model responses to specific queries, with GPT-3.5 serving as the baseline. Table 3 presents sample outputs, where 'reference' denotes the correct answer, 'predict' represents the baseline model's response, and 'rag-result' signifies the proposed model's output. For the illustrated queries, the proposed framework consistently aligns with the reference answers, while the baseline model exhibits discrepancies. This indicates that the baseline LLM possessed insufficient knowledge to address these queries, whereas the integration of external knowledge via the proposed framework enabled accurate responses.

Table 3. Examples of queries for which the proposed method led to a correct answer.

(a) Content of the Queries.	
Id	**Question**
Mercury_7234308	A scientist maps a long region in which earthquakes originate and determines this region is a transform plate boundary. Which evidence would cause the scientist to re-evaluate this determination?
Mercury_184975	To determine how closely related organisms are, scientists consider all of the following.
Mercury_SC_400578	Which is an example of learned behavior?
AKDE&ED_2008_4_26	Which example shows a relationship between a living thing and a nonliving thing?

Table 3. *Cont.*

(b) Answers.			
Id	**Reference**	**Predict**	**Rag-Result**
Mercury_7234308	A	B	A
Mercury_184975	C	B	C
Mercury_SC_400578	A	C	A
AKDE&ED_2008_4_26	C	B	C

5. Concluding Remarks

This paper investigates retrieval-augmented generation (RAG) as a means to mitigate the hallucination and latency challenges inherent to large language models (LLMs). To address the limitations introduced by RAG, such as the potential masking of LLM capabilities, we propose a novel framework employing four reflective tags to control the retrieval and evaluation of external sources. A search tag enables adaptive search, mitigating the over-reliance on irrelevant information. Evaluation tags facilitate a comprehensive assessment of retrieved documents based on relevance, support, and overall quality. The framework incorporates chain-of-thought (CoT) reasoning to decompose queries and generate responses incrementally, further enhancing output quality and reliability.

To evaluate the performance of the proposed framework, we conducted experiments on four benchmark datasets: ARC-Challenge, PubHealth, PopQA, and TriviaQA, where GPT-3.5 and Qwen served as baselines, with accuracy as the primary evaluation metric. The experimental results shown in Section 4 demonstrate the effectiveness of the proposed method, especially in improving the accuracy of the fact-checking benchmarks. The performance of the proposed method depends on the value of the hyperparameter k, and the optimal value of k depends on the baseline LLM. These findings collectively provide compelling evidence of the effectiveness of the proposed framework.

Future work includes comparisons with existing methods for improving the performance of RAGs and evaluation experiments using a wider range of datasets. We plan to compare our method with existing robust approaches, including RAG by Facebook AI [18], REALM by Google [1], and various domain-specific retrieval systems such as PubMed [19], BioBERT [20] PatentBERT [21], and FinancialBERT [22]. Additionally, we will evaluate against the latest state-of-the-art LLMs, such as Phi-3.5 [23], LLaMA 3.2 [24], and OLMo [25]. Specifically, our investigation will focus on the influence of prepared inputs, such as keywords, on generation performance, particularly as it relates to solving retrieval problems. Furthermore, it is essential to evaluate our method using domain-specific datasets in areas such as law, medicine, and science.

Author Contributions: Conceptualization, C.Y.; methodology, C.Y. and S.F.; software, C.Y.; validation, S.F.; resources, C.Y.; data curation, C.Y.; writing—original draft preparation, C.Y.; visualization, S.F.; supervision, S.F.; project administration, S.F. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The code for the prototype system described in this paper, the code for the experiments conducted on it, and the raw data of the experimental results can be obtained from the following URL: https://github.com/Ysiennnnnn/yao (accessed on 10 November 2024).

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Context Construction Through Chain of Thought (CoT)

To make this paper self-contained, an overview of the chain-of-thought (CoT) technology used in the content generation step and its advantages is summarized.

Appendix A.1. Overview

In recent years, CoT technology, which emulates the human problem-solving process, has garnered significant attention. CoT has been pioneered by Wei et al. [26] and Kojima et al. [27] and has been effectively applied in various contexts, including multimodal reasoning [28], multilingual scenarios [29], and knowledge-driven applications [30]. CoT was developed to enhance the reasoning capabilities of generative models, with its core technology centered around step-by-step reasoning and generation. When confronted with a complex problem, a model utilizing CoT does not attempt to provide the final answer directly. Instead, it addresses the problem incrementally through a series of logical steps. Each step corresponds to an independent reasoning process, with the model generating the reasoning for the subsequent step based on the results of the previous step. Such a step-by-step approach to reasoning ensures that answers to complex questions are systematic, organized, and evidence-based.

Appendix A.2. Advantages of CoT

The advantages of CoT are threefold. First, CoT significantly enhances the reasoning capabilities of LLMs in complex reasoning tasks. Conventional generative models are prone to errors, particularly when handling intricate tasks, due to the presence of step jumps and logical breaks. In contrast, CoT mitigates such errors and improves problem-solving accuracy by progressively refining the task so that each incremental step remains within the model's comprehension.

Second, CoT improves the logical coherence of LLM-generated answers. The step-by-step reasoning process allows the model to maintain logical relationships and consistency as it generates partial answers at each step. Crucially, the final answer is derived from multiple inferences and repeated validations, resulting in higher reliability and rigor.

Finally, CoT enhances the transparency and accountability of the LLM reasoning process. Traditional generative models are often perceived as 'black boxes', making it difficult for users to understand how the model arrives at its answers. CoT, on the other hand, provides a clear, step-by-step reasoning process that allows users to understand the model's thinking and logic. The intermediate results of each step and the process of generating the final answer are open and transparent, facilitating user understanding and verification.

References

1. Guu, K.; Lee, K.; Tung, Z.; Pasupat, P.; Chang, M. Retrieval augmented language model pre-training. In Proceedings of the International Conference on Machine Learning, PMLR, Virtual Event, 13–18 July 2020; pp. 3929–3938.
2. 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. *Adv. Neural Inf. Process. Syst.* **2020**, *33*, 9459–9474.
3. De Curtò, J.; de Zarzà, I.; Roig, G.; Calafate, C.T. Large Language Model-Informed X-Ray Photoelectron Spectroscopy Data Analysis. *Signals* **2024**, *5*, 181–201. [CrossRef]
4. Li, Y.; Wang, S.; Ding, H.; Chen, H. Large language models in finance: A survey. In Proceedings of the Fourth ACM International Conference on AI in Finance, Brooklyn, NY, USA, 27–29 November 2023; pp. 374–382.
5. Chen, J.; Lin, H.; Han, X.; Sun, L. Benchmarking large language models in retrieval-augmented generation. In Proceedings of the AAAI Conference on Artificial Intelligence, Vancouver, BC, Canada, 20–27 February 2024; Volume 38, pp. 17754–17762.
6. Zeng, A.; Xu, B.; Wang, B.; Zhang, C.; Yin, D.; Zhang, D.; Rojas, D.; Feng, G.; Zhao, H.; Wang, Z. ChatGLM: A Family of Large Language Models from GLM-130B to GLM-4 All Tools. *arXiv* **2024**, arXiv:2406.12793.
7. Chiang, W.; Li, Z.; Lin, Z.; Sheng, Y.; Wu, Z.; Zhang, H.; Zheng, L.; Zhuang, S.; Zhuang, Y.; Gonzalez, J.E.; et al. Vicuna: An Open-Source Chatbot Impressing GPT-4 with 90%* ChatGPT Quality. 2023. Available online: https://lmsys.org/blog/2023-03-30-vicuna/ (accessed on 10 November 2024).
8. Bai, J.; Bai, S.; Chu, Y.; Cui, Z.; Dang, K.; Deng, X.; Fan, Y.; Ge, W.; Han, Y.; Huang, F.; et al. Qwen technical report. *arXiv* **2023**, arXiv:2309.16609.
9. Ji, Y.; Deng, Y.; Gong, Y.; Peng, Y.; Niu, Q.; Ma, B.; Li, X. BELLE: Bloom-Enhanced Large Language Model Engine, GitHub, GitHub Repository. 2023. Available online: https://github.com/LianjiaTech/BELLE (accessed on 10 November 2024).
10. Creswell, A.; Shanahan, M.; Higgins, I. Selection-inference: Exploiting large language models for interpretable logical reasoning. *arXiv* **2022**, arXiv:2205.09712.

11. Yoran, O.; Wolfson, T.; Ram, O.; Berant, J. Making retrieval-augmented language models robust to irrelevant context. *arXiv* **2023**, arXiv:2310.01558.
12. Ram, O.; Levine, Y.; Dalmedigos, I.; Muhlgay, D.; Shashua, A.; Leyton-Brown, K.; Shoham, Y. In-context retrieval-augmented language models. *Trans. Assoc. Comput. Linguist.* **2023**, *11*, 1316–1331. [CrossRef]
13. Izacard, G.; Lewis, P.; Lomeli, M.; Hosseini, L.; Petroni, F.; Schick, T.; Dwivedi-Yu, J.; Joulin, A.; Riedel, S.; Grave, E. Atlas: Few-shot learning with retrieval augmented language models. *J. Mach. Learn. Res.* **2023**, *24*, 1–43.
14. Luo, H.; Chuang, Y.S.; Gong, Y.; Zhang, T.; Kim, Y.; Wu, X.; Fox, D.; Meng, H.; Glass, J. Sail: Search-augmented instruction learning. *arXiv* **2023**, arXiv:2305.15225.
15. Mallen, A.; Asai, A.; Zhong, V.; Das, R.; Khashabi, D.; Hajishirzi, H. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. *arXiv* **2022**, arXiv:2212.10511.
16. Jiang, Z.; Xu, F.F.; Gao, L.; Sun, Z.; Liu, Q.; Dwivedi-Yu, J.; Yang, Y.; Jamie, C.J.; Neubig, G. Active retrieval augmented generation. *arXiv* **2023**, arXiv:2305.06983.
17. Oh, C.; Lim, H.; Kim, M.; Han, D.; Yun, S.; Choo, J.; Hauptmann, A.; Cheng, Z.-Q.; Song, K. Towards calibrated robust fine-tuning of vision-language models. *arXiv* **2023**, arXiv:2311.01723.
18. Gupta, S.; Ranjan, R.; Singh, S.N. A Comprehensive Survey of Retrieval-Augmented Generation (RAG): Evolution, Current Landscape and Future Directions. *arXiv* **2024**, arXiv:2410.12837.
19. White, J. PubMed 2.0. *Med. Ref. Serv. Q.* **2020**, *39*, 382–387. [CrossRef]
20. Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C.H.; Kang, J. BioBERT: A pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **2020**, *36*, 1234–1240. [CrossRef]
21. Lee, J.S.; Hsiang, J. Patentbert: Patent classification with fine-tuning a pre-trained bert model. *arXiv* **2019**, arXiv:1906.02124.
22. Hazourli, A. Financialbert—A Pretrained Language Model for Financial Text Mining. Research Gate. 2022. Available online: https://www.researchgate.net/publication/358284785_FinancialBERT_-_A_Pretrained_Language_Model_for_Financial_Text_Mining (accessed on 10 November 2024).
23. Abdin, M.; Jacobs, S.A.; Awan, A.A.; Aneja, J.; Awadallah, A.; Awadalla, H.; Bach, N.; Bahree, A.; Bakhtiari, A.; Behl, H.; et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv* **2024**, arXiv:2404.14219.
24. Llama 3.2: Revolutionizing Edge AI and Vision with Open, Customizable Models. Available online: https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/ (accessed on 10 November 2024).
25. Groeneveld, D.; Beltagy, I.; Walsh, P.; Bhagia, A.; Kinney, R.; Tafjord, O.; Jha, A.H.; Ivison, H.; Magnusson, I.; Wang, Y. Olmo: Accelerating the science of language models. *arXiv* **2024**, arXiv:2402.00838.
26. Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Xia, F.; Le, Q.; Zhou, D. Chain-of-thought prompting elicits reasoning in large language models. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 24824–24837.
27. Kojima, T.; Gu, S.S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large language models are zero-shot reasoners. *Adv. Neural Inf. Process. Syst.* **2022**, *35*, 22199–22213.
28. Zhang, Z.; Zhang, A.; Li, M.; Zhao, H.; Karypis, G.; Smola, A. Multimodal chain-of-thought reasoning in language models. *arXiv* **2023**, arXiv:2302.00923.
29. Shi, F.; Chen, X.; Misra, K.; Scales, N.; Dohan, D.; Chi, E.H.; Schärli, N.; Zhou, D. Large language models can be easily distracted by irrelevant context. In Proceedings of the International Conference on Machine Learning, PMLR, Honolulu, HI, USA, 23–29 July 2023; pp. 31210–31227.
30. Wang, K.; Duan, F.; Wang, S.; Li, P.; Xian, Y.; Yin, C.; Rong, W.; Xiong, Z. Knowledge-driven cot: Exploring faithful reasoning in llms for knowledge-intensive question answering. *arXiv* **2023**, arXiv:2308.13259.