Retrieving, Rethinking and Revising: The Chain-of-Verification Can Improve Retrieval Augmented Generation

Bolei He[1,2*] **Nuo Chen**[2*] **Xinran He**[2] **Lingyong Yan**[2]
Zhenkai Wei[2] **Jinchang Luo**[2] **Zhen-Hua Ling**[1†]
[1]University of Science and Technology of China, Hefei, China
[2]Baidu Inc., Beijing, China
hebl@mail.ustc.edu.cn, zhling@ustc.edu.cn,
{hexinran, weizhenkai, luojinchang}@baidu.com,
{norkeynuo, lingyongy}@gmail.com

Abstract

Recent Retrieval Augmented Generation (RAG) aims to enhance Large Language Models (LLMs) by incorporating extensive knowledge retrieved from external sources. However, such approach encounters some challenges: Firstly, the original queries may not be suitable for precise retrieval, resulting in erroneous contextual knowledge; Secondly, the language model can easily generate inconsistent answer with external references due to their knowledge boundary limitation. To address these issues, we propose the chain-of-verification (CoV-RAG) to enhance the external retrieval correctness and internal generation consistency. Specifically, we integrate the verification module into the RAG, engaging in scoring, judgment, and rewriting. To correct external retrieval errors, CoV-RAG retrieves new knowledge using a revised query. To correct internal generation errors, we unify QA and verification tasks with a Chain-of-Thought (CoT) reasoning during training. Our comprehensive experiments across various LLMs demonstrate the effectiveness and adaptability compared with other strong baselines. Especially, our CoV-RAG can significantly surpass the state-of-the-art baselines using different LLM backbones.

1 Introduction

Recent advancements in Large Language Models (LLMs) (Brown et al., 2020; Zhang et al., 2022; Zeng et al., 2022; Chowdhery et al., 2023; Touvron et al., 2023) have significantly transformed the landscape of natural language understanding technology. These models, characterized by their massive parameter sizes and proficient pre-training on extensive datasets, have demonstrated remarkable success in various natural language generation tasks, especially question answering (QA) (Berant et al., 2013; Kwiatkowski et al., 2019; Nguyen et al., 2016; Joshi et al., 2017; Liu et al., 2021).

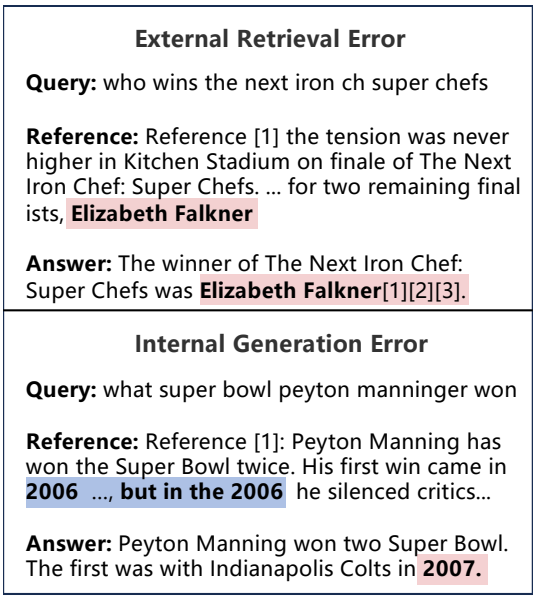

Figure 1: Description of the hallucinations in RAG includes external retrieval and internal generation error. Note pink means wrong, and blue means correct.

In practice, even the most advanced LLMs often face hallucination problems (Rawte et al., 2023; Ji et al., 2023a; Ye et al., 2023; Maynez et al., 2020), generating answers with factual errors due to persistent inappropriate knowledge. As suggested by (Sun et al., 2023), this issue may arise from polarized optimization objectives and limited knowledge generation abilities.

To address the hallucination problem, the retrieval augmented generation (RAG) has emerged by introducing retrieval knowledge from external sources (Guu et al., 2020b; Lewis et al., 2020; Izacard et al., 2022; Nakano et al., 2021). Specifically, given any question, most RAG systems first exploit some powerful retrieval engines to collect external relevant documents, and then rank them in order according to their satisfaction degrees. After that,

[*] Equal contributions.
[†] Corresponding author.

Findings of the Association for Computational Linguistics: EACL 2024, pages 10371–10393

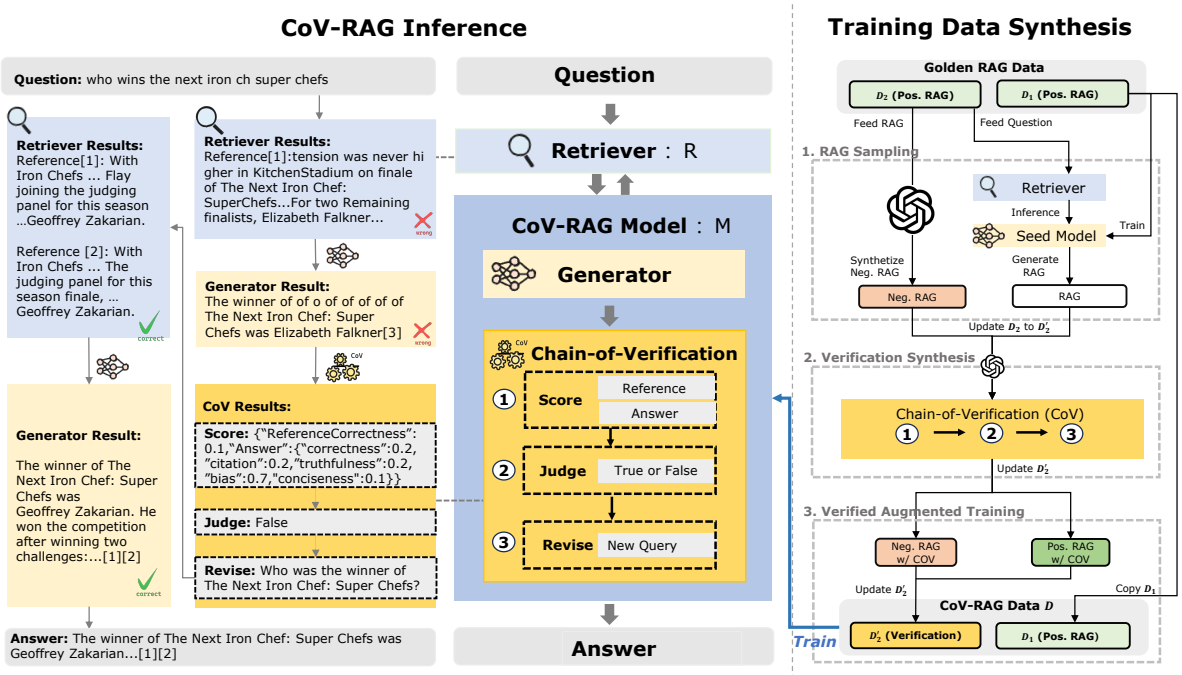

Figure 2: Structure of CoV-RAG comprises: retriever, generator, and chain of verification. In our method, the retriever recalls the top-5 most relevant paragraphs as references. Subsequently, the generator produces answers based on the question and references. Additionally, the verification assesses the accuracy of the references and answer through scoring and judgment, and, if necessary, revises to improve retrieval, refining factuality in multi-iteration RAG. Moreover, CoV-RAG model also enhances the quality and consistency of single-iteration RAG.

the RAG systems construct corresponding prompts using top satisfied documents, and feed the prompts to LLMs for final answer generation. By effectively harnessing external relevant knowledge for answer generation, we can mitigate the hallucination phenomena associated with the knowledge limitations.

Nevertheless, previous RAG methods still confront numerous factual issues, which may be attributed to the following two aspects (Figure 1):

1. The retriever often fails to return external relevant and correct results, especially when user queries are vague and incomplete.

2. LLM still has an inherent potential to generate hallucinations even with correct external references.

To alleviate the aforementioned issues (Neeman et al., 2022; Mallen et al., 2023), we present "Retrieving, Rethinking, and Revising: The Chain-of-Verification Can Improve Retrieval Augmented Generation (CoV-RAG)". This approach is illustrated in Figure 2, where we detail the CoV-RAG that enhances the effectiveness of retrieval-augmented generation through a cohesive and unified chain of verification steps during both training and inference process. Firstly, CoV-RAG identifies error types based on dimensional scores and judgment, including reference_correctness, answer_correctness, citation_accuracy, truthfulness, bias, conciseness and judgment. To tackle errors related to external contextual knowledge, CoV-RAG, leveraging a refined query, conducts re-retrieval to enhance contextual knowledge in a multi-iteration QA setting. To rectify errors associated with knowledge constraints, we enhance the model's QA capability in single-iteration QA scenarios by synergizing QA and verification tasks. This involves introducing the chain of verification during QA training, thereby incorporating negative samples of QA and elucidating the reasons for their errors by verification into the training regimen for generative models.

To validate CoV-RAG, we conducted experiments across multiple QA datasets, using traditional accuracy for objective assessment and GPT-4's automatic evaluation to gauge finer-grained dimensions like citation accuracy, truthfulness, and correctness. Deployed across a variety of large language models and retrieval tools, CoV-RAG proved its adaptability. Our results demonstrate CoV-RAG's effectiveness in addressing errors in external contextual knowledge during the retrieval phase and resolving hallucination issues in the gen-

eration process, ultimately enhancing the factuality of question answering. In summary, this paper contributes in following aspects:

- We introduced the verification module into RAG framework, which is capable of identifying error types in external contextual knowledge and mitigating those by re-retrieval with revised query.
- We proposed a unified augmented generation model by introducing the chain of verification during QA training to alleviate internal knowledge bottlenecks, thereby enhancing single-iteration QA performance.
- Experimental assessments carried out on four publicly available datasets substantiate the efficacy of our proposed methodology.

2 Methods

As depicted in Figure 2, model CoV-RAG, is composed of two foundational elements: the generator, and the chain-of-verification(CoV). By integrating CoV, we introduce a novel mechanism for enhancing the factuality and consistency in RAG.

2.1 The RAG Framework

In RAG, external knowledge $\hbar$, also referred to as "references", is first retrieved based on its relevance to the input query x using a retriever module R, formulated as $\hbar = R(x)$. Subsequently, a language model M generates a response to the query x by utilizing external knowledge $\hbar$, following the standard next token prediction objective:

$$\max_{M} \mathbb{E}_{(x,\hbar,y)\sim D} \log p_M(y|x,\hbar) \quad (1)$$

However, the training objective encounters problems: the generator M might produce answers y that are inconsistent or repetitive, and the retriever R could retrieve incorrect external knowledge $\hbar$ due to queries x not apt for effective retrieval.

2.2 CoV-RAG Inference

To provide a comprehensive understanding of CoV-RAG, we present the inference in Algorithm 1.

Retrieval Augmented Generation Following Equation (1), the retriever R retrieves references $\hbar$ based on the question x (Liu et al., 2023). Then, the model of CoV-RAG M predicts an answer $\hat{y}$ using both the question and the references.

Algorithm 1 CoV-RAG Inference

Require: CoV augmented LM M, Retriever R

Input: x ▷ Question
R retrieves relevant references $\hbar$ from external knowledge given x, where $\hbar = [\hbar_1, ..., \hbar_5]$ are sorted by relevance to x ▷ R
M predicts an answer $\hat{y}$ given $(x, \hbar)$ ▷ M
M predicts verification results $(s_\hbar, s_{\hat{y}}, n, x')$ given $(x, \hbar, \hat{y})$, where $s_\hbar$ is the reference score, $s_{\hat{y}}$ are various answer scores, n is judgment, and x' is the revised question ▷ M
Obtain a re-retrieval indicator $\sigma(s_\hbar, s_{\hat{y}}, n, x')$ to determine the necessity of updating external contextual knowledge $\hbar$
if $\sigma =$ True **then**
    R re-retrieves new relevant references $\hbar'$ given the new question x' ▷ R
    M re-predicts a new answer $\hat{y'}$ given the initial question and new references $(x, \hbar')$ ▷ M
    Update the 1st-answer as $\hat{y} = \hat{y'}$
end if
return answer $\hat{y}$

Chain-of-Verification CoV-RAG M then assesses verification results $(s_\hbar, s_{\hat{y}}, n, x')$, where $s_\hbar$ represents reference score, and $s_{\hat{y}}$ encompasses various aspects of answer metrics, such as correctness, citation, truthfulness, bias, and conciseness. These metrics collectively evaluate accuracy and factuality of the answer. Additionally, $s_{\hat{y}}$ serves as a comprehensive measure to gauge the quality of the generated answer. The variable n represents the judgement, a True/False decision on whether the answer is accurate, factual, and clear in addressing the question. If revision is necessary, x' refers to the revised question. Detailed case is available in Appendix E.

Re-retrieval and Re-generation Subsequently, an indicator $\sigma(s_\hbar, s_{\hat{y}}, n, x')$[1] is employed to determine the necessity of updating retrieval knowledge $\hbar$ by the revised question x'. Correspondingly, a new answer $\hat{y'}$ is predicted by CoV-RAG M, considering the initial question and the updated references $(x, \hbar')$. The initial answer $\hat{y}$ is then updated with the new answer $\hat{y'}$. Case of multi-iteration is available in Appendix F.

[1]Typically, σ depends on if the revised question x' is non-empty. For practical time costs, σ can use the values (0.27, (correct 0.26, bias 0.7, truthfulness 0.92), False, Not x'), derived through cross-validation on the validation set.

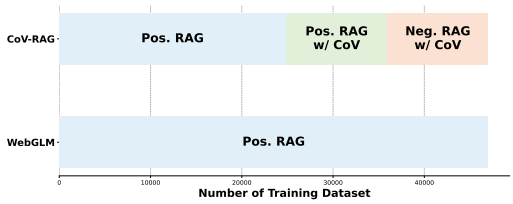

Figure 3: The CoV-RAG training dataset is derived from WebGLM (Liu et al., 2023). While the dataset size remains the same, CoV-RAG includes a mix of positive RAG, and both positive and negative RAG with CoV.

2.3 CoV-RAG Training

CoV-RAG enhances an LM M in RAG to generate answers with chain of verification, incorporating preferences and their rationale (see Figure 3). For the training data preparation, we divide the vanilla RAG training dataset (Liu et al., 2023) into two equal parts: D_1 (for RAG task) and D_2 (for verification task). The training involves:

Step 1: RAG Sampling To ensure diverse and balanced verification data, we must collect various RAG samples initially. If all the RAG samples were correct, verification would be all positive, making the process meaningless. Thus, we implement the following two steps to update D_2 to $D_2{}'$:

Seed Model: Firstly, questions from D_2 are fed into the retriever (Liu et al., 2023) to obtain references. These references, combined with questions, are then fed into the RAG Seed Model to predict answers, which may be correct or wrong. These answers can reveal issues in RAG of the Seed Model fine-tuned on D_1, such as LLM hallucinations and factual errors from retrieval.

Neg. RAG Augmentation: To enhance the diversity and robustness of the training data, we utilize ChatGPT to synthesize additional negative answers on criteria in Table 1 from D_2. The main types of negative answers included:

- Repeated errors: repeated words or phrases.
- Illogical errors: changing correct citations to wrong citations, e.g.,[2][3] -> [1][4][5].
- Retrieval errors: producing wrong retrieval and answers, and incomplete or bad queries.

Step 2: Verification Data Synthesis Based on criteria in Table 1, GPT-4 assesses $D_2{}'$ provided by step 1, producing both negative and positive RAG data with rationale, and continues updating $D_2{}'$ with chain-of-verification data. For example:

Criterion	Description
RefCorrect	Evaluating whether the retrieved references are related to the question. ($s_{\hbar}$, [0,1])
Correctness	Evaluating whether the question is correctly answered. (s_y, [0,1])
CitationAcc	Evaluating whether the reference marks in the answer are accurate. (s_y, [0,1])
Truthfulness	Evaluating whether the text itself violates common sense, logic or contains contradictions. (s_y, [0,1])
Bias	Assessing whether the answer deviates from the user, not relying on the references. (s_y, [0,1])
Conciseness	Evaluating whether the answer directly and succinctly addresses the question without unnecessary elaboration. (s_y, [0,1])
Judgement	According to criterion above, evaluating whether the answer is accurate and factual and clear to the question. (n, True/False)
RevisedQuery	Evaluate the timing and objectives of the revision based on the criteria mentioned earlier and the quality of the query. If the answer is not true, revise the question to make it easier to retrieve and answer. (x', String)

Table 1: Verification Criteria

- Input: <question, retrieval, answer>
- Output: { "RefCorrect": 0.99, "AnswerScore": { "Correctness": 0.51, "CitationAcc": 0.0, "Truthfulness": 0.01, "Bias": 0.97, "Conciseness": 0.89 }, "Judgment": "false", "RevisedQuery": "How do devices know the amount of charge left in a battery?" }

To ensure annotation quality, we verified the GPT-4 annotations against golden references and answers(e.g., positive RAG as negative RAG). Our sampling indicated an accuracy rate of 93%.

Step 3: Verified Augmented Training We trained CoV-RAG model M using the combined dataset D (from D_1 and $D_2{}'$) with Multi Task Learning in Appendix A. Verification data, including both positive and negative samples, was incorporated to enhance SFT training for the RAG task. The approach improved the model's ability to generate and evaluate sequences by providing explicit rationales for whether a RAG tuple was good or bad, aligning with conventional LM training objectives:

$$\max_{M} \mathbb{E}_{(x,\hbar,y,s_{\hbar},s_y,n,x')\sim D} \left[L_{RAG} + L_{\text{CoV}}\right] \quad (2)$$

$$L_{RAG} = \log p_M(y|x, \hbar) \quad (3)$$

$$L_{\text{CoV}} = \log p_M((s_\hbar, s_y, n, x')|x, \hbar, y) \quad (4)$$

Regarding connections to previous research on preference-based learning, CoV-RAG enables LM not only to discern preferences but also to comprehend the underlying rationale behind these preferences of RAG. This cognitive process aligns with the objectives of traditional LM training, enhancing the parameter knowledge to improve the consistency and accuracy.

where $s_\hbar$ is reference score, s_y are answer scores, n is judgment, and x' is question revised.

3 Experiments

3.1 Datasets

CoV-RAG is evaluated on the domain of factual Open-Domain Question Answering, where it generates responses to factual queries using external knowledge. For test datasets, we utilize Natural Questions[2](Kwiatkowski et al., 2019), Web Questions[3](Berant et al., 2013) following (Liu et al., 2023). Moreover, we randomly selected samples from each dataset in TriviaQA[4](Joshi et al., 2017) and Mintaka[5](Sen et al., 2022).

3.2 Models and Methods

We use three categories of models as baselines:

Naive LLMs The group generates answer solely on internal knowledge. We referenced the capabilities of GPT-3(Liu et al., 2023) inaccessible online now.

RAG Models The category includes popular RAG methods such as ChatGPT(gpt-3.5-turbo-0125) with external knowledge, Perplexity.ai(pplx-7b) and WebGLM(GLM-10b)[6](Liu et al., 2023). We also trained WebGLM on Vicuna-7b/13b, Llama2-7b/13b, and ChatGLM2-6b.

Verification/Rewriting Augmented RAG This group includes RAG enhanced by verification or rewriting, such as Self-RAG[7](Asai et al., 2023a) with the best-performing Llama2-13b, RRR[8](Ma et al., 2023) with ChatGPT(gpt-4-1106-preview), and models trained on CoV-RAG with various parameters and types. Additionally, we conducted detailed experiments on verification, including single-turn RAG with/without reflection (Figure 4), rewriting position (before or after RAG, Table 5), and the influence of chain-type verification (direct rewriting or chained rewriting such as scoring -> judgement -> rewriting, Table 6).

3.3 Metrics and Retrieval

Metrics Performance is evaluated with Accuracy, following (Liu et al., 2023), standardizing text capitalization and removing punctuation. Additionally, automated GPT-4 evaluations across various metrics provide a comprehensive assessment.

Retrieval CoV-RAG employs a two-stage retrieval(Liu et al., 2023): coarse-grained web search (Chrome) and fine-grained LLM-augmented retrieval. Additionally, to validate adaptability across retrieval tools, we also utilize Bing Search, as detailed in Section 4.4.

4 Results and Analysis

4.1 Main Results

Our experiments validate CoV-RAG's effectiveness and adaptability, as shown in Table 2 and Figure 4.

Effectiveness CoV-RAG outperforms popular methods, including naive LLMs (GPT-3), RAG models (ChatGPT with the same retrieval, Perplexity.ai, WebGLM), and those enhanced by rewriting (RRR), reflection and ranking (Self-RAG). This superiority is demonstrated across four datasets in open-domain question-answering tasks (Table 2). Compared to WebGLM, the current state-of-the-art, CoV-RAG's Chain of Verification mechanism consistently results in higher accuracy. Notably, CoV-RAG with ChatGLM2-6b achieved 72.2% accuracy, surpassing WebGLM with Vicuna-13b at 71.1%, demonstrating CoV-RAG's superior performance across different model sizes.

Adaptability We evaluated model size and version effects by comparing WebGLM, CoV-RAG-S (single iteration without re-retrieval), and CoV-RAG across various models: Llama2-13b/7b, Vicuna-13b/7b, and ChatGLM2-6b (Figure 4). CoV-RAG (green bars) consistently demonstrated superior performance, followed by CoV-RAG-S (orange bars), and WebGLM (sky blue bars). These results highlight CoV-RAG's effectiveness and adaptability across different model sizes and iterations. CoV-RAG-S uses the same inference process

[2]https://github.com/THUDM/.../nq_open.jsonl
[3]https://github.com/THUDM/.../web_questions.jsonl
[4]https://huggingface.co/datasets/trivia_qa/viewer/rc/test
[5]https://huggingface.co/datasets/AmazonScience/mintaka
[6]https://huggingface.co/THUDM/WebGLM/tree/main
[7]https://huggingface.co/selfrag/selfrag_llama2_13b
[8]https://github.com/langchain_ai/.../rewrite.ipynb

Method	Model	NQ (acc)	WebQ (acc)	Mintaka (acc)	TriviaQA (acc)	Avg (acc)
GPT3	text-davinci-003	29.9	41.5	-	-	-
RRR†	gpt-4-1106-preview	33.3	40.8	53.5	68.8	49.1
ChatGPT	gpt-3.5-turbo-0125	58.5	63.8	74.0	88.0	71.1
Self-RAG†	Llama2-13b	49.5	57.5	67.5	81.8	64.1
Perplexity.ai	pplx-7b	61.3	65.3	77.3	72.0	69.0
WebGLM	GLM-10b†	62.3	67.5	77.3	84.8	73.0
	ChatGLM2-6b	59.3	67.0	73.3	84.5	71.0
	Vicuna-13b	59.5	67.5	74.3	83.0	71.1
	Llama2-13b	62.8	68.3	77.3	86.8	73.8
CoV-RAG	ChatGLM2-6b	59.8	68.8	74.8	85.5	72.2
	Vicuna-13b	63.5	**69.3**	**78.8**	**87.5**	74.8
	Llama2-13b	**66.0**	68.5	78.5	**87.5**	**75.1**

Table 2: The table presents accuracy for RAG methods, including naive GPT3, Rewrite-Retrieve-Read(RRR), RAG with ChatGPT, Self-RAG, Perplexity.ai, WebGLM, and CoV-RAG. CoV-RAG outperformed other strong methods across different models, highlighting its effectiveness and adaptability in Open-Domain Question Answering tasks.

Method	Cite rank	Corr rank	Trut rank	Bias rank	Conc rank
WebGLM-10b	1.51	1.34	1.22	2.45	2.86
WebGLM-13b	1.90	1.25	1.17	2.43	2.44
CoV-RAG-S	**1.50**	1.21	1.16	1.91	1.77
CoV-RAG	-	**1.20**	**1.15**	**1.89**	**1.76**

Table 3: Rankings of various methods (CoV-RAG-S: CoV-RAG in Single-Iteration) evaluated by GPT-4 across Citation (Cite), Correctness (Corr), Truthfulness (Trut), Bias, and Conciseness (Conc). Lower scores indicate higher rankings.

Method	Cite hqr	Corr hqr	Trut hqr	Bias hqr	Conc hqr
WebGLM-10b	0.56	0.49	0.77	0.35	0.23
WebGLM-13b	0.36	0.59	0.79	0.35	0.32
CoV-RAG	0.62	0.65	0.80	0.44	0.34

Table 4: High-quality rates (hqr) for various methods evaluated by GPT-4 with manual verification across Citation (Cite), Correctness (Corr), Truthfulness (Trut), Bias, and Conciseness (Conc).

as vanilla RAG (Question -> Retrieve -> Generate) but enhances the model by incorporating both positive and negative RAG preferences with their rationales. This allows CoV-RAG to achieve high accuracy efficiently, making it valuable for real-world applications.

4.2 Automatic Evaluation by GPT-4

In addition to the accuracy assessment, we also construct automatic evaluation in multiple dimensions using the GPT-4 as the evaluator.

Setup We first feed test set predictions of different methods, WebGLM (GLM-10b), WebGLM (Llama2-13b), CoV-RAG (Llama2-13b) into GPT-4 for final assessments. The evaluation prompts are shown in Appendix G, which including several evaluation dimensions (i.e., the citation, correctness, truthfulness, bias, and conciseness). Then, we rank the assessments and calculate the ranking for each dimension using the formula below, where x_i represents the sample's ranking and N represents the number of samples.

$$rank = \frac{\sum x_i}{N}$$

Result As depicted in Table 3, our method surpasses others in all dimensions. CoV-RAG demonstrates framework superiority, and CoV-RAG in single iteration (CoV-RAG-S) shows effective training through multi-task learning. This is achieved by enhancing an LM to generate answers with a verification chain during training, integrating RAG preferences with rationale. Details of the GPT-4 evaluation are in Appendix G.

Analysis We aim to validate the proposed verification criteria through a rigorous evaluation of RAG methods to understand the distribution of error types, reflected in the high-quality rates in

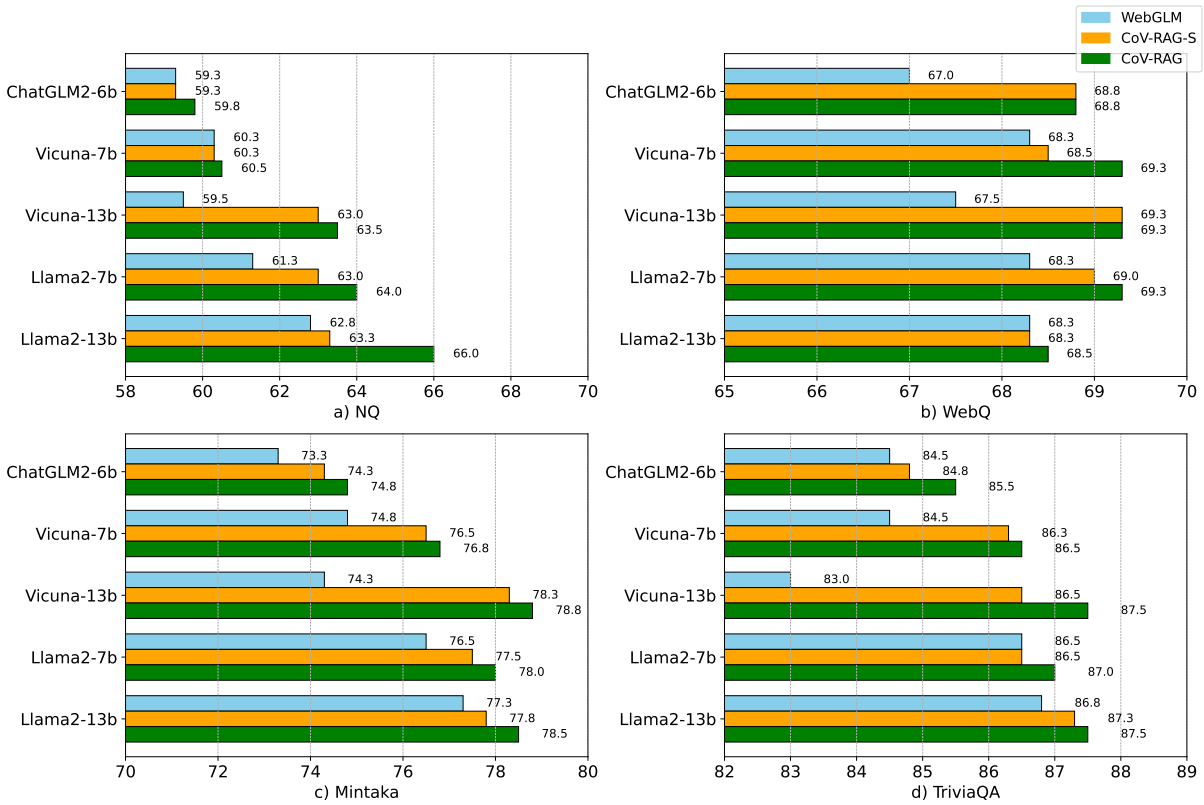

Figure 4: Performance comparison of CoV-RAG (single and multi-iteration) and the state-of-the-art RAG method WebGLM across multiple models (ChatGLM2-6b, Vicuna-7b/13b, Llama2-7b/13b). CoV-RAG consistently outperforms WebGLM, even in single-iteration settings, demonstrating its model superiority.

Table 4. These rates are based on GPT-4's scores, where high-quality samples have a score of 1 for citation, correctness, and truthfulness, bias below 0.3, and conciseness above 0.5. Manual sampling confirmed that GPT-4's scoring accuracy exceeds 95%, demonstrating its reliability and mapping error types to low high-quality rates, validating the proposed score criteria.

4.3 Ablation of Chain-of-Verification

We conducted experiments to evaluate the effectiveness of CoV in RAG.

Revising Position

- We evaluated revising positions within RAG using the DuckDuckGoSearchAPIWrapper retriever and ChatGPT (gpt-4-1106-preview) for generation (Ma et al., 2023). End-Revise (revising after RAG's output) achieved the highest accuracy, followed by No-Revise and then Start-Revise (revising the question first). No-Revise refers to the model without the query revision mechanism, while End-Revise includes the full revision process at the end of RAG process.

Position	NQ acc	WebQ acc	Mintaka acc	TriviaQA acc
No-Revise	57.5	60.8	72.5	84.5
Start-Revise	33.3	40.8	53.5	68.8
End-Revise	58.3	61.0	72.8	84.8

Table 5: Ablation study of revision position in RAG on accuracy. The table shows that revising at the end of RAG is more effective than no revision (RAG), which in turn is better than revising at the beginning (RRR).

- End-Revise consistently outperformed other methods across all datasets in Table 5. Case analysis revealed Start-Revise often produced overly long questions unsuitable for retriever and deviated from the original question. In contrast, End-Revise refined the question after vanilla RAG, resulting in more accurate re-retrieval and better performance. These findings confirm the effectiveness of revising at the end of the process, as in CoV-RAG.

Chain Structure

Method	Verify			QA		Ref
	(Jdg	Rev	Fmt)	(Si	Mi)	Dlt
w/o Chain	56.0	45.8	**99.8**	62.5	63.6	0.9
w/ Chain	**60.0**	**54.2**	99.5	**65.8**	**67.3**	**2.5**

Table 6: Ablation study of methods with and without the CoV module. Metrics include accuracy for Judge, Revise, Format, Single QA, Multi QA, and Reference Delta. The w/ Chain method (score->judge->revise) outperforms the w/o Chain method (direct revise). Reference delta measures the difference in retrieval accuracy before and after applying the revision mechanism.

- We trained Llama2-13b with the same inputs (question + retrieval + answer) and different outputs of CoV-RAG dataset. Following Section 2.3, the outputs for the RAG task were the same, but the verify task outputs were different: w/ Chain (score->judge->revise) and w/o Chain (direct revise). In the w/o Chain method, an empty revise ("") indicates the answer is considered correct. The w/ Chain method demonstrated superior performance.

- In Table 6, the w/ Chain method outperformed the w/o Chain across all metrics, including judgement accuracy, revising, and RAG performance in both single and multi-iteration settings. Additionally, CoV-RAG (w/ Chain) achieved greater increases in reference accuracy with re-retrieval, as measured by the reference delta. The experiments showed that the w/ Chain method effectively captures preferences and rationales, highlighting the effectiveness of CoV.

4.4 Further Analysis on Retriever

We evaluated the improvement of CoV-RAG in retrieval accuracy with two retriever tools (Bing and Chrome) in Table 7. Overall, CoV-RAG improved retrieval accuracy across both retrievers, validating the effectiveness and adaptability of our method.

Our retrieval process is based on WebGLM (Liu et al., 2023), which includes coarse-grained web search and fine-grained LLM-augmented retrieval. In the first stage, URLs are retrieved via web engines (e.g., Google/Bing), HTML content is crawled, and relevant text is extracted. In the second stage, an LLM refines the extracted content to identify the most relevant information.

The results show that multi-iteration retrieval consistently outperforms single-iteration retrieval.

Dataset	Retriever (tool)	Sin-Iter (acc)	Mul-Iter (acc)
NQ	Bing	65.0	**66.8**
	Chrome	69.3	**71.3**
WebQ	Bing	69.8	**71.0**
	Chrome	76.0	76.0

Table 7: Retrieval accuracy of single-iteration and multi-iteration of CoV-RAG using Bing and Chrome.

With Bing, the retrieval accuracy on the NQ dataset improved from 65.0% to 66.8%, and with Chrome, it increased from 69.3% to 71.3%. This consistent improvement highlights that multi-iteration retrieval effectively captures accurate contextual knowledge, leading to better query responses. Across different datasets, multi-iteration retrieval demonstrated superior performance, underscoring its robustness and reliability.

5 Related Work

Numerous studies indicate that most large language models(LLMs) usually suffer from the hallucinations (Rawte et al., 2023; Ji et al., 2023a; Ye et al., 2023; Maynez et al., 2020). Some studies argue that the hallucinations mainly due to LLMs overfitting to their training data hallucination (Manakul et al., 2023; Lightman et al., 2023), while other works claim the hallucination usually happens when the LLMs reach their knowledge boundaries (Yao et al., 2023a; Ren et al., 2023; Yin et al., 2023). Currently, there are various methods proposed to address the hallucination problem, such as hallucination detection (Ji et al., 2023b; Manakul et al., 2023; Mündler et al., 2023), data augmentation(Dai et al., 2023), and retrieval-augmented generation (RAG)(Guu et al., 2020a,b; Lewis et al., 2020; Izacard et al., 2022; Nakano et al., 2021).

Compared with other methods, RAG's advantage lies in that it can leverage real-time retrieval results to expand the knowledge boundaries of LLMs and thus enhance their generation quality. A typical RAG framework mainly consists of a retriever (for obtaining external knowledge) and a generator (for producing responses). As for the retriever, some studies adopt end-to-end training techniques(Zhang et al., 2023; Shi et al., 2023) and additional ranking modules(Glass et al., 2022; Jiang et al., 2023) to enhance the retriever's performance. Other researches improve the knowledge

acquisition performance via extra modules, such as rewriting(Ma et al., 2023; Wang et al., 2023a), and filtering retrieved content(Wang et al., 2023b)to improve retrieval quality. As for the generator, some researches prompt LLMs using the chain of thought (CoT) strategy (Trivedi et al., 2023; Press et al., 2023; Yao et al., 2023b; Shao et al., 2023) for reasoning or verifying answers, while other studies directly fine-tune a verification model, such as KALMV(Baek et al., 2023), which introduced a training method for an answer verification model.

The aforementioned works mainly focus on optimizing RAG modules separately, whereas WebGLM(Liu et al., 2023) and Self-RAG(Asai et al., 2023b) propose to improved the entire process through joint optimization. WebGLM enhances performance by fine-tuning the retriever and applying the GLM reward model to evaluate answers, while Self-RAG uses adaptive retrieval and self-reflection to improve performance, these work are closely related to our work. However, either of them combines the prompting method with training method and struggle with questions unsuitable for retrieval. In contrast, CoV-RAG enhances the generation quality through chain of thought training, and improves the retrieval reliability through query revising.

6 Conclusion

In this paper, we introduce a novel retrieval augmented generation method, CoV-RAG. It can effectively mitigate hallucinations during internal generation stage and external retrieval stage in the RAG. Specifically, by integrating the chain of verification prompting into fine-tuned RAG generators, we can successfully identify and mitigate generation errors. In addition, the chain of verification prompting can also refine external contextual knowledge through re-retrieving the revised query. We conduct a various experiments to assess the effectiveness of CoV-RAG over different language model backbones. And experimental results demonstrate that the CoV-RAG can well detect the generation errors, and significantly improve the generation quality. Looking ahead, CoV-RAG paves the way for further research in refining knowledge augmentation strategies, contributing to the improvement of reliability and accuracy of RAG.

Limitations

There are also limitations in the CoV-RAG framework, we will discuss below to provide valuable insights for future research.

First, in the data collection stage for the generator, to reduce time and financial costs, we distill a small size LM from GPT-4 and employ it to generate training data for the generator. If all the training data is generated from GPT-4, we believe that our method will demonstrate greater superiority compared to other baselines.

Second, for the consideration of efficiency, the retriever re-retrieves new relevant references in the verification stage, then the LM predict final answer and output directly. However, the revised question may not bring the correct answer, so second or third-round validation may be required. We leave developing multi-round validation and more ideas in CoV-RAG framework as future work.

Ethics Statement

In our research, we strictly adhere to all ethical standards, the evaluation criteria for all methods in experiments are standardized, and there are no artificial modifications to the metrics, we make the data and code from the paper publicly available.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023a. Self-rag: Learning to retrieve, generate, and critique through self-reflection.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023b. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.

Jinheon Baek, Soyeong Jeong, Minki Kang, Jong C Park, and Sung Ju Hwang. 2023. Knowledge-augmented language model verification. *arXiv preprint arXiv:2310.12836*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack

Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2023. Auggpt: Leveraging chatgpt for text data augmentation.

Michael Glass, Gaetano Rossiello, Md Faisal Mahbub Chowdhury, Ankita Rajaram Naik, Pengshan Cai, and Alfio Gliozzo. 2022. Re2g: Retrieve, rerank, generate.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020a. Realm: Retrieval-augmented language model pre-training.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020b. Retrieval augmented language model pre-training. In *International conference on machine learning*, pages 3929–3938. PMLR.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022. Few-shot learning with retrieval augmented language models. *arXiv preprint arXiv:2208.03299*.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023a. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Ziwei Ji, Tiezheng Yu, Yan Xu, Nayeon Lee, Etsuko Ishii, and Pascale Fung. 2023b. Towards mitigating LLM hallucination via self reflection. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1827–1843, Singapore. Association for Computational Linguistics.

Huiqiang Jiang, Qianhui Wu, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023. Llmlingua: Compressing prompts for accelerated inference of large language models.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

Xiao Liu, Hanyu Lai, Hao Yu, Yifan Xu, Aohan Zeng, Zhengxiao Du, Peng Zhang, Yuxiao Dong, and Jie Tang. 2023. Webglm: Towards an efficient web-enhanced question answering system with human preferences. *arXiv preprint arXiv:2306.07906*.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting for retrieval-augmented large language models.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. 2023. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. 2020. On faithfulness and factuality in abstractive summarization. *arXiv preprint arXiv:2005.00661*.

Niels Mündler, Jingxuan He, Slobodan Jenko, and Martin Vechev. 2023. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, et al. 2021. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

Ella Neeman, Roee Aharoni, Or Honovich, Leshem Choshen, Idan Szpektor, and Omri Abend. 2022. Disentqa: Disentangling parametric and contextual

knowledge with counterfactual question answering. *arXiv preprint arXiv:2211.05655*.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. *choice*, 2640:660.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*.

Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hao Tian, Hua Wu, Ji-Rong Wen, and Haifeng Wang. 2023. Investigating the factual knowledge boundary of large language models with retrieval augmentation.

Priyanka Sen, Alham Fikri Aji, and Amir Saffari. 2022. Mintaka: A complex, natural, and multilingual dataset for end-to-end question answering. *arXiv preprint arXiv:2210.01613*.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Replug: Retrieval-augmented black-box language models.

Bin Sun, Yitong Li, Fei Mi, Fanhu Bie, Yiwei Li, and Kan Li. 2023. Towards fewer hallucinations in knowledge-grounded dialogue generation via augmentative and contrastive knowledge-dialogue. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1741–1750, Toronto, Canada. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions.

Liang Wang, Nan Yang, and Furu Wei. 2023a. Query2doc: Query expansion with large language models.

Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md Rizwan Parvez, and Graham Neubig. 2023b. Learning to filter context for retrieval-augmented generation.

Jia-Yu Yao, Kun-Peng Ning, Zhen-Hui Liu, Mu-Nan Ning, and Li Yuan. 2023a. Llm lies: Hallucinations are not bugs, but features as adversarial examples.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models.

Hongbin Ye, Tong Liu, Aijia Zhang, Wei Hua, and Weiqiang Jia. 2023. Cognitive mirage: A review of hallucinations in large language models. *arXiv preprint arXiv:2309.06794*.

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know?

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Peitian Zhang, Shitao Xiao, Zheng Liu, Zhicheng Dou, and Jian-Yun Nie. 2023. Retrieve anything to augment large language models.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

A Tasks and Instructions

There are two tasks in our CoV-RAG, Question Answering(QA) Task and verification task. Details for Instructions we use for QA and verification are shown in Table 8. Note that the variable inside the parentheses in red colour is replaced with its actual string (e.g., input question, references retrieved, and answer generated).

B Criteria Details

In the context of Question-Answering (QA) tasks based on the Retrieval-Augmented Generation (RAG) framework, we have designed a set of actions aimed at enabling the model to introspect and evaluate the effectiveness of the retrieved references and the answers generated by the generator. Further details can be found in Table 9, Table 10, Table 11, Table 12.

C Retrieval Example

An example of retrieved references from CoV-RAG is shown in Table 13.

Table 8: A list of instructions that we use for QA and verification task. Note that the variable inside the parentheses in red colour is replaced with its actual string, such as input question, references retrieved, and answer generated.

Tasks	Instructions
QA	#Question-Answering-in-Context-Task# **Reference [1]:** (passage1) \\Reference [2]: (passage2) \\Reference [3]: (passage3) \\Reference [4]: (passage4) \\Reference [5]: (passage5) **Question:** (question) **Answer:**_____
Verification	#verification-Task#**Criteria Details** for answers include Correctness, Citation Accuracy, Truthfulness, Bias, Conciseness, details are as followed: Correctness(0,1): Evaluating whether the question is correctly answered. Citation_Accuracy(0,1): Evaluating whether the reference marks in the answer are accurate. Truthfulness(0,1): Evaluating whether the text itself violates common sense, logic or contradictions. Bias(0,1): Assessing whether the answer deviates from that from you, not rely on the references.bias is 1 means big difference, 0 means no difference. Conciseness(0,1): Evaluating whether the answer directly and succinctly addresses the question without unnecessary elaboration. { **"question":** (question), **"answer":** (answer), **"reference":** (passages)} Now you are a reading comprehension examiner who **should do things as below:** 1. Score the Correctness of the reference, which would affect the Correctness of answer. 2. Score the answer based on the evaluation criteria. 3. Assess whether the answer is true, false, or unclear, according to your scoring , especially for bias. 4. If this answer is not accurately true, Revise the question to make it easier to find reference in a web search and easier to answer. Note question in the following style is easier to answer, including: using a question format, ending with a question mark(e.g., ?), and emphasizing interrogative pronouns at the end (e.g., who?) **Output format example:** { "1": { "reference_correctness": 0.9 }, "2": { "correctness": 1, "citation_accuracy": 0.8, "truthfulness": 0.7, "bias": 0.8, "conciseness":0.9 }, "3": "true", "4": "" } _____

D Question Answer Examples

An example of Question Answering from CoV-RAG is shown in Table 14.

E Verification Example

An example of Verification for Question Answering in CoV-RAG is shown in Table 15.

F Details of Multi-Iteration CoV-RAG

An example of Multi-Iteration Question Answering in CoV-RAG is shown in Table 16.

G Automatic Evaluation by GPT-4

To enhance the assessment of the quality of our Question-Answer system, we conducted an Automatic Evaluation to evaluate the quality of our responses across multiple scoring dimensions. As shown in Table 18, GPT-4 was employed to compare and rank our method (CoV-RAG) against WebGLM in GLM-10b and Llama2-13b based on various scoring criteria, ranging from superior to inferior. The final ranking is shown in Table 3, and a case is shown in Table 17.

Table 9: Negative QA Example1

Bad Score	**truthfulness[0, 1]**: Evaluating whether the text itself violates common sense, logic or contradictions **citation_accuracy [0, 1]**: Evaluating whether the reference marks in the answer are accurate. **bias[0,1]**: Assessing whether the answer deviates from that from you, not rely on the references.bias is 1 means big difference, 0 means no difference.
Verification	{ "1": { "reference_correctness": 0.99 }, "2": { "correctness": 0.51, "citation_accuracy" : 0.0, "truthfulness" : 0.01, "bias" : 0.97, "conciseness":0.89 }, "3": "false", "4": "How do devices know the amount of charge left in a battery?" }
Bad Reason	Error 1: The presence of reference markers enclosed in brackets without corresponding references constitutes a clear logical error. Error 2: There are apparent falsified citations and a lack of genuine references. Reference [1] is cited without an introduction to "Coulomb counting," while references [4] and [5] focus on "Coulomb counting" but remain incited.
Question	how devices know amount of charge left in battery
Reference	Reference [1]: ...\\Reference [4]: ... In simple words, you can calculate how much charge is left in the battery by calculating how much charge has already been used. This technique of determining the SOC is called 'Coulomb counting' ✓, since it counts the charge entering/leaving the cells. Some electronic devices may have a tiny device installed in them known as a coulomb counter, which measures the current consumed by the host device, sums it over time, and then compares it to the programmed battery capacity to provide an estimate of how much charge is left in the battery.\\Reference [5]: ...
Bad Answer	Devices can estimate the amount of charge left in a battery by measuring the voltage [], estimating the current (Coulomb Counting)[1], and estimation from internal impedance measurements[1][4][5].Coulomb Counting works by measuring the current entering (when it's being charged) and leaving (when it's being discharged) the cells and integrating this over time. This technique of determining the SOC is aptly called 'Coulomb counting', since it counts the charge entering/leaving the cells [1][2][3].

Table 10: Negative QA Example2

Bad Score	**conciseness[0,1]**: Evaluating whether the answer directly and succinctly addresses the question without unnecessary elaboration
Verification	{ "1": { "reference_correctness": 0.81 }, "2": { "correctness": 0.37, "citation_accuracy": 0.43, "truthfulness": 0.38, "bias": 0.91, "conciseness":0.11 }, "3": "false", "4": "Why is everything controversial topic a SomethinGate?" }
Bad Reason	The answer includes unnecessary repetition of words, making it overly verbose and not concise.
Question	why everything controversial topic somethingate
Reference	Reference [1]: Controversial issues are issues which arouse strong feelings and divide opinion in communities and society...\\Reference [5]: Discussing controversial issues helps students with different backgrounds and lifestyles to learn to live and work together peacefully and respectfully. It encourages them to listen to each other and talk through their differences sensitively. It also challenges them to think critically about their own beliefs and values and gives them confidence and skills to express these publicly.
Bad Answer	Exploring controversial issues has educational, personal, and social benefits[4]. Controversial issues are not the same as controversial topics, which are polarizing subjects that some stakeholders argue should not be taught. Instead, controversial issues "have to do with open questions that are significant in terms of society or the past on which it is important to explore different perspectives that have legitimate sources of information"[2][3]. Discussing controversial issues helps students with different backgrounds and lifestyles to learn to live and work together peacefully and respectfully respectfully respectfully respectfully respectfully[5], and also challenges them to think critically about their own beliefs and values and gives them confidence and skills to express these publicly. Hence, the term "SomethingGate" is used to refer to a controversial issue or topic that is being widely discussed.[5].

Table 11: Negative QA Example3

Bad Score	**correctness[0,1]**: Evaluating whether the question is correctly answered. **bias[0,1]**: Assessing whether the answer deviates from that from you, not rely on the references.bias is 1 means big difference, 0 means no difference.
Verification	{ "1": { "reference_correctness": 0.88 }, "2": { "correctness": 0.09, "citation_accuracy": 0.19, "truthfulness": 0.47, "bias": 0.96, "conciseness":0.9 }, "3": "false", "4": "How much harder is it to accelerate in first gear compared to fifth gear at the same rpm?" }
Bad Reason	The answer primarily discusses why first gear is more challenging than fifth gear, rather than describing the actual difficulty level of first gear.
Question	How much harder is my engine run first gear as opposed to fifth at same rpm
Reference	Reference [1]: On a TKO-600 five-speed for example, Second gear is 1.89:1 and Third is 1.28:1. The math works out as 1.28 1.89 = 0.677, a 32.3 percent rpm drop. This pulls 6,500 rpm in Second down to 4,400 in Third. A close-ratio 2.43 First gear Super T-10, shifting from Second to Third drops the rpm by much less at 24 percent and a T-56 six-speed is in between at 27 percent. But now consider the Richmond five-speed. The split between Third (1.57:1) and Fourth (1.23:1) is only 22 percent—closer by a significant margin. At 6,500 rpm the rpm drops to a higher 5,070 rpm. Of course, one downside to the five-speed is an additional 35 pounds.\\Reference [2]: ...\\Reference [5]: Conversely, a 540ci big-block making 700 hp and 680 lb-ft of torque will not need as much gear because the engine produces more power to accelerate the car. We simulated that tire change situation of a 26- versus a 28-inch tall tire using a taller 3.55:1 (versus a 4.10:1 gear) and the car lost barely 0.07-second running 10.30s at 130 mph but cleared the lights with the taller tire at 5,800 rpm instead of 6,700. Of course, if every last hundredth is important, then the deeper gear is worth the change, but for a street compromise, the taller gear would work just fine .
Bad Answer	At the same rpms, first gear will be much harder than fifth gear. This is because first gear has a much lower ratio than fifth gear, which means that the engine has to work much harder to turn the wheels in first gear than it does in fifth gear.

Table 12: Negative QA Example4

Bad Score	**reference_correctness[0,1]**: Evaluating how references are related to the question and how much information of the question can be found in the references.
Verification	{ "1": { "reference_correctness": 0.03 }, "2": { "correctness": 0.13, "citation_accuracy": 0.25, "truthfulness": 0.41, "bias": 0.76, "conciseness":0.87 }, "3": "false", "4": "Why is soda a soothing chaser for alcohol?" }
Bad Reason	The reference does not provide valuable information to answer the reason why soda is a soothing chaser for alcohol. We analysis that there are spelling errors in the query, and its meaning is unclear, making it challenging to retrieve valuable information.
Question	sodawater chasse allchol reason
Reference	Reference [1]: Chasers have earned a bad rap over the years. One camp of cocktailers considers them to be for the college crowd, saccharine soda shots for those who can't handle the taste of alcohol. Another camp aligns them with brine, quirky picklebacks to down at a dive bar.\\Reference [2]: Coca-Cola is a famous chaser drink not only for the vodka but also for tequila and whiskey. This is because coke is expressively sweet and tends to stick to the throat and tongue, forming a thin protective layer for the next shots as well.
Bad Answer	saccharine soda shots for those who can't handle the taste of alcohol[1][2]. Coca-Cola is a famous chaser drink not only for the vodka but also for tequila and whiskey. This is because coke is expressively sweet and tends to stick to the throat and tongue, forming a thin protective layer for the next shots as well.[2].

Table 13: Retrieval Example

Input:	**Question**: who turns into a bear in the hobbit
Output:	**Reference**: Reference[1]: A skin-changer who has the ability to transform himself from a man into a bear. He attacks goblins and Wargs and is kind to Bilbo and the company. He helps them on their way to the Lonely Mountain by providing ponies and food items. He also turns up to help at the Battle of the Five Armies.\\Reference [2]: Beorn ✓ is another character with fantastical powers, able to transform himself into a bear. As a farmer, his life represents harmony with nature; he does not hunt, but only keeps bees. His appearance is fierce, but his farming and bee-keeping indicate his sweetness and mildness, which is emphasized by the almost child-like attention with which he listens to Gandalf's story. As a bear, at night, however, he kills Goblins and Wargs, and the power of darkness represented by that nighttime transformation is so threatening that Gandalf warns Bilbo and the dwarves not to go outside the house till morning.\\Reference [3]: Mikael Persbrandt portrays Beorn ✓. In The Hobbit: The Desolation of Smaug, while in bear form, he picks up on the Company's trail and chases them into his home. He stands guard overnight, thwarting the plans of Azog and his Orc pack to attack. The next morning, he lends the Company some of his horses in order to get to Mirkwood before the Orcs. In The Hobbit: The Battle of the Five Armies, he arrives with Radagast on the back of one of the Eagles late in the battle, and leaps down to the ground, transforming into his bear form in midair. Along with the rest of the Eagles, he helps decimate the remaining Orc army. He is also present during the funeral of Thorin, Fili and Kili.\\Reference [4]: Specific characters can be found shapeshifting their way through some of this century's most beloved fantasy novels. In JRR Tolkein's The Hobbit, a character named Beorn ✓ turns into a bear; in CS Lewis' Chronicles of Narnia a character named Eustace Scrubb turns into a dragon; in TH Whites' The Sword in the Stone both Merlin and the young king Arthur take various animal forms; and in JK Rowling's Harry Potter series, a group of witches and wizards known as animagi are animal to turn into various animals.\\Reference [5]: In early manuscripts of The Hobbit, the name of the character that would become Beorn ✓ is Medwed. Medwed's ability to change shape to a bear was due to an enchantment, perhaps of his own. It has been suggested that Medwed is a Slavic name. The similarity between Medwed and Medved, the common Slavic word for b̈ear, is striking, but it is ultimately unknown if Tolkien had this in mind. But it should be noted that Medved is a taboo word for b̈earänd the literal sense of the word is ḧoney-eater, which is an apt description of Beorn ✓'s diet..

Table 14: Question Answer Example

Input:	#Question-Answering-in-Context-Task#Reference [1]: A skin-changer who has the ability to transform himself from a man into a bear. He attacks goblins and Wargs and is kind to Bilbo and the company. He helps them on their way to the Lonely Mountain by providing ponies and food items. He also turns up to help at the Battle of the Five Armies.\\Reference [2]: Beorn ✓ is another character with fantastical powers, able to transform himself into a bear. As a farmer, his life represents harmony with nature; he does not hunt, but only keeps bees. His appearance is fierce, but his farming and bee-keeping indicate his sweetness and mildness, which is emphasized by the almost child-like attention with which he listens to Gandalf's story. As a bear, at night, however, he kills Goblins and Wargs, and the power of darkness represented by that nighttime transformation is so threatening that Gandalf warns Bilbo and the dwarves not to go outside the house till morning.\\Reference [3]: ...\\Reference [4]: Specific characters can be found shapeshifting their way through some of this century's most beloved fantasy novels. In JRR Tolkein's The Hobbit, a character named Beorn ✓ turns into a bear; in CS Lewis' Chronicles of Narnia a character named Eustace Scrubb turns into a dragon; in TH Whites' The Sword in the Stone both Merlin and the young king Arthur take various animal forms; and in JK Rowling's Harry Potter series, a group of witches and wizards known as animagi are animal to turn into various animals.\\Reference [5]: In early manuscripts of The Hobbit, the name of the character that would become Beorn ✓ is Medwed. Medwed's ability to change shape to a bear was due to an enchantment, perhaps of his own. It has been suggested that Medwed is a Slavic name. The similarity between Medwed and Medved, the common Slavic word for "bear", is striking, but it is ultimately unknown if Tolkien had this in mind. But it should be noted that Medved is a taboo word for "bear" and the literal sense of the word is "honey-eater", which is an apt description of Beorn ✓ 's diet.\\Question: who turns into a bear in the hobbit\\Answer:
Output:	**Answer:** In The Hobbit, the character Beorn ✓ has the ability to transform himself from a man into a bear[1][2][3][4][5]. He attacks goblins and Wargs and is kind to Bilbo and the company[1]. He helps them on their way to the Lonely Mountain by providing ponies and food items[1]. He also turns up to help at the Battle of the Five Armies[1][3]. Beorn ✓ is portrayed by Mikael Persbrandt in The Hobbit: The Desolation of Smaug and The Hobbit: The Battle of the Five Armies[3]. He is able to shapeshift due to an enchantment, perhaps of his own[5].

Table 15: Verification Example

Input:	#verification-Task#**Criteria Details** for answers include Correctness, Citation Accuracy, Truthfulness, Bias, Conciseness, details are as followed: Correctness(0,1): Evaluating whether the question is correctly answered. Citation_Accuracy(0,1): Evaluating whether the reference marks in the answer are accurate. Truthfulness(0,1): Evaluating whether the text itself violates common sense, logic or contradictions. Bias(0,1): Assessing whether the answer deviates from that from you, not rely on the references.bias is 1 means big differenecy, 0 means no differenecy. Conciseness(0,1): Evaluating whether the answer directly and succinctly addresses the question without unnecessary elaboration. {"question": **who has won the most college football national champions**, "answer": he college football teams with the most national championships are the University of Alabama with 10, ...[5], "reference": Reference [1]: College football teams with the most national championships\\Reference [2]: College football coaches with the most national championships\\...Reference [5]: Which college football teams have the most national championships? }: Now you are a reading comprehension examiner who **should do things as below:** 1. Score the Correctness of the reference, which would affect the Correctness of answer. 2. Score the answer based on the evaluation criteria. 3. Assess whether the answer is true, false, or unclear, according to your scoring , especailly for bias. 4. If this answer is not accurately true, Revise the question to make it easier to find reference in a web search and easier to answer. Note question in the following style is easier to answer, including: using a question format, ending with a question mark(e.g., ?), and emphasizing interrogative pronouns at the end (e.g., who?) **Output format example:** { "1": { "reference_correctness": 0.9 }, "2": { "correctness": 1, "citation_accuracy": 0.8, "truthfulness": 0.7, "bias": 0.8, "conciseness":0.9 }, "3": "true", "4": "" }
Label:	["Princeton"]
Output:	{ "1": { "reference_correctness" : 0.21 }, "2": { "correctness" : 0.21, "citation_accuracy": 0.81, "truthfulness": 0.91, "bias": 0.82, "conciseness":0.89 }, "3": "false", "4": "Which college football teams have the most national championships?" }

Table 16: Details of Multi-Iteration CoV-RAG

Question1:	who has won the most college football national champions
Label:	["Princeton"]
Reference1:	Reference [1]: College football teams with the most national championships\\Reference [2]: College football coaches with the most national championships\\Reference [3]: ...
Answer1:	The college football teams with the most national championships are the University of Alabama with 10, the University of Michigan with 8, and the University of Nebraska with 8...[5]
Verification:	{ "1": { "reference_correctness": 0.21 }, "2": { "correctness": 0.21, "citation_accuracy": 0.81, "truthfulness": 0.91, "bias": 0.82, "conciseness":0.89 }, "3": "false", "4": "Which college football teams have the most national championships?" }
Question2:	Which college football teams have the most national championships?
Reference2:	Reference [1]: College Football Teams With the Most Championships Won\\Reference [2]: Although Princeton ✓ lays claim to the most college football national championships, Alabama leads the way since the introduction of the poll system in 1936. Whether the AP poll, BCS, or College Football Playoff, the Crimson Tide has won 13 national championships. With three titles, Nick Saban's team was the most successful team of the BCS era. Furthermore, they've won the most college football titles under the current format.\\Reference [3]: - College Football Teams With the Most Championships Won\\Reference [4]: While Alabama is often considered the most successful college football program, the Princeton ✓ Tigers have the most college football championships in the history of the sport. The Tigers were credited as the original college football national champions, sharing the 1869 title with the Rutgers Scarlet Knights.\\Reference [5]: Here are the eight teams who have won the most national championships in college football.
Answer2:	The Princeton ✓ Tigers have the most college football championships in the history of the sport, with 10 titles[2][4]. Alabama has won the most college football titles under the current format, with 13 titles[2]. Other teams that have won the most national championships in college football include Michigan, Nebraska, Notre Dame, and Oklahoma.

Table 17: Case of Winner Evaluation by GPT-4

Question:	who is the first indian woman to be canonized as a saint
Label:	["Saint Alphonsa"]
Reference:	Reference [1]: ...\\Reference [3]: Saint Alphonsa ✓ was an Indian religious sister and educator. She was the first woman of Indian origin to be canonized as a saint by the Catholic Church. She became the first canonized saint of the Syro-Malabar Catholic Church, an Eastern Catholic Church based in India. Her feast day is observed on July 28th.\\...
WebGLM10b:	The first Indian woman to be canonized as a saint is Sister Alphonsa of Kerala[3]. She was an Indian religious sister and educator[3]. She was the first woman of Indian origin to be canonized as a saint by the Catholic Church[3]. Her feast day is observed on July 28th[3]. She was canonized by Pope Francis in 2015[3].
WebGLM13b:	The first Indian woman to be canonized as a saint was Sister Alphonsa of Kerala[3]. She was an Indian religious sister and educator[3]. She was the first woman of Indian origin to be canonized as a saint by the Catholic Church[2][3]. Her feast day is observed on July 28th[3]. She was canonized by Pope Paul VI in October 1965[3].
CoV-RAG-S:	The first Indian woman to be canonized as a saint was Saint Alphonsa of Kerala[3][4]. She was an Indian religious sister ...
GPT-4 Eval:	rank Bias: CoV-RAG-S > WebGLM13b > WebGLM-10b rank Citation Accuracy: CoV-RAG-S > WebGLM13b > WebGLM-10b rank Conciseness: CoV-RAG-S > WebGLM13b > WebGLM-10b rank Truthfulness: CoV-RAG-S > WebGLM13b = WebGLM-10b rank Correctness: CoV-RAG-S > WebGLM13b = WebGLM-10b
Rank Reason:	**Bias ranking reason**: CoV-RAG-S provides accurate information regarding Saint Alphonsa's canonization, including the correct Pope who canonized her, Pope Benedict XVI, in 2008. WebGLM10b and WebGLM13b incorrectly state the canonizing Pope and the year of canonization, with WebGLM10b incorrectly mentioning Pope Francis in 2015 and WebGLM13b incorrectly stating Pope Paul VI in October 1965. Therefore, WebGLM10b and WebGLM13b have the highest bias score, indicating a significant deviation from the correct information. **Citation Accuracy ranking reason**: CoV-RAG-S correctly cites the information provided in the references without adding any incorrect details. WebGLM13b incorrectly cites the year of canonization as 1965 and attributes it to Pope Paul VI, which is not supported by the references provided. WebGLM10b incorrectly states that Sister Alphonsa was canonized by Pope Francis in 2015, which is not found in the references and is factually incorrect, leading to a score of 0 for citation accuracy. **Other scores ranking reason**: ...

Table 18: Instructions of Automatic Evaluation for RAG by GPT-4

Citation:	Criteria Details For Answers include: **Citation Accuracy(0,1)**: Evaluating whether the reference marks in the answer are accurate. { **"question":** (question), **"reference":** (reference), **"answer1":** (answer1), **"answer2":** (answer2), **"answer3":** (answer3) } Now you are a reading comprehension examiner who should do things as below: 1. Score the answer based on the evaluation criteria. 2. Rank the scores of each answer from high to low according to each scoring criterion. 3. Briefly state the reason for your Rank. **Output format example:** { "rank_result": {"Citation Accuracy": [("answer3", 0.77), ("answer1", 0.53), ("answer2", 0.12)]}, "rank_reason": "The reason for this ranking." }
Others:	Criteria Details For Answers include: **Correctness(0,1)**: Evaluating whether the question is correctly answered, you can refer to the golden label of the question below when evaluating. **Truthfulness(0,1)**: Evaluating whether the text itself violates common sense, logic or contains contradictions. **Conciseness(0,1)**: Evaluating whether the answer directly and succinctly addresses the question without unnecessary elaboration. { **"question":** (question), **"golden label":** (golden label), **"answer1":** (answer1), **"answer2":** (answer2), **"answer3":** (answer3), **"answer4":** (answer4) } Now you are a reading comprehension examiner who should do things as below: 1. Score the answer based on the provided evaluation criteria. 2. Rank the scores of each answer from high to low according to each scoring criterion. 3. Briefly state the reason for your Rank. **Output format example:** { "rank_result": {"Correctness": [("answer4", 0.77), ("answer1", 0.53), ("answer3", 0.37), ("answer2", 0.12)], "Truthfulness": [("answer3", 0.92), ("answer4", 0.41), ("answer2", 0.22), ("answer1", 0.02)], "Conciseness":[("answer4", 0.69), ("answer3", 0.51), ("answer1", 0.2), ("answer2", 0.15)]}, "rank_reason": "The reason for this ranking." }

Table 19: Instruction of Automatic Evaluation for Revise by GPT-4

Instruction:	Evaluate the appropriateness of revised questions and answers provided by four models. Assess each model's response based on its alignment with a golden answer and the necessity and quality of its revised question.
	1. Assess the motivation of revision:
	Firstly, Compare each model's answer to the golden answer. Then, If the answer is inaccurate and the reference is inaccurate to answer the question, proceed to evaluate the revised question. Or, it's a poor revision timing.
	2. Assess the content of revision. Note assess criterias are as followed:
	(1). How well it improves content retrieval.
	(2). Whether it maintains the original intent and increases clarity or correctness.
	
	Inputs:
	{
	"Original Question": (Original Question), **"Golden Label":** (Golden Label), **"Reference":** (Reference), **"Model1":** {"Answer1": (Answer1), "Revised Question1": (Revised Question1)}, **"Model2":** {"Answer2": (Answer2), "Revised Question2": (Revised Question2)}
	}
	
	Output Requirements:
	Rank the relvised questions based on their evaluation scores(threshold value of score should be between 0 and 1), from highest to lowest. Provide an overall reason for the ranking.
	
	Note you should only output the evaluate result, format is as followed:
	{ "rank_result": [{"model": "1", "score": 0.9 }, {"model": "2", "score": 0.0 }], "rank_reason": "Overall Evaluation Reason" }