RJAG: Retrieval Judgment Augmented Generation

Kuangzhi Wang, Zhenhua Hu, Min Ren*, Xiangzhi Tao
University of Science and Technology of China, Hefei, China
{wangkz23, siche}@mail.ustc.edu.cn
renmin66@ustc.edu.cn
xztao@mail.ustc.edu.cn

Abstract

Large Language Models (LLMs) inevitably suffer from hallucinations, as relying solely on their parametric knowledge cannot guarantee the accuracy of generated content. To enhance text generation, retrieval-augmented generation (RAG) is proposed to incorporate external knowledge to achieve this. However, its effectiveness heavily depends on the relevance of retrieved documents, which poses a critical challenge: how to ensure the accuracy and reliability of model responses when retrieval results are inaccurate. Tackling this challenge, we propose **R**etrieval **J**udgment **A**ugmented **G**eneration (RJAG), a method that can enhance RAG through LLM-driven fine-grained relevance judgment mechanism and a task-adaptive knowledge combination strategy. RJAG judges and dynamically combines retrieved documents for both open-ended generation and closed-ended selection tasks. Additionally, large-scale web search is also included to expand the knowledge beyond static corpora. Experimental results on multiple benchmarks show that RJAG outperforms existing RAG methods, which will significantly enhance the accuracy and reliability while maintaining the system's simplicity. Code is available at https://github.com/wangkz2023/RJAG.

Keywords: Large Language Model , RAG , LLM-as-a-judge

1 Introduction

Large language models (LLMs) have gained increasing attention in natural language processing due to their strong ability to follow instructions and generate coherent text (Touvron et al., 2023). However, relying solely on their encapsulated parametric knowledge for text generation presents two inherent limitations: (1) LLMs are prone to hallucinations, generating content that may seem plausible but lacks factual accuracy (Huang et al., 2025). (2) Due to the static nature of their parameters, LLMs cannot update their knowledge in real time, making it difficult to include the latest information (Gao et al., 2024).

To mitigate these limitations, prior research has proposed retrieval-augmented generation (RAG), which enhances text generation by retrieving the knowledge relevant to the input and incorporating it into the generation process (Gao et al., 2024; Lewis et al., 2020). This approach can improve generation quality without increasing the number of model parameters (Guu et al., 2020). However, the performance of RAG systems heavily relies on the retrieval quality (Li et al., 2022; Gao et al., 2024). As shown in Fig. 1, low-quality retrieval results will introduce noise or irrelevant information into the generation process, which may exacerbate hallucinations and undermine generation reliability (Zhang et al., 2023b). Through careful analysis, we observe that there exist two primary issues of current RAG methods. Firstly, they lack a fine-grained document evaluation mechanism to consider the varying degrees of relevance between retrieved documents and the input. Secondly, they treat all retrieved results indiscriminately, which leads to inefficient knowledge integration (Rony et al., 2022).

*Corresponding author.

Proceedings of the 24th China National Conference on Computational Linguistics, pages 960-971, Jinan, China, August 11-14, 2025.

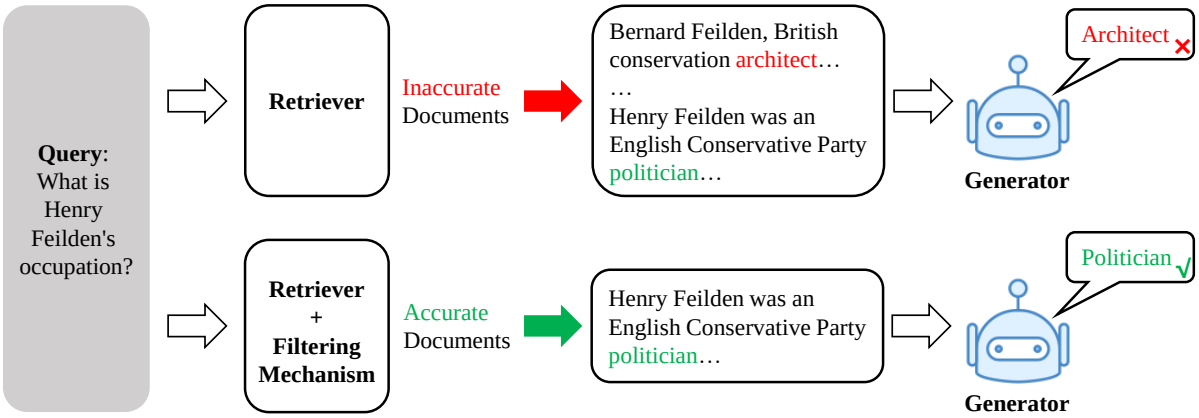

Figure 1: The example shows that the retriever tends to introduce substantial irrelevant information, which, if not properly filtered, can hinder the generator from acquiring accurate knowledge and may lead to erroneous outputs.

Focusing on above issues, this paper proposes **R**etrieval **J**udgment **A**ugmented **G**eneration (RJAG), a framework that can achieve significant improvements of prior RAG method's performance in various task scenarios. To address the problem of relevance differences among documents, RJAG introduces a three-level relevance judgment mechanism that evaluates, labels, and filters retrieved documents, ensuring that only the most relevant information will be utilized (Li et al., 2025; Zhuang et al., 2024). To mitigate the problem of the inefficiencies in knowledge integration, RJAG employs a task-adaptive knowledge combination strategy that prioritizes highly relevant documents and dynamically expands the knowledge corpus when necessary (Piktus et al., 2022; Komeili et al., 2022). Combining the above two points, RJAG can simply and effectively enhance the accuracy and reliability of RAG system. The experimental results on four benchmark datasets demonstrate that RJAG outperforms the state-of-the-art method CRAG (Yan et al., 2024).

The main contributions of this paper are as follows:

- We propose RJAG, an effective framework to improve the accuracy and reliability of generated results. We employ a fine-grained judgment mechanism to consider relevance differences among retrieved documents, and propose a task-adaptive strategy to integrate knowledge effectively.
- We adopt an end-to-end design that bridges relevance judgment and task-adaptive knowledge construction, which, to the best of our knowledge, has not been systematically explored or validated in prior work.
- We conduct extensive experiments on multiple datasets. Experimental results show that RJAG outperforms the state-of-the-art CRAG, demonstrating its broad applicability and effectiveness across diverse task scenarios.

2 Related Work

Hallucinations of LLMs. LLMs perform well in instruction comprehension and text generation, but their tendency to produce hallucinations will undermine the overall reliability (Huang et al., 2025; Zhang et al., 2023b). These errors primarily stem from outdated or inaccurate parametric knowledge, low-quality training data, and uneven data distribution. To mitigate hallucinations, researchers have proposed various methods such as knowledge editing, retrieval augmentation, and self-consistency (Huang et al., 2025). Among these approaches, retrieval-augmented generation is one of the most widely adopted solutions.

Proceedings of the 24th China National Conference on Computational Linguistics, pages 960-971, Jinan, China, August 11-14, 2025.

Retrieval-Augmented Generation. RAG is an effective approach to enhance generative language models by incorporating external knowledge (Lewis et al., 2020; Guu et al., 2020; Karpukhin et al., 2020). The theoretical framework of this approach is based on two key components: (1) retrieving documents relevant to the input query from a specific corpus (e.g., Wikipedia); (2) feeding these retrieved documents into the generative model as contextual information. This retrieval-augmented mechanism effectively alleviates the limitations of pre-trained language models in retaining parametric knowledge, thereby substantially improving their performance on knowledge-intensive tasks (Karpukhin et al., 2020). However, its effectiveness heavily depends on the retrieval quality, as irrelevant or inaccurate reference knowledge may introduce noise and mislead the generation process (Gao et al., 2024).

Advanced RAG. In recent years, advanced retrieval-augmented generation has achieved significant progress on the basis of traditional architectures. SelfRAG (Asai et al., 2024) innovatively introduces a critique model to dynamically determine the retrieval timing, effectively reducing redundant retrieval operations. However, this approach requires a complex training process and multiple rounds of label generation and evaluation during the generation phase. Adaptive-RAG (Jeong et al., 2024) can dynamically select the most suitable strategy for (retrieval-augmented) LLMs from the simplest to the most sophisticated ones based on the query complexity. SAIL (Luo et al., 2023) grounds the language generation and instruction following abilities on complex search results generated by in-house and external search engines. SKR (Qiao et al., 2025) enhances the retrieval stage through support-driven knowledge rewriting, aiming to improve the quality of retrieved results. Rowen (Ding et al., 2024) focuses on retrieval triggering by determining whether to retrieve based on consistency checking. CRAG (Yan et al., 2024) proposes a confidence-based retrieval strategy to assess the overall quality of retrieved documents for a query using a lightweight evaluator, integrates large-scale web search for better information acquisition, and optimizes document utilization with a decomposition-recombination algorithm, thereby improving the robustness of generation.

LLM-as-a-judge. The *LLM-as-a-judge* paradigm has recently emerged as an important methodological breakthrough in natural language processing. By leveraging LLMs' semantic understanding and text generation capabilities, it enables automated scoring, ranking, and selection across various tasks. Sun et al. (Sun et al., 2023) proposed a method for instruction-based ordering, where LLMs are directly prompted to generate a permutation of a set of paragraphs. This approach performs relevance ranking by inputting a query concatenated with all documents into the LLM. Unlike traditional evaluation methods based on static metrics and keyword matching, LLM-as-a-judge offers more fine-grained, human-like qualitative assessments (Li et al., 2025). This approach has shown notable advantages in diverse applications, including open-ended text generation and reasoning tasks (Li et al., 2025; Zhuang et al., 2024). Therefore, we introduce it into the RAG framework to judge the relevance of retrieved documents.

Previous studies (Luo et al., 2023; Asai et al., 2024; Yan et al., 2024) have achieved promising results. However, they face two main limitations. On the one hand, they need to train an auxiliary model to support decision-making, thus increasing the complexity of the RAG system. On the other hand, they lack a comprehensive framework for knowledge evaluation and utilization. To address these challenges, we firstly introduce a large language model instead of a self-training auxiliary model to assess the relevance between retrieved documents and the input query. This relevance judgment mechanism can assign corresponding labels to the retrieved documents. Considering various task scenarios, we also design a task-adaptive knowledge combination strategy to integrate and utilize the labeled documents, thereby constructing a complete framework to evaluate and utilize the knowledge. Unlike methods (Qiao et al., 2025; Ding et al., 2024) that mainly optimize the retrieval stage, our approach emphasizes enhancing the accuracy and reliability of the generation stage. In contrast to the existing method proposed by Sun et al. (Sun et al., 2023), our preliminary experiments reveal that directly applying this approach within the RAG framework introduces significant challenges, including severe context interference and difficulty in determining an appropriate threshold for the number of documents to retrieve per query. These issues undermine the stability and reliability of the overall evaluation process. To address this, we adopt a document-wise evaluation strategy, where each document is individually paired with the query and fed

into the LLM to assess its relevance. This enables more flexible and fine-grained knowledge selection and composition.

3 Method

3.1 Problem Formalization

Following previous work (Lewis et al., 2020; Asai et al., 2024; Yan et al., 2024), we also adopt the standard framework of RAG. Given input X and an accessible corpus containing a large number of knowledge documents $C = \{d_1, ..., d_N\}$, the system sequentially generates a textual output Y. The overall framework typically consists of two core components: a retriever R and a generator G. The retriever R selects the top-K most relevant documents from the corpus C based on the input X, forming a retrieved document set $D = \{d_{r_1}, ..., d_{r_k}\}$. Based on the input X and the retrieved results D, the generator G produces the output Y. This framework can be formalized as:

$$P(Y|X) = \sum_{D} P(Y|X, D)P(D|X) \tag{1}$$

This formulation highlights the close coupling between the retriever and the generator, resulting in a tightly integrated generation pipeline. However, such interdependence also introduces a fundamental limitation: the overall system performance is heavily influenced by the quality of retrieved documents, as retrieval failures or the inclusion of irrelevant information can significantly degrade the quality of the generated output. To address this issue, this paper focuses on improving the accuracy and reliability of generation results when retrieval results are inaccurate.

3.2 Overview of Model Inference

The overall framework of RJAG at inference is illustrated in Fig. 2 and Alg. 1. This method improves the accuracy and reliability of generated results by evaluating the relevance of retrieved documents and integrating external knowledge accordingly. Given an input query and a set of documents retrieved by the retriever, RJAG employs a large language model to assess the relevance of retrieved documents to the input query, and assign a corresponding label to each of them (Section 3.3). Based on the task types (open-ended generation tasks or closed-ended selection tasks), RJAG dynamically selects the most appropriate strategy to combine the labeled documents for subsequent generation process (Section 3.4). Due to its simplicity and effectiveness, RJAG can be integrated with any generative model.

3.3 Relevance Judgment Mechanism

Before utilizing the retrieved documents, it is crucial to determine their relevance to the input, as this helps identify irrelevant or misleading information. Therefore, the effectiveness of the relevance judgment mechanism plays a vital role in shaping the overall system performance, as it directly influences the outcomes of subsequent processes. Our objective is to evaluate the retrieved documents, providing a basis for subsequent knowledge utilization. Specifically, based on their relevance to the input query, the judgment mechanism classifies the retrieved documents into three categories and assigns each of them a corresponding relevance label: *Highly Relevant*, *Somewhat Relevant*, or *Not Relevant*. Given the strong semantic understanding and contextual analysis capabilities of LLMs, we directly employ a high-performance LLM as the evaluator. For every question, there are generally 10 documents retrieved. The question is concatenated with each single document as the input, and the evaluator assesses each question-document pair individually. Notably, a binary classification scheme ("Relevant" and "Not Relevant") may compromise system reliability, as LLMs often struggle with ambiguous cases that fall between the two categories, leading to classification errors or inconsistencies (Zhuang et al., 2024). To address this, our relevance judgment mechanism introduces an intermediate label, "Somewhat Relevant", to enable a more nuanced assessment of relevance.

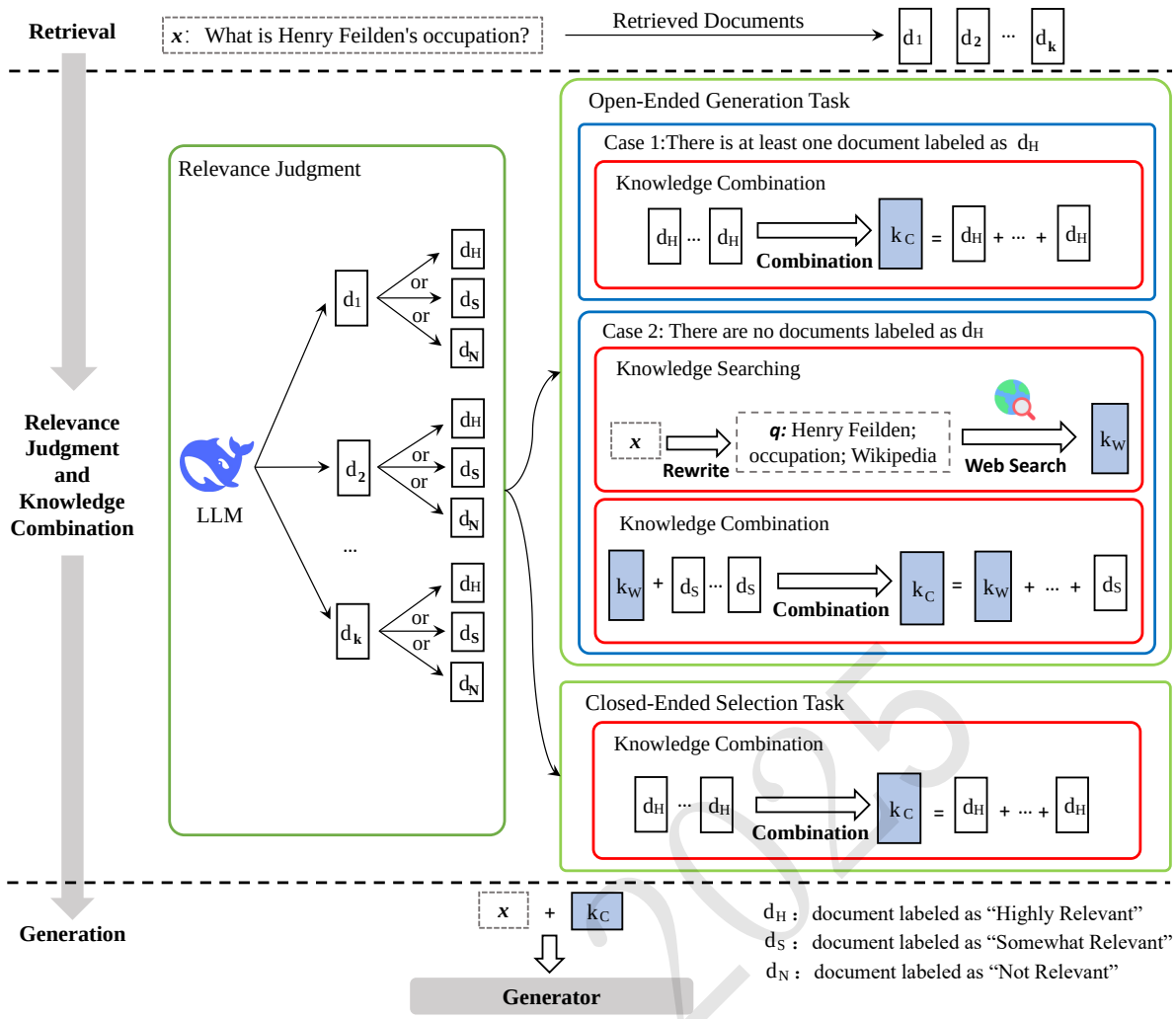

Figure 2: An overview of RJAG at inference. A large language model is employed to judge the relevance between retrieved documents and the input, and a task-adaptive knowledge combination strategy is adopted based on the task types.

3.4 Knowledge Combination Strategy

Based on above classification results, we propose an adaptive knowledge combination strategy to integrate the most relevant labeled documents in different task scenarios (the open-ended generation and closed-ended selection).

Open-Ended Generation Tasks. Open-ended generation tasks require the model to generate coherent and reasonable responses based on a given input. These tasks are suitable for scenarios that demand creative or flexible outputs. For these tasks, we design a knowledge combination strategy to integrate retrieved documents, and leverages web search to complement them when necessary.

► **Knowledge Combination.** For open-ended generation tasks, RJAG employs a dynamic knowledge combination strategy to enhance the accuracy and reliability of generated results. This strategy is based on the assumption that documents labeled "Highly Relevant" contain key information closely aligned with the input question, and can provide the most reliable knowledge for generation. Specifically, for a given input question, when one or more retrieved documents are labeled as "Highly Relevant", the system will prioritize integrating all such documents as reference knowledge. However, if no documents are labeled as "Highly Relevant", the system will activate a knowledge expansion mechanism to

Algorithm 1 RJAG Inference

Require: LLM (LLM Judge), W (Query Rewriter), G (Generator)
Input: x (Input question), $D = \{d_1, d_2, \ldots, d_k\}$ (Retrieved documents)
Output: y (Generated response)

$label_i$ = LLM judges the relevance of each pair $(x, d_i), d_i \in D$.
// label has 3 possible values: [Highly Relevant], [Somewhat Relevant] or [Not Relevant]
if x is an Open-Ended Generation Task **then**
  for each document $d_i \in D$ **do**
    if $label_i ==$ [Highly Relevant] **then**
      Add d_i to K_c
    end if
  end for
  if $K_c \neq \emptyset$ **then**
    $k_c = K_c$
  else
    $k_W =$ Web_Search(W rewrites x for searching)
    for each document $d_i \in D$ **do**
      if $label_i ==$ [Somewhat Relevant] **then**
        Add d_i to K_c
      end if
    end for
    $k_c = k_W + K_c$
  end if
else if x is a Closed-Ended Selection Task **then**
  for each document $d_i \in D$ **do**
    if $label_i ==$ [Highly Relevant] **then**
      Add d_i to K_c
    end if
  end for
  if $K_c \neq \emptyset$ **then**
    $k_c = K_c$
  else
    $k_c =$ None
  end if
end if
$y = G$ predicts y given x and k_c

acquire additional external knowledge through large-scale web search. Then it will combine the newly acquired web knowledge and the documents labeled as "Somewhat Relevant" to construct the final reference knowledge. This dynamic knowledge combination strategy effectively mitigates the issues caused by irrelevant or insufficiently relevant retrieval results. Additionally, by incorporating web knowledge, the model's knowledge coverage is broadened, thereby improving both the accuracy and reliability of generated results in open-ended generation tasks.

► **Web Search.** A truly intelligent system should possess the capability to evaluate whether its existing knowledge corpus is adequate to answer a given question and proactively seek additional external knowledge when necessary. Given that retrieval from a static knowledge corpus frequently results in reference knowledge with limited scope and inadequate content diversity, large-scale web search (Piktus et al., 2022; Komeili et al., 2022) has been introduced as a strategic extension of RAG. Therefore, in scenarios where no retrieved documents are deemed "Highly Relevant", actively acquiring additional knowledge becomes essential to ensure the reliability and factual consistency of the system's responses.

In the framework of RJAG, the inputs are rewritten into queries composed of keywords by ChatGPT to simulate the way users search. Specifically, we employ publicly accessible commercial web search APIs to generate a set of URLs corresponding to each query, and then access the content from these links.[1] However, large-scale web searches may introduce noise or unreliable information that may compromise the accuracy and reliability of generated results. To mitigate these risks, we prioritize authoritative and regulated knowledge sources, such as Wikipedia, to enhance the credibility and quality of the search results.

Closed-Ended Selection Tasks. Closed-ended selection tasks require the model to make choices or judgments from a predefined set of options. These tasks are suitable for scenarios that demand high accuracy and confidence in the responses.

► **Knowledge Combination.** For closed-ended selection tasks, RJAG employs a strict knowledge combination strategy to ensure reliable selections and judgments. Specifically, when one or more retrieved documents are labeled as "Highly Relevant", the system will integrate all such documents as reference knowledge. Conversely, if no retrieved documents are labeled as "Highly Relevant", the system will not utilize any reference knowledge, meaning the model will not rely on external knowledge for generating the answer. Since closed-ended selection tasks typically have only one correct answer, this design aligns with their high accuracy requirements and minimizes the risk of introducing low-relevance or misleading information, thereby reducing potential selection errors.

4 Experiments

We conduct a series of experiments to demonstrate RJAG's generalizability and practicality across both open-ended generation and closed-ended selection tasks.

4.1 Tasks, Datasets and Metrics

To comprehensively evaluate RJAG's performance, we conduct experiments on four datasets, including **PopQA** (Mallen et al., 2023) (*short-form* generation), **Biography** (Min et al., 2023) (*long-form* generation), **PubHealth** (Zhang et al., 2023a) (*true-or-false* question), and **Arc-Challenge** (Bhakthavatsalam et al., 2021) (*multiple-choice* question). Following previous studies, we adopt accuracy as the evaluation metrics for PopQA, PubHealth, and ARC-Challenge, while use FactScore (Min et al., 2023) for Biography. We employ the same metrics to ensure comparability with previous studies, and use the same retrieval results. The key difference lies in that our motivation is to improve the knowledge utilization by discriminate treatment of retrieved documents.

4.2 Baselines

We evaluate two public LLMs: LLaMA2-7B and Alpaca-7B, and three representative retrieval-augmented generation methods: (1) Standard RAG (Lewis et al., 2020), which serves as the basic retrieval-augmented generation approach; (2) Self-RAG (Asai et al., 2024), a representative advanced RAG method; (3) CRAG (Yan et al., 2024), the state-of-the-art advanced RAG method. To ensure fairness, the same retriever is used for document retrieval. This rigorous setup guarantees that performance differences arise solely from the core algorithmic design of each method, providing an objective evaluation of RJAG's effectiveness.

4.3 Results

The results on four datasets are presented in table 1. Notably, all experiments use *DeepSeek-V3* as the relevance judgment model and *LLaMA2-hf-7b* as the generation model. From these results, we can conclude the following findings:

First, compared to the baseline methods, RJAG shows significant improvements across all datasets. Specifically, as shown in table 1, RJAG significantly outperforms these two base models. Since Self-RAG requires a specialized model to achieve optimal results, direct comparison in this context may not

[1] In this study, Google Search API is utilized for searching.

Proceedings of the 24th China National Conference on Computational Linguistics, pages 960-971, Jinan, China, August 11-14, 2025.

Method	PopQA (Accuracy)	Bio (FactScore)	Pub (Accuracy)	ARC (Accuracy)
LLaMA2_{7B}	14.7	44.5	34.2	21.8
Alpaca_{7B}	23.6	45.8	49.8	45.0
RAG	50.5	71.4	48.9	43.4
Self-RAG*	29.0	32.2	0.7	23.9
CRAG	54.9	75.9	59.5	53.7
RJAG	**59.8**	**77.5**	**61.5**	**56.9**

* : The evaluation results are cited from the paper of CRAG.

Table 1: Overall evaluation results on the test sets of the four datasets. **Bold** numbers indicate the best performance among all methods. Other results, except RJAG, Self-RAG, and the Biography dataset, are cited from their original papers.

yield meaningful insights. Compared to the standard RAG method, RJAG achieves a 9.3% accuracy improvement on PopQA, a 6.1% increase in FactScore on Biography, a 12.6% accuracy improvement on PubHealth, and a 13.5% improvement on Arc-Challenge. Notably, compared to the current state-of-the-art method CRAG, RJAG maintains a performance advantage of 4.9%, 1.6%, 2.0%, and 3.2% on these four datasets, respectively. These results demonstrate the effectiveness of RJAG in improving the accuracy and reliability of generated results. [2]

Second, the experimental results demonstrate RJAG's strong generalizability across a diverse range of generation tasks. In particular, as shown in table 1, we conduct experiments on benchmark datasets that represent a range of practical scenarios including short-form generation (PopQA), long-form generation (Biography), true-or-false question (PubHealth), and multiple-choice question (Arc-Challenge). These results present the consistent effectiveness of RJAG. Its stable performance gains across diverse task types underscore its broad applicability in various real-world scenarios.

Third, RJAG adopts optimal knowledge combination strategies for open-ended generation and closed-ended selection tasks. Specifically, the performance of various combination strategies is presented in table 2. Based on these results, we identify the optimal strategies for different tasks: (1) For open-ended generation tasks (PopQA and Biography datasets), the optimal strategy is as follows: When one or more retrieved documents are labeled as "Highly Relevant", the system prioritizes integrating all such documents as reference knowledge. However, if no documents are labeled as "Highly Relevant", the system combines web knowledge and documents labeled "Somewhat Relevant" to construct the final reference knowledge. This strategy outperforms others by achieving an accuracy of 59.8% on PopQA and a FactScore of 77.5% on Biography. (2) For closed-ended selection tasks (PubHealth and Arc-Challenge datasets), we summarize the optimal strategy as follows: When retrieved documents labeled as "Highly Relevant" are available, the system solely uses these documents as reference knowledge. Conversely, if no retrieved documents are labeled as "Highly Relevant", the system does not use any reference knowledge. This strategy outperforms others by achieving an accuracy of 61.5% on PubHealth and 56.9% on Arc-Challenge. Experimental results demonstrate that this task-adaptive combination strategy not only enhances generation quality but also effectively filters out irrelevant or misleading information, thereby facilitating more effective knowledge utilization.

Fourth, the experimental results demonstrate RJAG's scalability and generality across different LLMs. In particular, as shown in table 3, we conduct experiments on the PopQA dataset using three representative large language models with distinct architectures and scales: *LLaMA2-hf-7b*, *Mistral-7B*, and *Qwen2.5-32B*. The results consistently show that our proposed RJAG outperforms both the standard

[2]Notably, previous studies used text-davinci-003 for FactScore evaluation, but it was deprecated on January 4, 2024, and replaced by gpt-3.5-turbo-instruct. To align with current standards, we use gpt-3.5-turbo-instruct for all FactScore results, except those of SelfRAG.

Knowledge Combination Strategy	PopQA (Accuracy)	Bio (FactScore)	Pub (Accuracy)	ARC (Accuracy)
D_H[1]	51.9	72.8	**61.5**	**56.9**
$D_H \cup_{\neq\emptyset}$[2] D_S[3]	54.2	73.0	59.5	54.9
$(D_H \cup_{\neq\emptyset} D_S) \cup_{\neq\emptyset} D_N$[4]	55.3	72.5	59.2	53.5
$D_H \cup$[5] D_S	57.0	73.7	57.1	49.4
$(D_H \cup D_S) \cup_{\neq\emptyset} D_N$	57.7	73.1	56.0	48.8
$D_H \cup D_S \cup D_N$	55.1	71.9	54.2	47.3
$(D_H \cup D_S) \cup_{\neq\emptyset} K_W$[6]	58.7	76.0	57.0	50.3
$D_H \cup_{\neq\emptyset} (D_S \cup K_W)$	**59.8**	**77.5**	59.2	52.1

[1] D_H: The set of documents labeled "Highly Relevant".
[2] $\cup_{\neq\emptyset}$: Non-empty priority combination operation, which prioritizes non-empty sets among candidate knowledge sources. If a higher-priority set is empty, the next available candidate set is selected until a non-empty set is found.
[3] D_S: The set of documents labeled "Somewhat Relevant".
[4] D_N: The set of documents labeled "Not Relevant".
[5] $\cup$: Standard set union operation.
[6] K_W: The set of external knowledge obtained through web search.

Table 2: Overall evaluation results on the test sets of the four datasets, assessing the impact of different knowledge combination strategies on generation performance. **Bold** numbers indicate the best performance among all strategies.

RAG and CRAG across all models. Specifically, under *LLaMA2-hf-7b*, RJAG improves the accuracy from 50.5 (RAG) and 54.9 (CRAG) to 59.8, demonstrating a substantial performance gain. Similarly, for *Mistral-7B*, RJAG achieves an accuracy of 64.9, surpassing RAG (60.4) and CRAG (64.0). Notably, on the larger-parameter *Qwen2.5-32B* model, RJAG attains an accuracy of 68.4, compared to 65.8 and 66.9 by RAG and CRAG, respectively. These results validate the effectiveness and robustness of RJAG regardless of the underlying LLM used as the generator.

LLM	Method	PopQA (Accuracy)
LLaMA2-hf-7b	RAG	50.5
	CRAG	54.9
	RJAG	**59.8**
Mistral-7B	RAG	60.4
	CRAG	64.0
	RJAG	**64.9**
Qwen2.5-32B	RAG	65.8
	CRAG	66.9
	RJAG	**68.4**

Table 3: Overall evaluation results on the PopQA dataset using three different large language models as the generator. **Bold** numbers indicate the best performance among all methods.

4.4 Ablation Study

The impact of the relevance judgment. To further verify the effectiveness of relevance judgment, we present the ablation experiment results in table 4, which show the impact of removing this step. When the relevance judgment step is removed, the system can no longer combine knowledge based on the labels of retrieved documents. Instead, all retrieved documents are directly fed into the generation model,

reverting the system to a traditional RAG model. From these results, it can be seen that removing the relevance judgment step leads to an average performance drop of 10.4%, including a significant 13.5% decrease in accuracy on Arc-Challenge. These results illustrate the critical role of relevance judgment in filtering out low-relevance or even irrelevant information.

	PopQA (Accuracy)	Bio (FactScore)	Pub (Accuracy)	ARC (Accuracy)
RJAG	**59.8**	**77.5**	**61.5**	**56.9**
w/o. Relevance Judgment	50.5	71.4	48.9	43.4
w/o. Knowledge Combination	55.1	71.9	54.2	47.3

Table 4: Ablation study for removing the relevance judgment and knowledge combination on four datasets in terms of accuracy or FactScore.

The impact of the knowledge combination. The ablation study results for knowledge combination are also presented in table 4. When the knowledge combination step is removed, the system is simplified to input retrieved documents into the generator according to a fixed priority ("Highly Relevant" → "Somewhat Relevant" → "Not Relevant"). The experimental results show that the optimal combination strategy improves the generation quality by an average of 6.8%. Notably, the improvement in accuracy on Arc-Challenge is particularly significant, reaching 9.6%. These results illustrate the key role of the adaptive knowledge combination mechanism in optimizing knowledge utilization.

5 Conclusion and Limitation

This paper studies the problem that how to ensure the accuracy and reliability of model responses when retrieval results are inaccurate. To this end, we proposed Retrieval Judgment Augmented Generation (RJAG) to enhance RAG. RJAG adopts an LLM-driven three-level relevance judgment mechanism to evaluate retrieved documents and assign relevance labels to them. Based on these labels, we employs a task-adaptive knowledge combination strategy to integrate knowledge effectively. By further leveraging web search, filtering out low-relevance documents and optimizing knowledge utilization, RJAG can significantly improve generation quality. Experiments extensively demonstrate its generalizability across open-ended generation and closed-ended selection tasks. While our primary contribution lies in enhancing RAG through a judgment mechanism and combination strategy, we acknowledge several limitations. First, the current method heavily relies on the accuracy of relevance judgments, which may introduce noise or bias. Second, the task-adaptive strategy is primarily based on empirical heuristics and lacks formal modeling or learning capabilities. In future work, we aim to reduce the dependency on judgment accuracy and explore incorporating reinforcement learning or meta-learning techniques to enable automatic strategy optimization, thereby improving adaptability and theoretical rigor.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Sumithra Bhakthavatsalam, Daniel Khashabi, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, and Peter Clark. 2021. Think you have solved direct-answer question answering? try arc-da, the direct-answer ai2 reasoning challenge. *arXiv preprint arXiv:2102.03315*.

Hanxing Ding, Liang Pang, Zihao Wei, Huawei Shen, and Xueqi Cheng. 2024. Retrieve only when it needs: Adaptive retrieval augmentation for hallucination mitigation in large language models. *arXiv preprint arXiv:2402.10612*.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. 2024. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

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

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. 2024. Adaptive-RAG: Learning to adapt retrieval-augmented large language models through question complexity. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 7036–7050.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wentau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 6769–6781.

Mojtaba Komeili, Kurt Shuster, and Jason Weston. 2022. Internet-augmented dialogue generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pages 8460–8478.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*.

Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. 2022. A survey on retrieval-augmented text generation. *arXiv preprint arXiv:2202.01110*.

Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. 2025. From generation to judgment: Opportunities and challenges of llm-as-a-judge. *arXiv preprint arXiv:2411.16594*.

Hongyin Luo, Tianhua Zhang, Yung-Sung Chuang, Yuan Gong, Yoon Kim, Xixin Wu, Helen Meng, and James Glass. 2023. Search augmented instruction learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 3717–3729.

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. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pages 9802–9822.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100.

Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Dmytro Okhonko, Samuel Broscheit, Gautier Izacard, Patrick Lewis, Barlas Oğuz, Edouard Grave, Wen tau Yih, and Sebastian Riedel. 2022. The web is your oyster - knowledge-intensive nlp against a very large web corpus. *arXiv preprint arXiv:2112.09924*.

Zile Qiao, Wei Ye, Yong Jiang, Tong Mo, Pengjun Xie, Weiping Li, Fei Huang, and Shikun Zhang. 2025. Supportiveness-based knowledge rewriting for retrieval-augmented language modeling. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 2728–2740.

Md Rashad Al Hasan Rony, Ricardo Usbeck, and Jens Lehmann. 2022. DialoKG: Knowledge-structure aware task-oriented dialogue generation. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 2557–2571.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is ChatGPT good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14918–14937.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation. *arXiv preprint arXiv:2401.15884*.

Tianhua Zhang, Hongyin Luo, Yung-Sung Chuang, Wei Fang, Luc Gaitskell, Thomas Hartvigsen, Xixin Wu, Danny Fox, Helen Meng, and James Glass. 2023a. Interpretable unified language checking. *arXiv preprint arXiv:2304.03728*.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, Longyue Wang, Anh Tuan Luu, Wei Bi, Freda Shi, and Shuming Shi. 2023b. Siren's song in the ai ocean: A survey on hallucination in large language models. *arXiv preprint arXiv:2309.01219*.

Honglei Zhuang, Zhen Qin, Kai Hui, Junru Wu, Le Yan, Xuanhui Wang, and Michael Bendersky. 2024. Beyond yes and no: Improving zero-shot LLM rankers via scoring fine-grained relevance labels. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 358–370.