Fine-grained Knowledge Enhancement for Retrieval-Augmented Generation

Jingxuan Han[1], **Zhendong Mao**[1]*, **Yi Liu**[3], **Yexuan Che**[1]
Zheren Fu[1] and **Quan Wang**[2]
[1]University of Science and Technology of China, Hefei, China
[2]MOE Key Laboratory of Trustworthy Distributed Computing and Service,
Beijing University of Posts and Telecommunications, Beijing, China
[3]State Key Laboratory of Communication Content Cognition, People's Daily Online
{hjx999222,yexuanche,fzr}@mail.ustc.edu.cn, liuyi2023@people.cn
wangquan@bupt.edu.cn, zdmao@ustc.edu.cn

Abstract

Retrieval-augmented generation (RAG) effectively mitigates hallucinations in large language models (LLMs) by filling knowledge gaps with retrieved external information. Most existing studies primarily retrieve knowledge documents based on semantic similarity to assist in answering questions but ignore the fine-grained necessary information within documents. In this paper, we propose a novel fine-grained knowledge enhancement method (FKE) for RAG, where fine-grained knowledge primarily includes sentence-level information easily overlooked in the document-based retrieval process. Concretely, we create a disentangled Chain-of-Thought prompting procedure to retrieve fine-grained knowledge from the external knowledge corpus. Then we develop a decoding enhancement strategy to constrain the document-based decoding process using fine-grained knowledge, thereby facilitating more accurate generated answers. Given an existing RAG pipeline, our method could be applied in a plug-and-play manner to enhance its performance with no additional modules or training process. Extensive experiments verify the effectiveness and generality of our method.

1 Introduction

Large language models (LLMs) have achieved impressive advancements across various tasks (Bang et al., 2023; Li et al., 2024) and applications (Zhao et al., 2024; Yuan et al., 2024) in recent years. However, LLMs still lack knowledge underrepresented in their training data, especially in up-to-date and domain-specific settings (Zhuang et al., 2024; Liu et al., 2024). To address these limitations, Retrieval-Augmented Generation (RAG) techniques have been widely adopted to retrieve external knowledge and enhance LLMs in diverse tasks, such as question-answering(Mansurova et al., 2024), information extraction(Glass et al., 2023), and dialogue systems (Wang et al., 2024a).

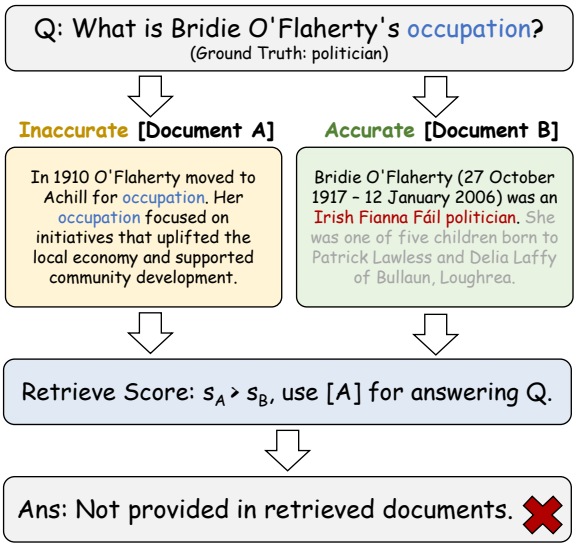

Figure 1: A document-based RAG process, where the red portion represents the fine-grained necessary information. Due to keyword matches (blue portion) in the inaccurate document and irrelevant sentences (gray portion) in the accurate document, the inaccurate document receives a higher retrieval score and is consequently selected, thereby leading to an incorrect answer.

Previous RAG methods can be broadly divided into two groups. The first group (Shi et al., 2024a; Yan et al., 2024) typically adopts single-round retrieval for one-hop questions, while the second group (Zhang et al., 2024; Shi et al., 2024b; Su et al., 2024) mostly employs multi-round retrieval for multi-hop questions. Both groups mostly retrieve knowledge documents based on document semantic similarity and use entire retrieved documents as input to help question answering. However, the fine-grained necessary information within documents is easily overlooked in this process, even though it is critical for question answering. Therefore, the retrieved knowledge is not always accurate enough to facilitate correct answers.

*Corresponding author: Zhendong Mao.

Findings of the Association for Computational Linguistics: ACL 2025, pages 10031–10044

Figure 1 illustrates a document-based RAG process. Due to the influence of keyword matches and irrelevant sentences, the inaccurate retrieved document appears more relevant to the query at the document level, despite its lack of fine-grained necessary information. Consequently, an incorrect answer is derived from the inaccurate retrieved document. This phenomenon naturally arises in most RAG pipelines and suggests that we could enhance this task from a fine-grained perspective.

In this work, we propose a Fine-grained Knowledge Enhancement method for RAG **(FKE)**. Our method contains a fine-grained knowledge retrieval paradigm and a decoding enhancement strategy, where the fine-grained knowledge primarily includes sentence-level information easily overlooked in the document-based retrieval process. Initially, we create a disentangled prompting procedure to retrieve fine-grained knowledge from the external knowledge corpus. In this procedure, the LLM is prompted to first explicitly identify the knowledge fragments beneficial for question answering and then extract the query-focused retrieval sentences as fine-grained knowledge based on these fragments. Subsequently, we propose to enhance the document-based decoding process with fine-grained retrieved knowledge. During the decoding process, the generator produces output distributions based on fine-grained retrieved sentences and original retrieved documents. We directly combine the fine-grained distribution with the document-based distribution and sample from the new distribution, where fine-grained retrieved sentences serve as sentence-level supervision to constrain the original distribution, making it more precise and helpful to the question.

Currently, a few methods filter the retrieved document and use only the extracted relevant sentences for question answering. These methods typically rely on additional annotations via GPT-4 (Asai et al., 2023; Yan et al., 2024) or automatic metrics (Wang et al., 2023), and design specific training methods to optimize the extra filter module. However, these complex annotation and extraction procedures often overlook whether the sentences are truly beneficial to question answering or merely semantically related to the question, so the quality of the extracted sentences remains a concern. Moreover, they primarily retain only the extracted sentences to assist answer generation, which may result in inaccurate outputs due to the lack of context-related knowledge. In contrast, our retrieval paradigm directly utilizes the existing generator model in a one-shot manner to retrieve the query-focused fine-grained knowledge, which is more helpful to answer generation. Besides, our decoding process retains the document-level retrieved knowledge to integrate context-relevant information and thereby achieve better performance. Given an existing RAG pipeline, our method could be applied **plug-and-play** to enhance its performance with **no additional modules or training process**.

In conclusion, our contributions are summarized as follows:

- We propose a disentangled fine-grained knowledge retrieval paradigm for the RAG field, where we collect the scarce fine-grained retrieved knowledge in the RAG area.
- We propose a decoding enhancement strategy that incorporates fine-grained knowledge to constrain the document-based decoding process and enable more precise generation.
- For a distinct comparison, we combine our method with two representative basic RAG pipelines from the two mainstream groups and conduct extensive experiments on four benchmark datasets. The results demonstrate the effectiveness and generality of our method.[1]

2 Approach

We propose a Fine-grained Knowledge Enhancement method for RAG **(FKE)**. In this method, we primarily create a **Fine-grained Knowledge Retrieval** paradigm to retrieve sentence-level fine-grained knowledge from the external corpus. Subsequently, we develop a **Decoding Enhancement** strategy to optimize the decoding process with retrieved fine-grained knowledge. Ultimately, these two components are integrated into the basic RAG pipeline to further enhance its performance. We will first briefly formulate the RAG technique and then elaborate on the two components.

2.1 Problem Formulation

Given a generative LLM $\mathcal{M}$, a natural language question q, and an external knowledge corpus $\mathcal{K}$, the RAG technique aims to retrieve relevant documents $\mathcal{D} = (d_1, d_2, ..., d_n)$ from $\mathcal{K}$, which enable

[1]Please email Jingxuan Han with your affiliation and a short description of how you will use our fine-grained knowledge and source code, and we will provide access to it.

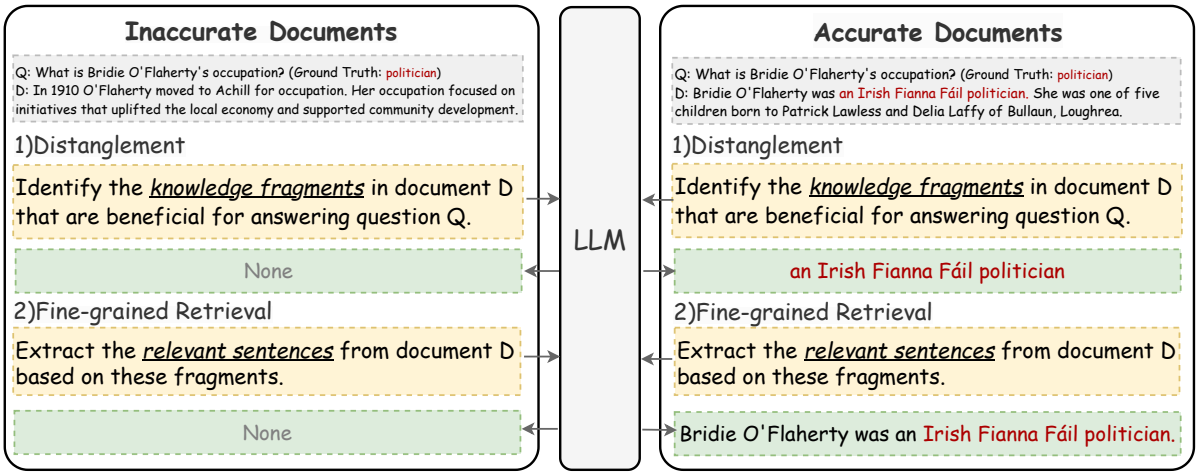

Figure 2: The disentangled prompting procedure to retrieve fine-grained knowledge. The gray box contains the question Q and the retrieved documents D. The yellow box represents the input prompt for guiding the LLM, while the green box depicts the LLM's output.

$\mathcal{M}$ to produce more accurate responses. The current RAG retrieval methods can be roughly categorized into three groups. The first group has used sparse retrieval methods that match query terms with document terms and compute relevance scores to select the most relevant documents. The second group has employed dense retrieval methods, which retrieve relevant documents based on the semantic similarity between queries and documents. The last group has attempted to leverage LLMs that distill the ranking capabilities of LLMs into their models to rank knowledge documents. Finally, single-hop questions are typically resolved with a single retrieval step, while multi-hop questions often require multiple retrieval steps.

2.2 Fine-grained Knowledge Retrieval

We create a fine-grained knowledge retrieval paradigm to obtain fine-grained information, which enhances the document-based process with additional sentence-level details. When solving a complicated problem, it is typical to decompose the problem into intermediate steps and solve each before giving final answers (Wei et al., 2022). Inspired by this, we develop a disentangled prompting procedure to retrieve fine-grained knowledge, where the LLM is first prompted to explicitly identify the knowledge fragments beneficial for question answering and then extract the query-focused sentences based on these fragments.

Figure 2 illustrates our disentangled prompting procedure for processing two types of documents from the external knowledge corpus $\mathcal{K}$. We assume the query q = *"What is Bridie O'Flaherty's occupation?"*, the inaccurate document d_1 = *"In 1910 O'Flaherty moved to Achill for occupation. Her occupation focused on initiatives that uplifted the local economy and supported community development."*, and the accurate document d_2 = *"Bridie O'Flaherty (27 October 1917 – 12 January 2006) was* ***an Irish Fianna Fáil politician****. She was one of five children born to Patrick Lawless and Delia Laffy of Bullaun, Loughrea."*. The d_2 contains the sentence-level details beneficial for answering q, whereas d_1 lacks the useful information. When employing our prompting procedure, the inaccurate document is discarded due to the absence of necessary knowledge fragments, while the fine-grained knowledge is extracted as query-focused sentences from the accurate document.

Specifically, we employ the existing generator model as LLM in a one-shot manner to implement our fine-grained retrieval paradigm on the external knowledge corpus $\mathcal{K}$. Ultimately, we obtain a set of query-focused sentences $\mathcal{S} = (s_1, s_2, ..., s_m)$ as retrieved fine-grained knowledge, where m is the number of query-focused sentences.

2.3 Decoding Enhancement

We develop a decoding enhancement strategy to enhance the document-based decoding process using the retrieved fine-grained knowledge. The overall architecture is illustrated in Figure 3. During the decoding stage, the generator produces two probability distributions: one based on fine-grained retrieved sentences and the other based on the original retrieved documents. The fine-grained distribution is then integrated to constrain

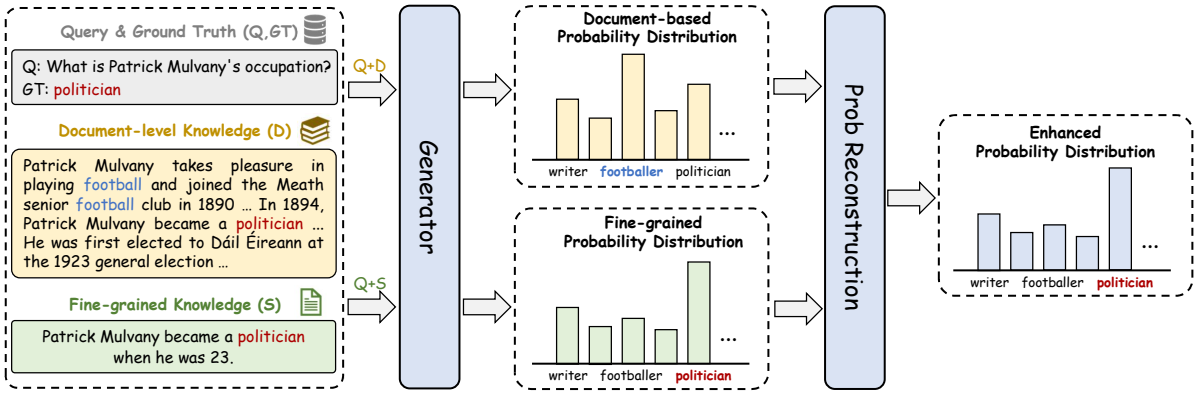

Figure 3: Decoding Enhancement Strategy. For each query, we incorporate additional fine-grained knowledge to reconstruct the document-based probability distribution.

the document-based distribution and produce the enhanced distribution, which enables the model to focus more on sentence-level fine-grained information and achieve better performance.

Fine-grained probability distribution For each question q and its corresponding fine-grained retrieved knowledge $\mathcal{S} = (s_1, s_2, ..., s_m)$, we enclose them into a specific generation template[2] T and the generator's input x_s is $T(q, S)$, where S is the concatenation of s_i. The generator θ then takes the template as input and generates the corresponding answer y_s in an auto-regressive manner. At each time step t, the generator θ compute the logits $z_{s,t}$ for the t-th token $y_{s,t}$:

$$z_{s,t} = \theta(x_s, y_{s,<t}) \tag{1}$$

The fine-grained probability distribution can be obtained by normalizing $z_{s,t}$:

$$p_\theta(y_{s,t}|x_s, y_{s,<t}) = \text{softmax}(z_{s,t}) \tag{2}$$

Then, the actual token $y_{s,t}$ in answer y_s is generated through certain sampling strategies:

$$y_{s,t} \sim p_\theta(y_{s,t}|x_s, y_{s,<t}) \tag{3}$$

When sampling from the fine-grained probability distribution, the generator will produce query-focused content that is more likely to yield helpful answers to the question.

Document-based probability distribution For each question q and its corresponding retrieved knowledge documents $\mathcal{D} = (d_1, d_2, ..., d_n)$, the generator's input x_d is $T(q, D)$, where D is the concatenation of d_i. At each time step t, the generator θ compute the logits $z_{d,t}$ for the t-th token $y_{d,t}$ and the document-based probability distribution can be obtained by normalizing $z_{d,t}$:

$$z_{d,t} = \theta(x_d, y_{d,<t}) \tag{4}$$

$$p_\theta(y_{d,t}|x_d, y_{d,<t}) = \text{softmax}(z_{d,t}) \tag{5}$$

Then, the actual token $y_{d,t}$ in answer y_d is generated through certain sampling strategies:

$$y_{d,t} \sim p_\theta(y_{d,t}|x_d, y_{d,<t}) \tag{6}$$

When sampling from the document-based probability distribution, the generator can integrate fragmented knowledge within documents, which consequently enables more precise answers in context-aware QA scenarios.

Enhanced probability distribution We integrate two types of distribution and obtain the enhanced distribution $p_\theta(y_t)$ based on Eq.2 and Eq.5:

$$p_\theta(y_t) = \frac{\text{softmax}(z_{d,t}/\tau_d) + \alpha\, \text{softmax}(z_{s,t}/\tau_s)}{1+\alpha}, \tag{7}$$

where α is the control strength and τ is the temperature. These two parameters can adjust different probability distributions for better coordination. Commonly, a token will only get a high probability if it has a high probability under both $p_\theta(y_t|x_d, y_{<t})$ and $p_\theta(y_t|x_s, y_{<t})$, where x_d and x_s represent the generator's inputs based on the original retrieved documents and the fine-grained retrieved sentences.

Sampling fluent output from language models commonly requires truncating the unreliable tail of the probability distribution (Liu et al., 2021), as in top-k (Fan et al., 2018) or nucleus sampling (Holtzman et al., 2019). Inspired by this, we truncate the

[2]For example, $T(q, k)$="*Generate an answer to question q based on the retrieved knowledge k.*"

logits $z_{d,t}$ and $z_{s,t}$ to obtain the fluent output:

$$\hat{z}_{d,t}[v], \hat{z}_{s,t}[v] = \begin{cases} z_{d,t}[v], z_{s,t}[v] & \text{if} \quad v \in \mathcal{V}' \\ -\infty, -\infty & \text{otherwise} \end{cases}, \quad (8)$$

where v denotes the candidate token and $\mathcal{V}'$ represents the set of tokens that are a part of the top-k vocabulary of the generator at time step t. By substituting $z_{d,t}, z_{s,t}$ with $\hat{z}_{d,t}, \hat{z}_{s,t}$ in Eq 7, the enhanced distribution $\hat{p}_\theta(y_t)$ is formulated as follows:

$$\hat{p}_\theta(y_t) = \frac{\text{softmax}(\hat{z}_{d,t}/\tau_d) + \alpha \, \text{softmax}(\hat{z}_{s,t}/\tau_s)}{1+\alpha} \quad (9)$$

Then, the final token y_t in answer y is generated through certain sampling strategies:

$$y_t \sim \hat{p}_\theta(y_t | x_d, x_s, y_{<t}) \quad (10)$$

By sampling from the enhanced probability distribution, the fine-grained knowledge can help the generator produce query-focused content that is more likely to yield helpful answers to the question, while document-level knowledge can integrate fragmented knowledge within different documents. These two types of knowledge are incorporated in a comparative decoding manner to achieve better performance.

3 Experimental Settings

3.1 Datasets

Following prior work on RAG, our method is evaluated on four datasets, including two MultihopQA datasets: **2WikiMultihopQA** (Ho et al., 2020), **HotpotQA** (Yang et al., 2018), and two SinglehopQA datasets: **PopQA** (Mallen et al., 2023), **ARC-Challenge** (Clark et al., 2018).

2WikiMultihopQA contains over 192k samples and provides evidence-based information with reasoning paths to evaluate a model's reasoning capabilities. It combines textual data from Wikipedia with structured knowledge from Wikidata, ensuring that the questions require multi-step reasoning.

HotpotQA consists of 113k Wikipedia-based question-answer pairs which require multi-hop reasoning across different documents. It comprises various question types and provides sentence-level supporting facts for explainable predictions.

PopQA includes 14k question-answer pairs that require long-tail Wikidata knowledge. It is constructed by sampling knowledge triples from Wikidata and focuses on less popular information.

ARC-Challenge is a multiple-choice reasoning dataset about daily commonsense science phenomena. It contains 2.5k natural science questions difficult for retrieval-based algorithms and word co-occurrence algorithms.

We use 1.0k test samples from both the 2WikiMultihopQA and HotpotQA datasets, along with 1.4k and 1.2k test samples from the PopQA and ARC-Challenge datasets, respectively.

3.2 Automatic Evaluation

Following previous works(Su et al., 2024; Yan et al., 2024), we adopt Exact Match (EM) score, F1 score and Accuracy as automatic evaluation metrics for the RAG task. The EM score measures the percentage of generated answers that exactly match the ground truth. The F1 score measures the token-level overlap between the predicted answer and the ground truth. The Accuracy score evaluates whether the generated answer contains the correct response. To facilitate comparison with the basic pipeline, we applied the same evaluation metrics to each dataset (Accuracy for the SinglehopQA dataset and EM, F1 for the MultihopQA dataset).

3.3 Implementation Details

We select two typical basic RAG pipelines (Yan et al., 2024; Su et al., 2024) to apply our method. DRAGIN (Su et al., 2024) utilizes multi-round retrieval for multi-hop questions, which optimizes the retrieval process based on the generator LLM's self-attention across its generated content. CRAG (Yan et al., 2024) adopts single-round retrieval for one-hop questions and incorporates a lightweight retrieval evaluator to improve the overall quality of retrieved documents for answer generation. As follows, we enhance basic RAG pipelines in retrieval and generation processes.

In retrieval, we utilize the existing generator model from the basic RAG pipeline in a one-shot manner to extract additional fine-grained knowledge from the external knowledge corpus $\mathcal{K}$. The number of fine-grained retrieved sentences m is set to 3, matching the number of retrieved documents n to ensure a fair comparison. Concretely, we first use a document-based method to retrieve 10 documents for each question from the external knowledge using Contriever (Izacard et al., 2021). Then we utilize the existing generator model (e.g., Llama2-7B-chat) to extract 3 fine-grained knowledge sentences from these documents, while the top 3 knowledge documents are considered the original

Multi-hop Methods	2WikiMulti		HotpotQA		Single-hop Methods	PopQA	ARC
	EM	F1	EM	F1		Acc	Acc
wo-RAG	14.6	22.3	18.4	27.5	$\text{LLaMA2}_{\text{7B}}$	38.2	48.0
SR-RAG	16.9	25.5	16.4	25.0	$\text{Alpaca}_{\text{7B}}$	46.7	48.0
FL-RAG	11.2	19.2	14.6	21.1	$\text{LLaMA2}_{\text{13B}}$	45.7	26.0
FS-RAG	18.9	26.5	21.4	30.4	$\text{Alpaca}_{\text{13B}}$	46.1	57.6
FLARE	14.3	21.3	14.9	22.1	Self-RAG	29.0	23.9
DRAGIN	21.4	29.3	23.2	31.2	CRAG	61.8	50.4
DRAGIN+FKE(Ours)	**24.2**	**32.5**	**28.1**	**34.7**	CRAG+FKE(Ours)	**68.1**	**55.8**

Table 1: The overall experimental results on two typical RAG pipelines across four datasets. The bold numbers indicate a better performance of pipeline+FKE than the corresponding pipeline alone.

document-level knowledge. In generation, we enhance the decoding process of basic RAG pipelines by incorporating the retrieved fine-grained knowledge. The temperature τ_d, τ_s and balancing parameter α are 0, 0.2 and 1.0, respectively. The process can be conducted on 1 NVIDIA A800 GPU. The detailed prompt and the introduction of two basic pipelines can be found in Appendix A.

4 Results and Analysis

4.1 Baselines

Based on the settings of DRAGIN (Su et al., 2024), we choose the following baselines for comparison in the Multi-hop QA task: (1)**wo-RAG**: LLM directly answers questions without using RAG. (2)**SR-RAG** (Single-round RAG): Relevant passages are retrieved from an external corpus based on the initial question and added to the LLM's input. (3)**FL-RAG** (Fix Length RAG) (Ram et al., 2023): The retrieval module is triggered every n tokens, and the tokens generated in the previous window are used as the query. (4)**FS-RAG** (Fix Sentence RAG): The retrieval module is triggered for every sentence, and the last generated sentence is used as the query. (5)**FLARE** (Jiang et al., 2023): The retrieval module is triggered by uncertain tokens, and the generated sentence without them is defined as the query.

Referring to CRAG's (Yan et al., 2024) settings, we select the following RAG baselines for the Single-hop QA task, including several public instruction-tuned LLMs: (6)**LLaMA2-7B**, (7)**LLaMA2-13B**, (8)**Alpaca-7B**, (9)**Alpaca-13B**. Additionally, we also incorporate (10)**Self-RAG** (Asai et al., 2023), a method that enhances the LLM's factuality through retrieval and self-reflection with the assistance of GPT-4 annotations.

4.2 Automatic Evaluation Result

We select DRAGIN (Su et al., 2024) and CRAG (Yan et al., 2024) as the basic RAG pipelines to apply our method. Since our method is applied in a plug-and-play manner to enhance these two basic pipelines, we **primarily focus on comparison with these two basic RAG pipelines**. The automatic evaluation results are presented in Table 1, which demonstrate the strong effectiveness and generality of our method.

Our method achieves significant improvements in MultihopQA tasks. Specifically, on the 2WikiMultihopQA dataset, FKE increases DRAGIN's EM score by **2.8** and F1 score by **3.2**. Similarly, on the HotpotQA dataset, FKE enhances DRAGIN's EM score by **4.9** and F1 score by **3.5**. These results highlight our method's specialized capability to address complex, multi-step reasoning scenarios.

Our method also performs well on SinglehopQA tasks. Concretely, FKE improves CRAG's accuracy by **6.3** on the PopQA dataset and **5.4** on the ARC dataset, demonstrating its ability to manage short-form entity generation (PopQA) and closed-set question answering (ARC) scenarios.

In summary, our proposed method achieves superior performance on both MultihopQA and SinglehopQA tasks, which indicates its robustness and effectiveness in enhancing the basic RAG pipeline with no additional modules or training process.

4.3 Ablation Study

To validate the effects of individual components in FKE, we conduct comprehensive ablation stud-

Knowledge	Length	Acc
CRAG	246517	67.0
Standard prompting	187653	69.0
Disentangled prompting	101406	75.2

Table 2: Ablation study of the disentangled prompting procedure on the PopQA dataset. Standard prompting and disentangled prompting refer to fine-grained knowledge retrieved by different prompting procedures, while CRAG represents document-level knowledge in the same quantity. **Length** refers to the count of tokens in knowledge content, and **Acc** indicates the probability that the retrieved knowledge contains the ground truth.

Methods	Acc
CRAG (with document-level knowledge)	61.8
CRAG (with fine-grained knowledge alone)	66.7
CRAG (with comparative decoding strategy)	68.1

Table 3: Ablation study of fine-grained knowledge.

ies on specific datasets and these results remain consistent when applied to other datasets.

Ablation of disentangled prompting procedure To validate the effectiveness of our disentangled prompting retrieval procedure, we evaluate the retrieved knowledge under different conditions. Under the standard prompting condition, the LLM is directly prompted to extract relevant sentence-level knowledge from the external corpus. In contrast, with disentangled prompting, the LLM first explicitly identifies knowledge fragments beneficial for question answering and then extracts the query-focused sentences based on these fragments. The results are presented in Table 2. Although the standard prompting procedure improves knowledge accuracy and shortens the original document-level knowledge to some extent, it remains less effective than our disentangled prompting procedure. Therefore, we require the disentangled prompting procedure for better performance.

Ablation of fine-grained knowledge We have directly used fine-grained knowledge alone for generation and the results on the PopQA dataset are shown in Table 3. Using fine-grained knowledge alone can bring better results than using original document-level knowledge, which demonstrates that fine-grained knowledge contains more accurate information. Combining the two types of knowledge with the comparative decoding strategy, the model achieves the best performance by benefiting from the strengths of both.

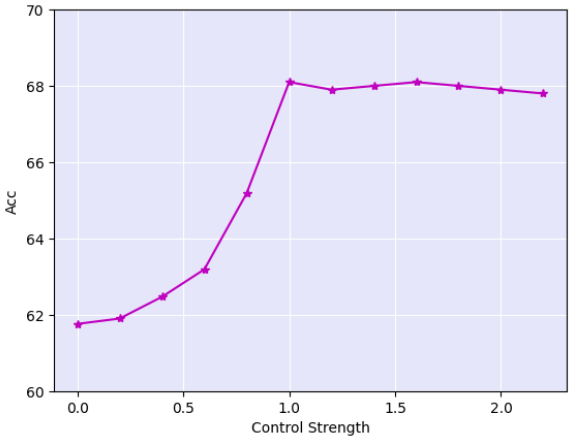

Figure 4: Accuracy score of CRAG+FKE with different control strength α on the PopQA dataset.

Methods	EM	F1
DRAGIN(7B)	21.4	29.3
DRAGIN(7B)+FKE	24.2	32.5
DRAGIN(13B)	26.0	33.3
DRAGIN(13B)+FKE	28.0	36.2

Table 4: Ablation study of the model scale on the 2WikiMultihopQA dataset.

Ablation of control strength To investigate the impact of control strength α, we employ the CRAG pipeline and conduct experiments on the PopQA dataset with different α values. The Accuracy variation curve is depicted in Figure 4. When $\alpha = 0$, the fine-grained knowledge is not incorporated, and the model's performance aligns with the basic pipeline. As α increases within a certain range ($0 \sim 1.0$), the decoding process incorporates more fine-grained information, and the model's performance improves accordingly. Ultimately, when $\alpha > 1.0$, further improvement is no longer observed, as fine-grained knowledge has already been fully incorporated into the document-level decoding process. Based on this, we set $\alpha = 1.0$ for optimal performance.

Ablation of model scale To explore the effect of the model scale, we additionally use Llama2-13B-chat as DRAGIN's generator and conduct experiments on the 2WikiMultihopQA dataset with other hyperparameters unchanged. The results in Table 4 show that our method consistently enhances DRAGIN's performance across different model scales.

Ablation of temperature To evaluate the influence of the temperature, we conduct experiments on the PopQA dataset with different temperatures

Methods	CRAG	CRAG+FKE (Ours)
Query 1	What is Bedřich Feigl's occupation?	
Retrieved Knowledge	Feigl studied at the Prague Academy of Fine Arts with Vlaho Bukovac and Františka Thieleho. ……. In Berlin he became familiar with the art of Max Liebermann. In 1907 he attended the first exhibition in Prague Group Eight. ……He fled Prague in 1939 and settled in London, with his wife, where he died in 1965. His works are placed in galleries around the world.	Bedřich Feigl (also known as Friedrich Feigl; March 6, 1884 – 17 December 1965) was a Czech-Jewish painter and illustrator.
Answer	It is not provided in the given documents.	Bedřich Feigl's occupation is a painter and illustrator.
Query 2	What is Henry Feilden's occupation?	
Retrieved Knowledge	Henry Master Feilden (21 February 1818 – 5 September 1875) was an English Conservative Party politician.....His love of architecture was inherited from his grandfather, Brightwen Binyon (1846-1905), an Ipswich architect and former pupil of Alfred Waterhouse. …….He set up an architectural practice with David Mawson in 1956, to which offices in Norwich, London and Cambridge were later added.	Henry Master Feilden (21 February 1818 – 5 September 1875) was an English Conservative Party politician.
Answer	Henry Feilden's occupation is an architect.	Henry Feilden's occupation is a Conservative Party politician.

Table 5: Case Study. The red portion indicates the necessary information, while the blue portion represents the misleading information. CRAG's answers are incorrect due to the absence of necessary information and the influence of misleading information. Our method retrieves additional sentence-level knowledge, which contains the fine-grained necessary information for question answering and assists CRAG in generating more accurate outputs.

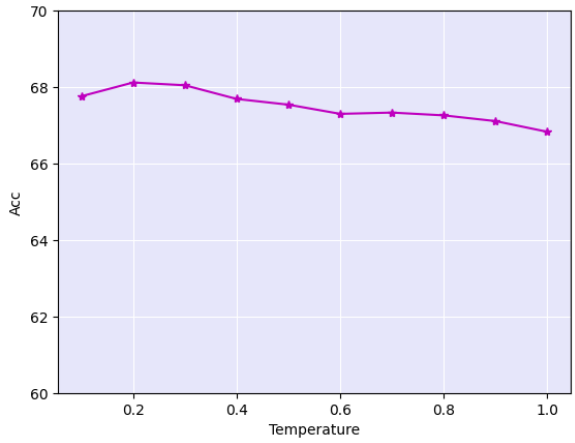

Figure 5: Accuracy score of CRAG+FKE with different temperature τ_s on the PopQA dataset.

τ_s while keeping τ_d constant to stabilize the basic pipeline's performance in Table 1. Figure 5 illustrates the variation curve of the Acc score, which indicates that our method provides a stable enhancement to the basic RAG pipeline (CRAG, Acc=61.8). Accordingly, we choose temperature $\tau_s = 0.2$ as it yields the best results.

4.4 Case Study

To better understand our method, we select the CRAG pipeline and sample several generated answers from the PopQA dataset shown in Table 5. For query 1, the retrieved document of CRAG is semantically relevant but lacks the fine-grained necessary information for question answering, preventing the generation of accurate answers. For query 2, although CRAG retrieves the correct document containing the necessary information for question answering, the generated answer remains inaccurate due to the misleading information in the document. In contrast, our method retrieves additional sentence-level knowledge that contains the fine-grained necessary information, enabling the basic pipeline to generate more accurate answers.

5 Related Work

Retrieval-Augmented Generation Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) can effectively alleviate the hallucination caused by knowledge gaps in large language models (LLMs) (Huang et al., 2023). Existing RAG methods can be roughly divided into two groups.

The first group typically performs single-round retrieval from an external corpus, which is particularly effective for questions requiring explicit information. Self-RAG (Asai et al., 2023) trains an arbitrary language model (LM) to adaptively retrieve passages and generate content on demand using special reflection tokens. REPLUG (Shi et al., 2024a) leverages likelihood scores from the LLM to fine-tune the retriever, which enables it to retrieve more relevant and useful documents. CRAG (Yan et al., 2024) evaluates the quality of retrieved documents using a lightweight evaluator and enhances retrieval results accordingly through large-scale web searches.

The second group iteratively extracts relevant knowledge during the generation process, making it especially suitable for multi-hop questions and complex queries. DRAGIN (Su et al., 2024) dynamically determines when and what to retrieve based on self-attention weights across its generated content. GenGround (Shi et al., 2024c) iteratively decomposes the multi-hop question into sub-questions and grounds their answers in retrieved documents to correct potential errors until the final answer is derived. ReSP (Jiang et al., 2024) employs query-focused summarization with an LLM-based summarize to mitigate the context overload problem caused by multiple rounds of retrieval.

Fine-grained RAG Method Recently, several RAG studies have attempted to develop their frameworks in a fine-grained manner. FILCO (Wang et al., 2023) trains a context filtering model to extract relevant context from documents using lexical and information-theoretic approaches. ConvRAG (Ye et al., 2024) introduces a fine-grained retriever that performs document-level retrieval and paragraph-level reranking for conversational question answering. REAR (Wang et al., 2024b) estimates fine-grained relevance scores using lexical and semantic similarity, which ultimately serve as supervision for training the generator model. GeAR (Liu et al., 2025) trains a text decoder to generate fine-grained information from the fused representation of the query and retrieved documents. In contrast, our method operates plug-and-play within existing RAG pipelines, enhancing performance without additional modules or training processes.

6 Conclusion

In this paper, we propose a novel fine-grained knowledge enhancement method for retrieval-augmented generation (RAG) tasks. Unlike previous methods primarily relying on document-level retrieved knowledge, we extract and utilize fine-grained knowledge to enhance RAG performance. Concretely, we design a disentangled prompting procedure to retrieve fine-grained knowledge and develop a decoding enhancement strategy to constrain the document-based decoding process with fine-grained knowledge. Our method requires no additional modules or training process and can be widely integrated into basic RAG pipelines. Experiments on two mainstream RAG pipelines and four benchmark datasets demonstrate the efficacy and generality of our proposed method.

Limitations

Since RAG basic pipelines based on larger-scale LLMs (e.g., 70B) require extensive computational resources and are not the mainstream methods, we have not yet evaluated our method in this setting. Moreover, on rare occasions when the external knowledge corpus lacks the necessary information for question answering, our method will not provide significant enhancements in these cases.

Ethics Statement

The RAG technique helps reduce factual errors in LLM-generated content. However, since the external corpus comes from various sources, the development of RAG may also introduce potential risks. For example, if attackers add biased or discriminatory content to the external knowledge corpus, such statements may be generated. Moreover, if the corpus contains incorrect or outdated information, it could lead to the spread of fake news. Additionally, RAG could be misused to create false information that harms reputations or manipulates public opinion. Therefore, as RAG technology advances, it is important to carefully manage the corpus and ensure its use follows ethical and legal standards to minimize potential risks. In this work, we rigorously adhere to the human-centered principle to ensure the responsible development and application of RAG technologies. The datasets utilized in this work do not contain any personal privacy information or offensive content, and no personal data has been collected, minimizing the potential risks associated with RAG.

Acknowledgements

We would like to express our sincere gratitude to the professional reviewers for their suggestions and comments. This work is supported by the National Science Fund for Excellent Young Scholars under Grant 62222212 and the National Natural Science Foundation of China under Grant 62376033.

References

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

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei

Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 675–718.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics.

Michael Glass, Xueqing Wu, Ankita Rajaram Naik, Gaetano Rossiello, and Alfio Gliozzo. 2023. Retrieval-based transformer for table augmentation. *arXiv preprint arXiv:2306.11843*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992.

Zhouyu Jiang, Mengshu Sun, Lei Liang, and Zhiqiang Zhang. 2024. Retrieve, summarize, plan: Advancing multi-hop question answering with an iterative approach. *arXiv preprint arXiv:2407.13101*.

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.

Mingchen Li, Halil Kilicoglu, Hua Xu, and Rui Zhang. 2024. Biomedrag: A retrieval augmented large language model for biomedicine. *arXiv preprint arXiv:2405.00465*.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. 2021. Dexperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706.

Haoyu Liu, Shaohan Huang, Jianfeng Liu, Yuefeng Zhan, Hao Sun, Weiwei Deng, Feng Sun, Furu Wei, and Qi Zhang. 2025. Gear: Generation augmented retrieval. *arXiv preprint arXiv:2501.02772*.

Yanming Liu, Xinyue Peng, Xuhong Zhang, Weihao Liu, Jianwei Yin, Jiannan Cao, and Tianyu Du. 2024. Ra-isf: Learning to answer and understand from retrieval augmentation via iterative self-feedback. *arXiv preprint arXiv:2403.06840*.

Alex Troy 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 *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Aigerim Mansurova, Aiganym Mansurova, and Aliya Nugumanova. 2024. Qa-rag: Exploring llm reliance on external knowledge. *Big Data and Cognitive Computing*, 8(9):115.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024a. Replug: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8364–8377.

Yucheng Shi, Qiaoyu Tan, Xuansheng Wu, Shaochen Zhong, Kaixiong Zhou, and Ninghao Liu. 2024b. Retrieval-enhanced knowledge editing in language

models for multi-hop question answering. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2056–2066.

Zhengliang Shi, Shuo Zhang, Weiwei Sun, Shen Gao, Pengjie Ren, Zhumin Chen, and Zhaochun Ren. 2024c. Generate-then-ground in retrieval-augmented generation for multi-hop question answering. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7339–7353, Bangkok, Thailand. Association for Computational Linguistics.

Weihang Su, Yichen Tang, Qingyao Ai, Zhijing Wu, and Yiqun Liu. 2024. Dragin: Dynamic retrieval augmented generation based on the real-time information needs of large language models. *arXiv preprint arXiv:2403.10081*.

Hongru Wang, Wenyu Huang, Yang Deng, Rui Wang, Zezhong Wang, Yufei Wang, Fei Mi, Jeff Z Pan, and Kam-Fai Wong. 2024a. Unims-rag: A unified multi-source retrieval-augmented generation for personalized dialogue systems. *arXiv preprint arXiv:2401.13256*.

Yuhao Wang, Ruiyang Ren, Junyi Li, Xin Zhao, Jing Liu, and Ji-Rong Wen. 2024b. REAR: A relevance-aware retrieval-augmented framework for open-domain question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5613–5626, Miami, Florida, USA. Association for Computational Linguistics.

Zhiruo Wang, Jun Araki, Zhengbao Jiang, Md Rizwan Parvez, and Graham Neubig. 2023. Learning to filter context for retrieval-augmented generation. *arXiv preprint arXiv:2311.08377*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

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

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Linhao Ye, Zhikai Lei, Jianghao Yin, Qin Chen, Jie Zhou, and Liang He. 2024. Boosting conversational question answering with fine-grained retrieval-augmentation and self-check. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2301–2305.

Dong Yuan, Eti Rastogi, Gautam Naik, Sree Prasanna Rajagopal, Sagar Goyal, Fen Zhao, Bharath Chintagunta, and Jeffrey Ward. 2024. A continued pretrained llm approach for automatic medical note generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 565–571.

Jiahao Zhang, Haiyang Zhang, Dongmei Zhang, Liu Yong, and Shen Huang. 2024. End-to-end beam retrieval for multi-hop question answering. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1718–1731.

Andrew Zhao, Daniel Huang, Quentin Xu, Matthieu Lin, Yong-Jin Liu, and Gao Huang. 2024. Expel: Llm agents are experiential learners. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19632–19642.

Ziyuan Zhuang, Zhiyang Zhang, Sitao Cheng, Fangkai Yang, Jia Liu, Shujian Huang, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, and Qi Zhang. 2024. Efficientrag: Efficient retriever for multi-hop question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 3392–3411.

A Appendix

A.1 Disentangled Implementation Appendix

Due to limited computational resources, we cannot directly retrieve fine-grained knowledge from the external knowledge corpus. Therefore, we first use a document-based method to retrieve a set of documents and then extract fine-grained knowledge from them. Specifically, we retrieve 10 documents for each question from the external knowledge corpus using Contriever (Izacard et al., 2021). Then, we utilize the existing generator model (Llama2-7B-chat) to extract 3 fine-grained knowledge sentences from these documents, while the top 3 knowledge documents are considered the original document-level knowledge. The detailed prompt is shown in Figure 6.

A.2 DRAGIN Implementation Appendix

DRAGIN (Su et al., 2024) consists of two components: Real-time Information Needs Detection (RIND) and Query Formulation based on Self-attention (QFS). Once the RIND module identifies position i requiring external knowledge, the QFS module formulates a query and utilizes an existing retrieval model to obtain relevant documents from external knowledge bases. The LLM's output up to

Disentangled Prompt

Your task is to extract useful sentences from a document based on the knowledge fragments needed to answer the specific question. You should first identify the knowledge fragments beneficial for answering the question, and then extract the relevant sentence based on the knowledge fragments above. Output a new text composed of these relevant sentences.

For example:
Question: {What is Jim Brown's occupation?}
Document: {James Nathaniel Brown (born February 17, 1936) is a former American football player, sports analyst and actor. He played as a fullback for the Cleveland Browns of the National Football League (NFL) from 1957 through 1965. Considered to be one of the greatest running backs of all time, as well as one of the greatest players in NFL history, Brown was a Pro Bowl invitee every season he was in the league, was recognized as the AP NFL Most Valuable Player three times, and won an NFL championship with the Browns in 1964. He led the league in rushing yards in eight out of his nine seasons, and by the time he retired, he had shattered most major rushing records. In 2002, he was named by 'The Sporting News' as the greatest professional football player ever.}.
The expected output should be: {James Nathaniel Brown (born February 17, 1936) is a former American football player, sports analyst and actor.}.

Now, the question is {}, and the document is {}.
The output should be:

Figure 6: The detailed prompt of the disentangled procedure.

position i is preserved and the retrieved documents are integrated with the preserved output using a meticulously designed prompt template. The LLM then continues generating content based on the new input. This process repeats until the question is fully addressed. **Our method** uses the formulated query generated by the QFS module to retrieve fine-grained knowledge, which is then applied as an additional input to enhance the LLM's generation process, ultimately leading to better results. The details of the RIND and QFS components are introduced as follows.

RIND For any given token t_i in the generated sequence $T = \{t_1, t_2, ..., t_n\}$, RIND quantifies the uncertainty by computing entropy $\mathcal{H}_i$ as follows:

$$\mathcal{H}_i = -\sum_{v \in \mathcal{V}} p_i(v) \log p_i(v), \tag{11}$$

where $p_i(v)$ denotes the probability of generating the token v over all tokens in the vocabulary $\mathcal{V}$ at position i.

In addition, RIND quantifies the impact of token t_i on the subsequent context by leveraging the attention value $A_{i,j}$ between t_i and t_j ($i < j$), which is computed as follows:

$$A_{i,j} = \text{softmax}\left(\frac{Q_i K_j^{\mathrm{T}}}{\sqrt{d_k}}\right), \tag{12}$$

where Q_i represents the query vector of token t_i, K_j represents the key vector of token t_j, and d_k denotes the dimensionality of the key vector. Following this, the maximum attention value $a_{\max}(i)$ of token t_i is identified as follows:

$$a_{\max}(i) = \max_{j>i} A_{i,j} \tag{13}$$

Moreover, for concentrating on tokens with significant semantic value, RIND employs a binary semantic indicator s_i to indicate whether a word belongs to the stopwords set $\mathcal{S}$:

$$s_i = \begin{cases} 0, & \text{if } t_i \in \mathcal{S} \\ 1, & \text{otherwise} \end{cases} \tag{14}$$

Finally, combining uncertainty, impact, and semantics, RIND computes a comprehensive score for each token t_i:

$$\mathcal{S}_{RIND}(t_i) = \mathcal{H}_i \cdot a_{\max}(i) \cdot s_i \tag{15}$$

When $\mathcal{S}_{RIND}(t_i)$ exceeds a predefined threshold θ, the retrieval module will be activated to retrieve external knowledge.

QFS The attention scores reflect the self-assessed importance of each token in generating token t_i. Therefore, QFS formulates the query by selecting the most important preceding tokens through attention scores. This process includes: (1) Extract the attention scores of the last Transformer layer; (2) Sort the scores in descending order to identify the top n tokens; (3) Find the words corresponding to these tokens from the vocabulary and arrange them according to their original order in the text; (4) Construct the query Q_i using these words.

A.3 CRAG Theoretical Appendix

CRAG (Yan et al., 2024) introduces retrieval corrective strategies to improve the robustness of text generation. Given an input query q and retrieved documents $\mathcal{D} = (d_1, d_2, ...d_n)$ from any retriever, a lightweight retrieval evaluator categorizes the documents into one of three confidence levels: {`Correct`, `Incorrect`, `Ambiguous`}. Depending on the confidence level, different knowledge retrieval actions are triggered to optimize the retrieval results and obtain the corrective retrieval results $\mathcal{M} = (m_1, m_2, m_3)$. Finally, an arbitrary generative model is used to answer the question based on the corrective retrieval results. **Our method** employs the input query q and retrieved documents $\mathcal{D} = (d_1, d_2, ...d_n)$ to extract fine-grained knowledge $\mathcal{S} = (s_1, s_2, s_3)$, which is then applied as an additional input to enhance the arbitrary model's generation process. The details of the retrieval evaluator and actions are provided as follows.

Retrieval Evaluator The retrieval evaluator predicts the relevance score for each question-document pair independently. CRAG initializes the evaluator using the T5-large (Raffel et al., 2020) pre-trained language model. Relevance signals for fine-tuning can be obtained from existing datasets (e.g., wiki titles). Negative samples for fine-tuning are randomly selected from retrieval results that are similar but not relevant to the input query. The confidence level of the retrieval is assigned based on the relevance scores from the fine-tuned evaluator. A retrieval is considered `Correct` if the relevance score exceeds the upper threshold. Conversely, a retrieval is classified as `Incorrect` if the relevance scores fall below the lower threshold. If neither condition is met, the retrieval is labeled as `Ambiguous`.

Knowledge Retrieval Actions Based on the fine-tuned evaluator, two actions are implemented to enhance the retrieval results: Knowledge Refinement and Web Search.

Knowledge Refinement focuses on extracting the most relevant knowledge from the retrieved documents. Each relevant document is divided into smaller strips, typically consisting of a few sentences, based on the document's total length. The retrieval evaluator is then used to calculate the relevance score between the question and each strip, filtering out any irrelevant ones.

Web Search first converts the inputs into search queries using ChatGPT. A public web search API is then employed to return a series of web pages for each query. Since knowledge from web pages may contain biases or unreliable information, authoritative sources like Wikipedia are prioritized. The same knowledge refinement method is applied to extract relevant information from these pages.

If the confidence is `Correct`, the knowledge from the documents is considered reliable, and Knowledge Refinement is used to extract the most important information. If the confidence is `Incorrect`, it means all retrieved documents are irrelevant, so Web Search is used to obtain external knowledge. In the case of `Ambiguous`, where the accuracy is uncertain, both actions are combined to complement each other.

A.4 Human Evaluation Appendix

In addition to automatic evaluation, we also incorporate human evaluation to evaluate our method and two basic RAG pipelines. Specifically, we randomly sampled 200 outputs from each dataset, resulting in a total of 400 outputs per basic pipeline. Three annotators independently evaluated the responses, assigning scores from 1 (Very Bad) to 5 (Very Good) based on their correctness (C) and relevance (R) in answering the questions. Considering the difference between the two datasets, annotators will get \$0.1 for each answer in the Singlehop dataset and \$0.2 for each answer in the Multihop dataset. There are 800 sentences evaluated, so each annotator was rewarded \$120 in total.

The human evaluation results are shown in Table 6, which are generally consistent with the automatic evaluation, further confirming the capability of our method. Specifically, the higher correctness score indicates that our answers align better with objective facts, while the greater relevance score reflects the improved connection of our answer with

Methods	2WikiMulti		HotpotQA	
	C	R	C	R
DRAGIN	3.2	3.9	3.4	3.5
DRAGIN+FKE	**3.6**	**4.0**	**3.7**	**3.6**
Methods	PopQA		ARC	
	C	R	C	R
CRAG	3.8	3.7	3.1	3.4
CRAG+FKE	**4.2**	**3.8**	**3.2**	**3.6**

Table 6: Human evaluation result. Each score represents the average score of three annotators, where C, R represent correctness and relevance, respectively.

Methods	2WikiMulti		HotpotQA	
	C	R	C	R
DRAGIN	0.781	0.724	0.753	0.719
DRAGIN+FKE	0.799	0.735	0.769	0.714
Methods	PopQA		ARC	
	C	R	C	R
CRAG	0.792	0.733	0.769	0.722
CRAG+FKE	0.802	0.747	0.779	0.715

Table 7: The inter-annotator agreement score for human evaluation. The C, R represent correctness and relevance, respectively.

both the retrieved knowledge and the question.

Moreover, we calculate the Fleiss' Kappa coefficient to measure the inter-annotator agreement score for each human evaluation metric. Fleiss' Kappa coefficient is a statistical measure used to assess the reliability or agreement between multiple raters or annotators when they are classifying items into categories. The results are shown in Table 7. The high consistency among human annotators confirms the reliability of our human evaluation.