TAIKC: Tree-Organized Active Internal Knowledge Completion for Multi-Hop Question Answering

Anonymous ACL submission

Abstract

Iterative dynamic Retrieval-Augmented Generation (RAG) methods have demonstrated strong performance on Multi-Hop Question Answering (MHQA). However, they still suffer from high inference costs, redundant information processing, and retrieval decisions that depend heavily on internal states. To this end, we propose **T**ree-Organized **A**ctive **I**nternal **K**nowledge **C**ompletion **(TAIKC)**, a novel approach designed to address two significant challenges: efficient information aggregation and active retrieval decision-making. TAIKC hierarchically decomposes multi-hop questions into a tree of sub-questions. For each sub-question, the model either extracts confident internal knowledge based on its perception of knowledge boundaries or leverages external knowledge to fill the knowledge gap. This process incrementally constructs a knowledge tree that integrates both internal and external information, and knowledge chains are then induced from the knowledge tree to solve the complex question. Furthermore, we align the model with our framework via knowledge distillation and model bootstrapping. Extensive experiments on four MHQA datasets demonstrate the effectiveness of our method.

1 Introduction

RAG mitigates the limitations of Large Language Models (LLMs) (Brown et al., 2020; Ouyang et al., 2022; Touvron et al., 2023) in handling long-tail and temporal knowledge (Wang et al., 2025; Li et al., 2024) by incorporating external information into the generation process (Xu et al., 2025; Wei et al., 2025; Asai et al., 2024). This enables LLMs to maintain high-quality responses in environments where knowledge is continually evolving.

Traditional methods of RAG typically retrieve relevant information from external corpora in a single-pass manner based on the initial query (Zhuang et al., 2024; Yu et al., 2024), which performs well on relatively simple tasks. However, such methods often struggle in multi-hop question answering due to difficulty integrating multiple relevant passages for reasoning (Chu et al., 2024; V et al., 2025; Cao et al., 2023). To address this, *iterative multi-round retrieval* frameworks have been proposed. These approaches leverage intermediate outputs in the current step (e.g., reasoning steps and sub-questions) as queries for the subsequent retrieval round (Su et al., 2024; Lyu et al., 2024; Press et al., 2023), thereby incorporating task-relevant external information into the generation process progressively and iteratively (Jin et al., 2025; Yao et al., 2023b). Furthermore, to improve retrieval efficiency and generation quality, some studies have incorporated *dynamic retrieval mechanisms*. These methods dynamically determine when and what to retrieve based on model-internal signals (Su et al., 2024; Jiang et al., 2023), such as token probabilities and self-attention weights.

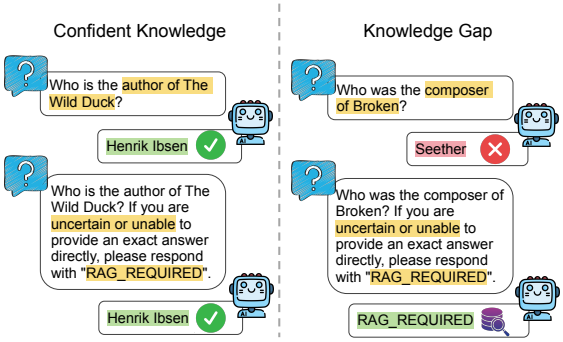

Figure 1: A comparison between confident knowledge and knowledge gap. Prompt can be found in Table 17.

Although iterative dynamic RAG methods demonstrate strong reasoning capabilities in complex problem solving, they significantly increase inference overhead due to generating and accumulating extensive intermediate information. As the context length grows, the model's ability to locate key information deteriorates, negatively impacting generation quality. In addition, their reliance on internal signals such as attention weights limits their

applicability in proprietary models, where such signals are typically inaccessible. To address these challenges, this work focuses on three research questions: (1) **RQ1**: How can information be efficiently aggregated to solve complex problems? (2) **RQ2**: How can models make effective active retrieval decisions? (3) **RQ3**: How can the proposed framework better adapt to the model?

To address **RQ1**, we decompose multi-hop questions into hierarchical sub-questions, forming a tree-structured representation where each node corresponds to a sub-question. By sequentially solving all nodes (*through the solution for RQ2*), we construct a Knowledge Tree supported by both internal and external knowledge sources. We then summarize the branches of this tree to generate multiple Knowledge Chains, which facilitate reasoning and the final resolution of complex questions. For **RQ2**, for each sub-question, based on the model's ability to perceive its knowledge boundaries (i.e., to determine whether it knows the answer to a given question; see Appendix C for details), we use the Confident QA (see Table 17) approach to extract high-confidence internal knowledge from the model, referred to as Confident Knowledge (see Figure 1). For sub-questions that the model cannot answer directly, we supplement them with external knowledge via RAG. Finally, to address **RQ3**, we enhance the model's instruction-following capability and knowledge boundary awareness through knowledge distillation and model bootstrapping, thereby achieving alignment between the framework and the model.

Our main contributions are as follows: (1) We propose a hierarchical reasoning framework that aggregates information and generates high-quality reasoning chains, thereby improving both the efficiency and quality of solving complex tasks; (2) We enable the model to actively determine when to invoke external knowledge, reducing reliance on internal signals; (3) We strengthen the model's ability to follow instructions and recognize the boundaries of its knowledge, allowing the framework to be effectively adapted to different models.

2 Related work

2.1 Multi-hop Question Answering

Multi-hop question answering aims to address questions that require reasoning over multiple knowledge passages and performing multi-step inference (Zhang et al., 2024; Yang et al., 2018). Early approaches leveraged the reasoning capabilities of LLMs, first generating the reasoning process and then producing the final answer (Wei et al., 2022; Yao et al., 2023a). Building on this, subsequent work introduced multi-turn interactions between the retriever and the reader, incorporating external documents into the reasoning process to reduce hallucination and using intermediate reasoning results to guide subsequent retrieval, thereby enhancing knowledge completeness and coherence (Xu et al., 2024b; Trivedi et al., 2023; Khattab et al., 2023). Additionally, some studies employed the decomposition ability of LLMs to iteratively break down complex questions into simpler sub-questions until the final answer was derived (Shi et al., 2024; Press et al., 2023). More recently, the emergence of large reasoning models (DeepSeek-AI et al., 2025; OpenAI et al., 2024b) has introduced a new paradigm for tackling complex problems, owing to their strong and sophisticated reasoning capabilities (Jin et al., 2025; Song et al., 2025).

2.2 Retrieval-Augmented Generation

Retrieval-augmented generation improves LLMs by integrating external knowledge (Yue et al., 2025; Lewis et al., 2020). Early studies proposed a *retrieve-then-read* architecture, in which documents relevant to the input query are first retrieved and then used by a generation module to produce answers (Tan et al., 2024; Asai et al., 2024). To facilitate better coordination between internal and external knowledge, some works leverage token probabilities to decide when to incorporate external knowledge via retrieval (Su et al., 2024; Jiang et al., 2023). Subsequently, to further improve retrieval effectiveness and generation quality, techniques such as query rewriting (Mao et al., 2024; Wang et al., 2023) and document re-ranking (Chen et al., 2024) have been introduced into the RAG pipeline. Moreover, given that retrievers may return irrelevant information, several studies attempt to filter out unrelated documents by assessing document-query relevance (Yoran et al., 2024; Liu et al., 2024), or extract useful information from large texts via document summarization and compression methods (Li et al., 2025; Yoon et al., 2024; Xu et al., 2024a). Other lines of work enhance RAG performance by improving the quality of the offline knowledge base through the integration of knowledge graphs and related techniques during its construction (Zhang et al., 2025; Gutierrez et al., 2024; Edge et al., 2025).

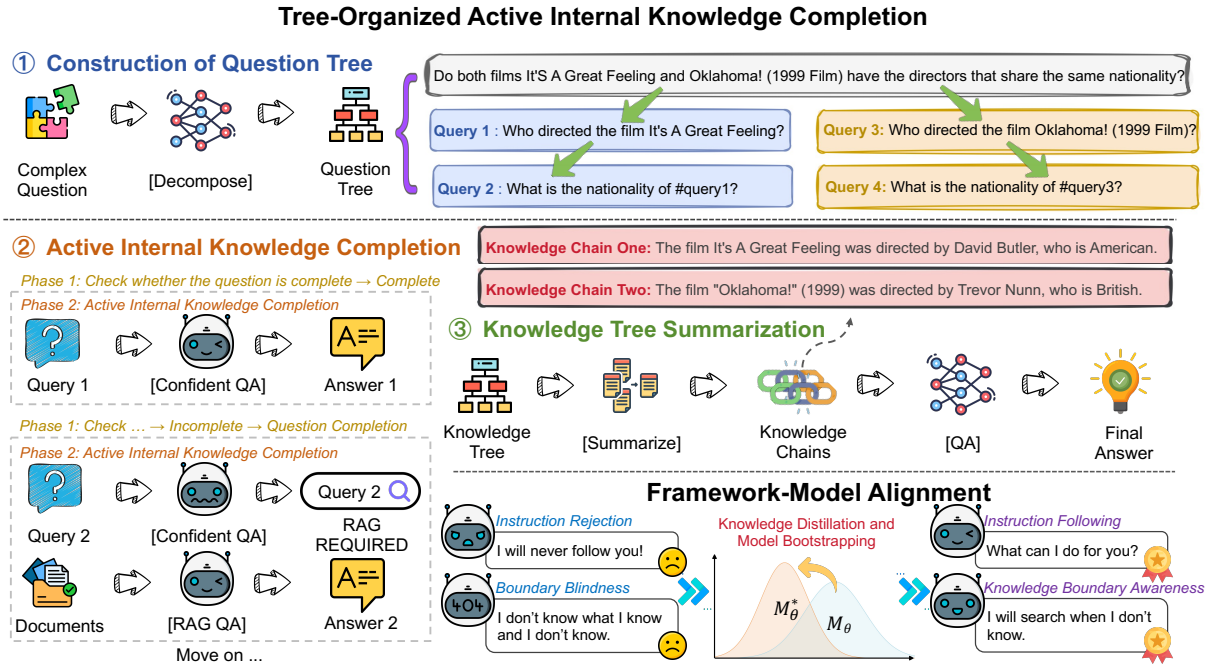

Figure 2: The overview of the framework we proposed. [Action] represents the interaction with the LLM.

3 TAIKC: Tree-Organized Active Internal Knowledge Completion

Here is an overview of our method. TAIKC decomposes multi-hop questions into a sub-question tree and processes them node by node. The model prioritizes the use of internal knowledge for answering, while dynamically incorporating external knowledge when necessary to fill in gaps. It then constructs a logically coherent knowledge chain along the complete knowledge tree to support reasoning and answering complex questions. The overall framework is illustrated in Figure 2.

3.1 Construction of Question Tree

Multi-hop questions often exhibit complex structures such as bridge, comparison, or their combinations. To effectively represent these structures, we introduce a tree-based framework to parse and model complex questions. Specifically, we construct each question $\mathcal{Q}$ into a question tree $\mathcal{T}$, where the root node q_0 represents the original complex question, and all intermediate and leaf nodes correspond to sub-questions:

$$\begin{aligned} \mathcal{Q} \rightarrow \mathcal{T} &= (\mathcal{V}, \mathcal{E}) \\ \mathcal{V} = \{q_0, q_1, \ldots, q_n\}, &\quad \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V} \end{aligned} \tag{1}$$

Following prior work, we adopt the QDMR (Chu et al., 2024; Wolfson et al., 2020) format to represent the decomposed questions. In practice, we use placeholders of the form #queryX to refer to the answers of preceding sub-questions. These placeholders are later resolved during a traversal process by retrieving and substituting the corresponding sub-question answers, thereby incrementally constructing the complete semantic chain of the original question. As illustrated in Figure 2, this tree structure clearly reveals the hierarchical relationships and dependency paths among sub-questions.

3.2 Active Internal Knowledge Completion

Phase 1: Pre-order Traversal and Question Completion During the problem-solving phase, we perform a pre-order traversal over the tree structure $\mathcal{T}$, excluding the root node q_0. For each sub-question node q_i encountered during traversal, we first check whether the question is complete. If it contains a placeholder, we identify the referenced sub-question based on the index X (e.g., #queryX), retrieve the answer a_x to that sub-question q_x, and replace the placeholder accordingly:

$$\exists\, \#q_x \in q_i\ ,\ q_x \in \mathcal{T}_{\text{pre}} \Rightarrow\ q_i := q_i\big|_{\#q_x \leftarrow a_x} \tag{2}$$

where $\mathcal{T}_{\text{pre}}$ denotes the pre-order traversal sequence of the tree structure before q_i.

Phase 2: Active Internal Knowledge Completion We apply the Confident QA template (see Table 17) to the question q_i and feed the concatenated input into the model to obtain the response $\hat{r}_i$. If the model is sufficiently confident in its ability to answer the current question q_i based on its internal

knowledge, it directly outputs a high-confidence answer. Otherwise, if the model determines that it lacks the necessary information to provide an accurate response, it outputs a special identifier RAG_REQUIRED. This triggers the RAG process $f(q_i, \mathcal{D}_i)$, wherein the retrieval model $\mathcal{R}$ searches an external knowledge base $\mathcal{D}$ for a set of documents $\mathcal{D}_i$ most relevant to the question. These retrieved documents, along with the question, are then passed to the generation model, which produces the final answer a_i by integrating the external knowledge:

$$a_i = \begin{cases} \hat{r}_i, & \hat{r}_i \neq \mathcal{S} \\ f(q_i, \mathcal{D}_i), & \hat{r}_i = \mathcal{S} \end{cases} \quad \mathcal{D}_i = \mathcal{R}(q_i) \tag{3}$$

where $\mathcal{S}$ represents the retrieval identifier; $f(q_i, D_i)$ represents the generation function based on the question q_i and the retrieved document set D_i.

This mechanism enables active knowledge augmentation: when the model's internal knowledge is insufficient to answer a given question, the system automatically supplements it with external resources, thereby improving both the question's solvability and the answer's accuracy. Notably, suppose the model believes that the provided text does not contain the answer. In that case, we instead apply the Direct QA template to prompt the model to directly generate an answer, thereby avoiding interruptions in the reasoning chain. Additional details are provided in Appendix H.

Phase 1 and Phase 2 are executed alternately until all sub-questions in the question tree have been successfully parsed and answered, at which point the construction of the entire knowledge tree $\mathcal{T}^\star$ is complete:

$$\begin{aligned} \mathcal{T}^\star = (\mathcal{V}, \mathcal{E}) \quad \mathcal{E} \subseteq \mathcal{V} \times \mathcal{V} \\ \mathcal{V} = \{q_0, (q_1, a_1), \ldots, (q_n, a_n)\} \end{aligned} \tag{4}$$

3.3 Knowledge Tree Summarization

In the previous stage, we constructed a knowledge tree $\mathcal{T}^\star$ that captures the structural composition of complex question $\mathcal{Q}$. Building upon this, we further process the tree by extracting summary information along each path π from the root to a leaf node to generate corresponding Knowledge Chains $\mathcal{C}$. This transformation is grounded in the observation that the sub-questions and their answers along each path are semantically coherent, exhibiting informational continuity and logical dependency. As illustrated in the Figure 2, Knowledge Chain One consists of sub-questions Query 1 and Query 2 along with their respective answers. This chain not only preserves the decomposition pathway of the complex question embedded in the tree structure, but also distills the key information required to resolve that particular path:

$$\mathcal{C} = \left\{ \bigcup_{(q_k, a_k) \in \pi} (q_k, a_k) \,\middle|\, \pi \in \mathcal{P}(\mathcal{T}^\star) \right\} \tag{5}$$

where $\mathcal{P}(\mathcal{T}^\star)$ denotes the set of all root-to-leaf paths extracted from the knowledge tree $\mathcal{T}^\star$; $\pi \in \mathcal{P}(\mathcal{T}^\star)$ denotes a specific path (i.e., a branch) within the tree; (q_k, a_k) denotes a sub-question and its corresponding answer on the path.

Finally, the generation model $\mathcal{M}$ produces the final answer $\mathcal{A}$ based on the knowledge chains $\mathcal{C}$ and the complex question $\mathcal{Q}$:

$$\mathcal{A} = \mathcal{M}(\mathcal{C}, \mathcal{Q}) \tag{6}$$

4 Framework-Model Alignment

Instruction-Following (IF) via Knowledge Distillation. To improve the instruction alignment ability of models such as Llama-3.1-8B-Instruct within our framework, we transfer the strengths of the GPT series in instruction comprehension and execution. Specifically, we randomly sampled 2,000 examples from the training sets of HotpotQA and 2WikiMultihopQA, and solved these problems using GPT-4o. We retain intermediate outputs for the successfully solved cases such as question decomposition, answers, and summaries generated during the problem-solving process, thereby constructing an instruction-following dataset. This dataset is then used to enhance the target model's instruction-following capabilities.

Knowledge Boundary Awareness Enhancement (KBAE) via Model Bootstrapping. To enhance models' ability to recognize the boundaries of their own knowledge, we propose a mechanism based on multiple single-hop QA datasets. When a model produces a correct answer in the Direct QA mode, we expect it to output the same answer when switched to the Confident QA mode. Conversely, if the model produces an incorrect answer in the Direct QA mode, it is required to return a special retrieval indicator in the Confident QA mode, signaling that it has recognized its own knowledge limitation. This approach results in a knowledge boundary awareness enhancement dataset, which

Statistic	Value
# The data scale	12685
# The data scale of **IF**	9589
# The average length of input instruction	267.8
# The average length of output	27.4
# The data scale of **KBAE**	3096
# The average length of input instruction	148.3
# The average length of output	3.9

Table 1: Statistics of the synthetic dataset.

improves the model's self-awareness of its knowledge coverage. It is worth noting that the determination of knowledge boundaries for the same question may differ due to variations in internal knowledge among different models. As such, the knowledge boundary annotations are model-specific.

Objective of Training. The statistics of the synthetic dataset $\mathcal{D}_s$ are summarized in Table 1. More details about training data can be found in Appendix F. Our training objective is to fit the large language model $\mathcal{M}_\theta$ to the distribution of the synthetic dataset $\mathcal{D}_s$. During model training, we adopt the commonly used next token prediction task in language modeling and use cross-entropy loss as the objective function, as defined below:

$$\theta^* = \arg\min_\theta \mathbb{E}_{(x,y)\sim\mathcal{D}_s} \left[\mathcal{L}_{CE}(\mathcal{M}_\theta(x), y)\right] \quad (7)$$

5 Experimental Settings

Datasets and Evaluation Metrics We evaluate our approach on four widely-used MHQA datasets: HotpotQA (Yang et al., 2018), 2WikiMultihopQA (Ho et al., 2020), MuSiQue (Trivedi et al., 2022), and Bamboogle (Press et al., 2023). For Bamboogle, we use all 125 examples from its test set. For the other datasets, we randomly sample 500 examples from their respective development sets. Evaluation is conducted using three metrics: F1, Exact Match (EM), and Semantic Accuracy (Acc†). The F1 score measures the token-level overlap between the model's prediction and the ground truth. Exact Match requires the predicted answer to match the ground truth exactly. Semantic Accuracy leverages an LLM to assess whether the predicted answer is semantically correct with respect to the ground truth. Further details about semantic accuracy evaluation can be found in Appendix G.

Baselines We compare our approach against both *Generation w/o Retrieval* and *Generation w/ Retrieval* methods. Here, *w/* stands for *with* and *w/o* for *without*. *Generation w/o Retrieval* methods include: (1) Direct QA, which directly prompts the model to generate the final answer; (2) CoT (Wei et al., 2022), which first generates intermediate reasoning steps before producing the final answer.

Generation w/ Retrieval methods include: (1) One-time Retrieval (OneR), where the model answers the question based on documents retrieved in a single step; (2) RetGen (Shao et al., 2023), which integrates iterative retrieval conditioned on previously generated text and queries; (3) Self-Ask (Press et al., 2023), which decomposes the original question into sub-questions and builds the final answer step by step; (4) FLARE (Jiang et al., 2023), which dynamically adjusts the retrieval timing and content based on token probabilities of intermediate reasoning; (5) DRAGIN (Su et al., 2024), which uses internal model signals to determine when and what to retrieve; (6) GenGround (Shi et al., 2024), which alternates between answer generation and answer revision stages; (7) CompAct (Yoon et al., 2024), which dynamically retains key information and integrates content across multiple documents; (8) DyPlan (Parekh et al., 2025), which dynamically selects strategies based on the input question and perform internal verification after generating the answer; (9) Search-R1 (Jin et al., 2025), which trains LLMs via reinforcement learning to perform autonomous retrieval during reasoning.

Implementation Details We employ GPT-4o (OpenAI et al., 2024a) and Llama-3.1-8B-Instruct (Grattafiori et al., 2024) (Llama-3.1 for short) as the backbone models for our framework and all baselines. For the knowledge source, we use the Wikipedia dump provided by KILT (Petroni et al., 2021), dated August 1, 2019. BM25 (Robertson and Walker, 1994) is adopted as the retrieval model, while e5-base-v2 (Wang et al., 2024) is used for reranking. When retrieval is required for a given question, we return the top 5 passages with the highest reranking scores. For model training, we set the learning rate to 5e-5 and adopt the LoRA (Hu et al., 2022) method for efficient parameter fine-tuning. The models are trained for three epochs using the AdamW (Loshchilov and Hutter, 2019) optimizer.

6 Experimental Results

6.1 Overall Performance

The experimental results are presented in Table 2. We observe that for GPT-4o, a powerful proprietary model, the gain from single-turn retrieval is rela-

Methods	HotpotQA			2WikiMultihopQA			MuSiQue			Bamboogle			Average
	F1	EM	Acc†	F1	EM	Acc†	F1	EM	Acc†	F1	EM	Acc†	
generation w/o retrieval based on GPT-4o													
Direct QA	48.0	36.2	52.2	42.5	34.2	41.8	20.9	8.0	20.8	40.4	30.4	39.2	34.5
CoT	55.1	41.2	63.8	55.6	45.8	60.6	30.0	16.4	31.6	72.6	54.4	74.4	50.1
generation w/ retrieval based on GPT-4o													
OneR	52.4	39.6	58.4	42.9	35.2	44.0	17.0	7.0	17.6	36.8	25.6	40.8	34.8
CompAct	54.9	40.8	62.2	44.2	36.2	47.6	19.2	10.6	21.6	39.7	28.8	44.0	37.5
RetGen	52.2	37.8	67.6	44.2	33.0	55.4	25.7	13.2	29.6	41.2	32.8	55.2	40.6
Self-Ask	50.6	38.0	61.8	52.5	44.4	57.4	25.6	13.4	28.8	57.8	43.2	59.2	44.4
FLARE	53.1	40.6	59.2	51.7	42.4	55.0	24.6	12.8	25.2	68.3	54.4	71.2	46.5
GenGround	**61.3**	**46.8**	**68.8**	61.6	47.4	65.4	32.4	17.4	33.8	70.9	58.4	70.0	52.9
TAIKC	58.1	44.2	64.2	**64.2**	**53.8**	**67.2**	**33.9**	**20.2**	**35.8**	**73.9**	**65.6**	**76.0**	**54.8**
generation w/o retrieval based on Llama-3.1-8B-Instruct													
Direct QA	30.5	23.0	34.2	29.2	24.4	28.2	9.0	2.6	8.0	16.3	11.2	16.0	19.4
CoT	33.0	22.6	40.2	23.5	18.0	21.8	11.4	3.8	11.2	49.4	38.4	48.8	26.8
generation w/ retrieval based on Llama-3.1-8B-Instruct													
OneR	43.7	33.4	50.0	26.5	20.6	27.0	9.8	3.8	8.6	23.5	15.2	24.0	23.8
CompAct	45.6	34.2	51.0	31.9	24.2	32.4	11.6	4.8	10.2	22.8	17.6	22.4	25.7
RetGen	40.0	28.0	50.2	32.2	24.2	36.6	13.0	6.4	14.4	20.6	12.0	34.4	26.0
Self-Ask	40.5	30.0	48.0	35.9	30.6	40.2	13.9	7.2	16.4	38.2	27.2	42.4	30.9
FLARE	37.3	27.0	42.4	32.5	27.0	33.4	13.6	5.2	12.8	50.1	31.2	50.0	30.2
GenGround	41.3	30.2	48.6	34.5	29.0	38.2	12.5	5.2	11.4	26.3	17.6	24.8	26.6
DRAGIN	48.6	37.2	53.4	43.9	36.6	45.8	18.5	9.0	16.6	50.4	40.8	49.6	37.5
DyPlan	49.7	38.0	55.0	49.8	42.4	52.4	18.7	8.0	15.6	46.7	36.8	46.4	38.3
Search-R1	**52.9**	**41.0**	**58.8**	45.6	39.2	50.4	19.5	11.2	18.4	47.4	33.6	46.4	38.7
TAIKC	49.7	38.2	54.8	**57.7**	**49.8**	**59.8**	**22.1**	**11.4**	**21.4**	**50.5**	**42.4**	**50.4**	**42.3**

Table 2: Experimental results on four open-domain multi-hop question answering datasets. The best and second-best results are highlighted in **bold** and underlined.

tively small (34.5 → 34.8). In contrast, the open-source Llama-3.1 model benefits more significantly from retrieval (19.4 → 23.8), which we attribute to the larger knowledge gap between the two models. Compared to single-turn retrieval, iterative dynamic retrieval leads to substantial improvements, as the model can acquire more external knowledge in an iterative manner based on its specific needs, thereby enabling better problem-solving.

As shown in Table 2, our method achieves the best results on three datasets: 2WikiMultihopQA, MuSiQue, and Bamboogle. On average, our approach outperforms previous state-of-the-art (SOTA) methods GenGround and Search-R1 by significant margins (+1.9 and +3.6). We attribute these improvements to the following factors: (i) hierarchical knowledge integration enhances the relevance of information aggregation, effectively mitigating the interference from redundant context through structured knowledge tree construction; (ii) the knowledge-boundary-based active retrieval mechanism enables efficient coordination between internal and external knowledge sources, reducing ineffective retrievals and improving information utilization; (iii) the enhanced instruction-following and boundary-awareness capabilities improve the adaptability between the model and the framework.

Further analysis in Figure 3 shows that our framework achieves notable gains on complex reasoning types such as bridge comparison and 4-hop questions. These tasks typically involve indirect associations among multiple entities or information chains spanning multiple paragraphs, demanding stronger capabilities in knowledge organization and deeper reasoning. We attribute the performance gains to our proposed hierarchical question modeling and knowledge chain construction mechanisms, which enable more effective organization and utilization of knowledge in complex reasoning scenarios. In contrast, existing approaches often lack structured organization when handling intermediate information, which can lead to the accumulation of

Methods	HotpotQA		2WikiMultihopQA		MuSiQue		Bamboogle		Average
	F1	Acc†	F1	Acc†	F1	Acc†	F1	Acc†	
TAIKC (GPT-4o)	58.1	64.2	64.2	67.2	33.9	35.8	73.9	76.0	59.2
Internal Only	51.7 ($\downarrow_6$)	57.4 ($\downarrow_7$)	57.2 ($\downarrow_7$)	60.0 ($\downarrow_7$)	29.8 ($\downarrow_4$)	32.2 ($\downarrow_4$)	78.7 ($\uparrow_5$)	80.0 ($\uparrow_4$)	55.9 ($\downarrow_{5\%}$)
External Only	54.6 ($\downarrow_4$)	61.6 ($\downarrow_3$)	61.2 ($\downarrow_3$)	64.4 ($\downarrow_3$)	30.0 ($\downarrow_4$)	32.0 ($\downarrow_4$)	60.2 ($\downarrow_{14}$)	62.4 ($\downarrow_{14}$)	53.3 ($\downarrow_{10\%}$)
Based on Prob	49.5 ($\downarrow_9$)	54.2 ($\downarrow_{10}$)	55.7 ($\downarrow_9$)	56.0 ($\downarrow_{11}$)	29.5 ($\downarrow_4$)	29.6 ($\downarrow_6$)	71.9 ($\downarrow_2$)	73.6 ($\downarrow_2$)	52.5 ($\downarrow_{11\%}$)
TAIKC (Llama-3.1)	49.6	54.2	57.4	59.6	22.7	21.8	49.7	48.8	45.5
w/o IF	45.6 ($\downarrow_4$)	49.8 ($\downarrow_4$)	47.6 ($\downarrow_{10}$)	49.8 ($\downarrow_{10}$)	17.5 ($\downarrow_5$)	16.0 ($\downarrow_6$)	41.5 ($\downarrow_8$)	40.0 ($\downarrow_9$)	38.5 ($\downarrow_{15\%}$)
w/o KBAE	37.6 ($\downarrow_{12}$)	41.2 ($\downarrow_{13}$)	43.6 ($\downarrow_{14}$)	43.2 ($\downarrow_{16}$)	15.1 ($\downarrow_8$)	13.2 ($\downarrow_9$)	43.8 ($\downarrow_6$)	42.4 ($\downarrow_6$)	35.0 ($\downarrow_{22\%}$)
w/o IF & KBAE	34.7 ($\downarrow_{15}$)	38.2 ($\downarrow_{16}$)	34.8 ($\downarrow_{23}$)	35.4 ($\downarrow_{24}$)	12.7 ($\downarrow_{10}$)	10.8 ($\downarrow_{11}$)	40.5 ($\downarrow_9$)	36.8 ($\downarrow_{12}$)	30.5 ($\downarrow_{30\%}$)

Table 3: Results of ablation study. The upper part of the table presents ablation settings for different knowledge collaboration strategies, while the lower part reports ablation settings for various training modules.

redundancy and the overshadowing of critical information, thereby hindering the stable construction of complete reasoning paths.

6.2 Ablation Study

Effect of Knowledge Collaboration Strategy We designed three sets of experiments: one using only internal knowledge, one using only external knowledge, and one leveraging model output probabilities to determine when to perform retrieval. The results show that, except for Bamboogle, models experience a more substantial performance drop when relying solely on internal knowledge (Internal ↓35 vs. External ↓21), suggesting that internal knowledge alone is insufficient for task completion and highlighting the critical role of external knowledge. Interestingly, internal knowledge yields the best performance on Bamboogle, likely because the dataset was constructed in 2023, whereas the external knowledge base is outdated (last updated in 2019) and offers limited support. Overall, using a single source of knowledge significantly degrades performance (Internal ↓5% & External ↓10%), while combining internal and external knowledge (Unified Knowledge) achieves the best results, validating the effectiveness of the integration strategy. Moreover, using output probability to decide whether to retrieve knowledge leads to a performance drop (↓11%), further demonstrating the superiority of our Confident QA approach.

Effect of Model Modules To investigate the contribution of key modules, we performed ablations on the IF and KBAE modules. The results indicate that removing the KBAE module causes a greater performance drop (↓22%) compared to removing the IF module (↓15%), suggesting that KBAE plays a critical role in boundary recognition and coordination between internal and external knowledge. Removing both modules results in the most significant performance degradation (↓30%), indicating a synergistic effect: the IF module enhances the understanding and response to user instructions, while KBAE improves knowledge boundary awareness. Together, they support the model's reasoning and knowledge retrieval capabilities.

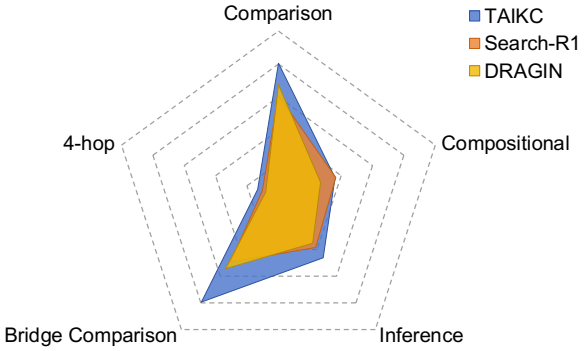

Figure 3: Performance comparison of various methods across the four question types in 2WikiMultihopQA and the 4-hop question type in MuSiQue.

7 Analyses and Discussions

7.1 Knowledge Collaboration

We conducted a systematic analysis of the models' performance in coordinating the use of internal and external knowledge, with the results presented in Figure 4. The analysis reveals the following findings: (1) *There are significant differences in collaboration patterns across datasets.* Specifically, on the Bamboogle dataset, the model tends to exhibit high confidence in its self-generated answers and primarily relies on internal knowledge to complete the task. In contrast, on the HotpotQA dataset, the model more frequently leverages externally retrieved information to support reasoning. This suggests that the task characteristics of a dataset influence the extent to which a model depends on external knowledge. (2) *Models also differ in their knowledge coordination strategies.*

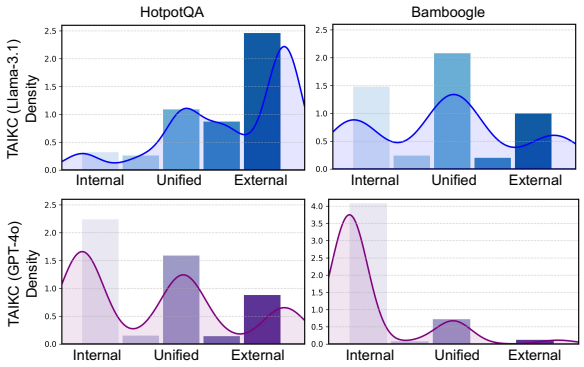

Figure 4: Distribution of knowledge collaboration in reasoning. *Unified* represents collaboration between internal and external knowledge sources, while the two ends indicate reliance on a single source of knowledge. The bars represent the original discrete distribution, and the curve denotes the kernel density estimation (KDE).

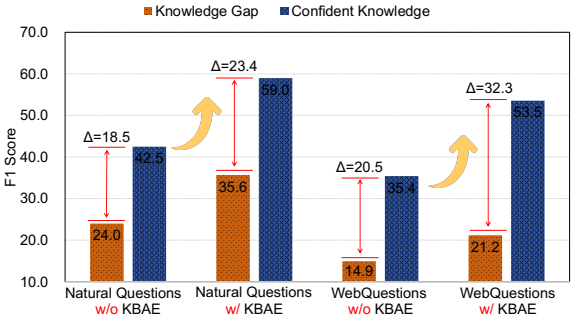

Figure 5: Comparison of knowledge boundary awareness performance on two datasets *w/o* and *w/* KBAE.

For instance, comparing GPT-4o and Llama-3.1, the latter tends to rely more heavily on external information sources in most scenarios. This result aligns with our expectations: compared to large proprietary models like GPT-4o, Llama-3.1 has more limited internal knowledge coverage and thus depends more on external knowledge to compensate for internal deficiencies.

7.2 Effectiveness of the KBAE Module

To validate the effectiveness of the KBAE module in enhancing the model's awareness of knowledge boundaries, we conducted a comparative analysis of the model's performance before and after incorporating the KBAE module. The detailed results are shown in Figure 5. First, the model's answer accuracy in the Confident Knowledge domain significantly improved (42.5→59.0, 35.4→53.5), indicating that the KBAE module effectively enhances the model's ability to utilize known knowledge. Second, the accuracy gap between the Confident Knowledge and Knowledge Gap domains also increased notably (18.5→23.4, 20.5→32.3), suggesting a strengthened capacity to distinguish between knowledge within and beyond the boundary. Combined with the conclusion from the pilot study in Appendix C (accuracy should be high, and accuracy gap between Confident Knowledge and Knowledge Gap should be large), we infer that the KBAE module helps improve the model's awareness of its knowledge boundaries, allowing it to better judge whether it possesses sufficient knowledge to answer a question, and thus make more appropriate decisions regarding external information retrieval.

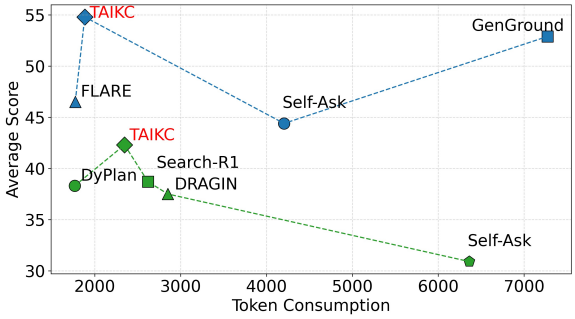

Figure 6: Results are averaged over four datasets. The upper-left quadrant indicates higher efficiency and better performance. The blue icons indicate that the backbone model is GPT-4o, while the red represent Llama-3.1.

7.3 Analysis of Reasoning Cost

We compare our framework with previous SOTA methods in terms of both performance and efficiency, as shown in Figure 6. Whether based on the GPT-4o model or the Llama-3.1 model, our framework demonstrates superior overall performance and higher efficiency. We attribute this to the fact that our approach avoids the accumulation and iterative processing of intermediate outputs and employs active retrieval decision-making, which significantly reduces the inference costs.

8 Conclusions

This paper introduces TAIKC for knowledge-intensive multi-hop question answering. TAIKC decomposes complex questions into a tree structure of interrelated sub-questions addressed via pre-order traversal. It leverages LLMs' knowledge boundary perception capabilities to actively select strategies for solving each sub-question. The completed tree structure is then used to generate coherent knowledge chains to solve the complex question. Overall, TAIKC facilitates multi-hop reasoning through efficient information aggregation and active retrieval decision-making. Extensive experiments on four multi-hop datasets demonstrate its effectiveness.

Limitations

The effectiveness of our method can be attributed to the groundbreaking advances made by the research community in mitigating hallucinations in LLMs, which have enabled current LLMs to develop a clear awareness of their own knowledge boundaries. Specifically, our approach functions optimally only when the model can accurately determine whether it knows the answer to a given query; conversely, its performance may decline when the model exhibits uncertainty about its own knowledge state (i.e., suffers from significant hallucination). We conducted preliminary experiments and analyses on the adaptability of our framework to several popular models, as presented in Appendix D. In brief, we do not recommend applying our framework to models with 7B parameters or fewer, as such models typically exhibit more severe hallucinations and struggle to determine when to retrieve based on their internal knowledge state. Although we have proposed aligning the framework with smaller models (e.g., Llama-3.1-8B-Instruct) through techniques such as knowledge distillation and model bootstrapping, we argue that these efforts are insufficient to overcome the fundamental limitations faced by models with fewer than 7B parameters. We further posit that, as large language models continue to evolve, the synergistic effect between our method and these models will demonstrate even greater potential.

Ethics Statement

In this study, we strictly adhered to ethical guidelines to ensure the fairness and reliability of our research. All experimental designs and measurement tools were based on publicly available standards and validated resources, ensuring high transparency and reproducibility. Furthermore, all foundation models, retrieval models, and datasets used in this work are publicly accessible, primarily sourced from open-access academic repositories and public data platforms. This approach minimizes data bias and promotes research fairness. We have carefully considered the potential impact of our research on individuals and communities, avoided any harm to persons or organizations, and ensured that neither the research process nor its outcomes involve misleading information or data misuse.

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*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1533–1544, Seattle, Washington, USA. Association for Computational Linguistics.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, and 1 others. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Shulin Cao, Jiajie Zhang, Jiaxin Shi, Xin Lv, Zijun Yao, Qi Tian, Lei Hou, and Juanzi Li. 2023. Probabilistic tree-of-thought reasoning for answering knowledge-intensive complex questions. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 12541–12560, Singapore. Association for Computational Linguistics.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 2318–2335, Bangkok, Thailand. Association for Computational Linguistics.

Zheng Chu, Jingchang Chen, Qianglong Chen, Haotian Wang, Kun Zhu, Xiyuan Du, Weijiang Yu, Ming Liu, and Bing Qin. 2024. BeamAggR: Beam aggregation reasoning over multi-source knowledge for multi-hop question answering. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1229–1248, Bangkok, Thailand. Association for Computational Linguistics.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2025. From local to global: A graph rag approach to query-focused summarization. *Preprint*, arXiv:2404.16130.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

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, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Zhengbao Jiang, Frank 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, Singapore. Association for Computational Linguistics.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. 2025. Search-r1: Training llms to reason and leverage search engines with reinforcement learning. *Preprint*, arXiv:2503.09516.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2023. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive nlp. *Preprint*, arXiv:2212.14024.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering

research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

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 *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Dongyang Li, Junbing Yan, Taolin Zhang, Chengyu Wang, Xiaofeng He, Longtao Huang, Hui Xue', and Jun Huang. 2024. On the role of long-tail knowledge in retrieval augmented large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 120–126, Bangkok, Thailand. Association for Computational Linguistics.

Yuankai Li, Jia-Chen Gu, Di Wu, Kai-Wei Chang, and Nanyun Peng. 2025. BRIEF: Bridging retrieval and inference for multi-hop reasoning via compression. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 5449–5470, Albuquerque, New Mexico. Association for Computational Linguistics.

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. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4730–4749, Bangkok, Thailand. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Yuanjie Lyu, Zihan Niu, Zheyong Xie, Chao Zhang, Tong Xu, Yang Wang, and Enhong Chen. 2024. Retrieve-plan-generation: An iterative planning and answering framework for knowledge-intensive LLM generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 4683–4702, Miami, Florida, USA. Association for Computational Linguistics.

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 (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Shengyu Mao, Yong Jiang, Boli Chen, Xiao Li, Peng Wang, Xinyu Wang, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2024. RaFe: Ranking feedback improves query rewriting for RAG. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 884–901, Miami, Florida, USA. Association for Computational Linguistics.

OpenAI, :, Aaron Hurst, Adam Lerer, Adam P. Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, Aleksander Mądry, Alex Baker-Whitcomb, Alex Beutel, Alex Borzunov, Alex Carney, Alex Chow, Alex Kirillov, and 401 others. 2024a. Gpt-4o system card. *Preprint*, arXiv:2410.21276.

OpenAI, :, Aaron Jaech, Adam Kalai, Adam Lerer, Adam Richardson, Ahmed El-Kishky, Aiden Low, Alec Helyar, Aleksander Madry, Alex Beutel, Alex Carney, Alex Iftimie, Alex Karpenko, Alex Tachard Passos, Alexander Neitz, Alexander Prokofiev, Alexander Wei, Allison Tam, and 244 others. 2024b. Openai o1 system card. *Preprint*, arXiv:2412.16720.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, and 1 others. 2022. Training language models to follow instructions with human feedback. *Advances in neural information processing systems*, 35:27730–27744.

Tanmay Parekh, Pradyot Prakash, Alexander Radovic, Akshay Shekher, and Denis Savenkov. 2025. Dynamic strategy planning for efficient question answering with large language models. In *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6038–6059, Albuquerque, New Mexico. Association for Computational Linguistics.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. 2021. KILT: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544, Online. Association for Computational Linguistics.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711, Singapore. Association for Computational Linguistics.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, and 25 others. 2025. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

S. E. Robertson and S. Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '94, page 232–241, Berlin, Heidelberg. Springer-Verlag.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274, Singapore. Association for Computational Linguistics.

Zhengliang Shi, Shuo Zhang, Weiwei Sun, Shen Gao, Pengjie Ren, Zhumin Chen, and Zhaochun Ren. 2024. 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.

Huatong Song, Jinhao Jiang, Yingqian Min, Jie Chen, Zhipeng Chen, Wayne Xin Zhao, Lei Fang, and Ji-Rong Wen. 2025. R1-searcher: Incentivizing the search capability in llms via reinforcement learning. *Preprint*, arXiv:2503.05592.

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. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12991–13013, Bangkok, Thailand. Association for Computational Linguistics.

Jiejun Tan, Zhicheng Dou, Yutao Zhu, Peidong Guo, Kun Fang, and Ji-Rong Wen. 2024. Small models, big insights: Leveraging slim proxy models to decide when and what to retrieve for LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4420–4436, Bangkok, Thailand. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. MuSiQue: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada. Association for Computational Linguistics.

Venktesh V, Mandeep Rathee, and Avishek Anand. 2025. SUNAR: Semantic uncertainty based neighborhood aware retrieval for complex QA. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5818–5835, Albuquerque, New Mexico. Association for Computational Linguistics.

Liang Wang, Haonan Chen, Nan Yang, Xiaolong Huang, Zhicheng Dou, and Furu Wei. 2025. Chain-of-retrieval augmented generation. *Preprint*, arXiv:2501.14342.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2024. Text embeddings by weakly-supervised contrastive pre-training. *Preprint*, arXiv:2212.03533.

Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query expansion with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, Singapore. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Zhepei Wei, Wei-Lin Chen, and Yu Meng. 2025. InstructRAG: Instructing retrieval-augmented generation via self-synthesized rationales. In *The Thirteenth International Conference on Learning Representations*.

Tomer Wolfson, Mor Geva, Ankit Gupta, Matt Gardner, Yoav Goldberg, Daniel Deutch, and Jonathan Berant. 2020. Break it down: A question understanding benchmark. *Transactions of the Association for Computational Linguistics*, 8:183–198.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2024a. RECOMP: Improving retrieval-augmented LMs with context compression and selective augmentation. In *The Twelfth International Conference on Learning Representations*.

Peng Xu, Wei Ping, Xianchao Wu, Chejian Xu, Zihan Liu, Mohammad Shoeybi, and Bryan Catanzaro. 2025. ChatQA 2: Bridging the gap to proprietary LLMs in long context and RAG capabilities. In *The Thirteenth International Conference on Learning Representations*.

Shicheng Xu, Liang Pang, Huawei Shen, Xueqi Cheng, and Tat-Seng Chua. 2024b. Search-in-the-chain: Interactively enhancing large language models with search for knowledge-intensive tasks. In *The Web Conference 2024*.

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*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2023a. Tree of thoughts: Deliberate problem solving with large language models. In *Advances in Neural Information Processing Systems*, volume 36, pages 11809–11822. Curran Associates, Inc.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023b. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Chanwoong Yoon, Taewhoo Lee, Hyeon Hwang, Minbyul Jeong, and Jaewoo Kang. 2024. CompAct: Compressing retrieved documents actively for question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21424–21439, Miami, Florida, USA. Association for Computational Linguistics.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2024. Making retrieval-augmented language models robust to irrelevant context. In *The Twelfth International Conference on Learning Representations*.

Tian Yu, Shaolei Zhang, and Yang Feng. 2024. Autorag: Autonomous retrieval-augmented generation for large language models. *Preprint*, arXiv:2411.19443.

Zhenrui Yue, Honglei Zhuang, Aijun Bai, Kai Hui, Rolf Jagerman, Hansi Zeng, Zhen Qin, Dong Wang, Xuanhui Wang, and Michael Bendersky. 2025. Inference scaling for long-context retrieval augmented generation. In *The Thirteenth International Conference on Learning Representations*.

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, Mexico City, Mexico. Association for Computational Linguistics.

Nan Zhang, Prafulla Kumar Choubey, Alexander Fabbri, Gabriel Bernadett-Shapiro, Rui Zhang, Prasenjit Mitra, Caiming Xiong, and Chien-Sheng Wu. 2025. SireRAG: Indexing similar and related information for multihop reasoning. In *The Thirteenth International Conference on Learning Representations*.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyan Luo. 2024. LlamaFactory: Unified efficient fine-tuning of 100+ language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

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, Miami, Florida, USA. Association for Computational Linguistics.

A Overview

For readers seeking to explore additional questions or gain further details, we provide a comprehensive appendix with dedicated sections addressing specific topics. The correspondence between each appendix and its contents is as follows:

(1) In Appendix B, we list the models and datasets used in our study along with their respective licenses. According to the details of these licenses, all resources are permitted for academic research use.

- What are the license terms for the models, datasets, and other resources used in this work?

(2) We also present the single-hop and multi-hop datasets employed in our experiments in Appendix E.1.

- Which datasets are used in this study?
- What are the characteristics of the data contained in these datasets?

(3) Appendix C provides an analysis of the knowledge boundary awareness of the GPT-4o model.

- What is knowledge boundary awareness?
- Does the model truly possess this capability?
- How can we investigate whether a model is able to perceive its knowledge boundaries?

(4) To investigate whether our framework can be adapted to other popular models beyond GPT-4o, we conducted a preliminary test and evaluation of their knowledge boundary awareness capabilities, as detailed in Appendix D.

- Do models of different series and scales also exhibit knowledge boundary awareness?

(5) Appendix F provides detailed information about training.

- How is the augmented training data constructed for the model?
- What hyperparameters are used for model training?

(6) Appendix G presents the details of the semantic evaluation, as well as the rationale for selecting the Llama-3.1-8B-Instruct model over the GPT-4o model for evaluation, due to its highly consistent decision outcomes and lower resource consumption.

- What prompts are used to assess semantic accuracy?
- Is it feasible to use Llama-3.1 as a substitute for the GPT-series models commonly used in prior work for semantic evaluation?

(7) Appendix H presents the details of the reasoning continuation mechanism.

- How is inference continuation implemented?
- Approximately how much of the data triggers this continuation mechanism?
- What is the impact on framework performance when this mechanism is disabled?

(8) Furthermore, despite the strong overall performance of our framework, we identify several scenarios where its effectiveness decreases. A detailed error analysis of these cases is provided in Appendix I.

- What types of errors may arise within the framework, and how do they affect its performance?

(9) To facilitate a deeper understanding of our proposed framework, we provide additional materials in Appendix J.

B Licenses

The large language models, including the Qwen-2.5 (Qwen et al., 2025) series, are released under the Apache License 2.0, while the Llama-3.1 (Grattafiori et al., 2024) series is distributed under the LLAMA 3.1 COMMUNITY LICENSE. The retrieval model e5-base-v2 (Wang et al., 2024) is licensed under the MIT License. Detailed information can be found on their respective GitHub pages. These licenses permit users to freely use, modify, and distribute the data. The GPT series models used in our work are developed and released by OpenAI.

For single-hop datasets, Natural Questions (Kwiatkowski et al., 2019) and TriviaQA (Joshi et al., 2017) are released under the Apache License 2.0, WebQuestions (Berant et al., 2013) under the CC-BY 4.0 License, and PopQA (Mallen et al., 2023) under the MIT License. Regarding multi-hop datasets, HotpotQA (Yang et al., 2018) is licensed under CC BY-SA 4.0, 2WikiMultihopQA (Ho et al., 2020) under Apache License 2.0, MuSiQue (Trivedi et al., 2022) under CC-BY 4.0, and Bamboogle (Press et al., 2023) under MIT License. In summary, all of these licenses permit academic use.

C Pilot Study

For a given question, the model is clearly aware of whether it knows the correct answer. We refer to this ability as knowledge boundary-awareness.

Methods	Natural Questions				TriviaQA				PopQA				WebQuestions				Avg
	Num	F1	EM	Acc†	Num	F1	EM	Acc†	Num	F1	EM	Acc†	Num	F1	EM	Acc†	
DirQA	100	53.0	34.0	68.0	100	91.8	90.0	90.0	100	61.1	54.0	58.0	100	40.8	18.0	60.0	**59.9**
ConfQA	67	68.7	46.3	82.1	93	94.6	92.5	94.6	57	79.5	73.7	75.4	78	51.3	25.6	65.4	**70.8**
DirQA§	33	22.0	21.2	33.3	7	35.7	28.6	42.9	43	35.8	30.2	32.6	22	9.8	0.0	36.4	**27.4**

Table 4: Performance comparison across different datasets. The § symbol denotes that the model responds in the DirQA setting but returns RAG_REQUIRED in the ConfQA scenario (Knowledge Gap). We present the accuracy results of Confident Knowledge (outputs in the ConfQA scenario) and Knowledge Gap in Figure 9 for visualization.

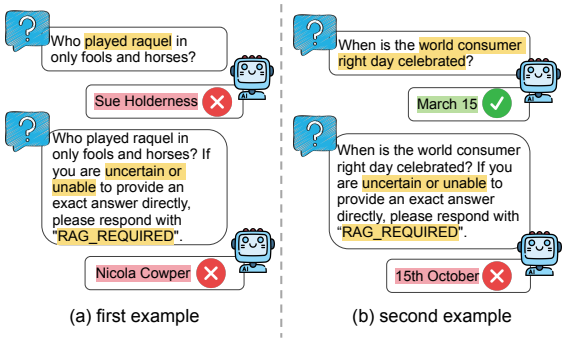

Figure 7: Two examples illustrating inconsistent model outputs under the DirQA and ConfQA settings.

C.1 Hypotheses

To investigate whether large language models possess an awareness of their own knowledge boundaries, we designed and conducted an experiment. Specifically, we randomly sampled 100 instances from each of four widely used single-hop question answering datasets, constructing a test set of 400 samples. We evaluated the performance of the GPT-4o model under two different prompting settings:

- **Direct QA:** The model answers each question in the conventional manner, prompted using the *Prompt for Direct QA*.
- **Confident QA:** The model provides an answer only when it is very confident in its response; otherwise, it outputs "RAG_REQUIRED." This is implemented using the *Prompt for Confident QA*. For answers provided by the model under the Confident QA setting, we refer to them as Confident Knowledge, and for questions where the model opts to perform RAG, we refer to them as Knowledge Gap, as illustrated in Figure 1.

The prompts used for Direct QA and Confident QA in our study can be found in Table 17. The objective of the experiment is to validate the following two core hypotheses:

- **Consistency Hypothesis:** When the model exhibits high confidence in its answers, such as in the case of simple arithmetic questions (e.g., "What is the capital of France?"). These questions are assumed to fall well within the model's knowledge boundaries. In such cases, the model's responses should remain consistent across both the Direct QA (DirQA) and Confident QA (ConfQA) settings. Specifically, regardless of whether the model is asked to respond under normal conditions or only when it is "very confident" in its answers, the outputs should be highly consistent. It should not produce entirely different answers in the ConfQA setting compared to those in the DirQA setting.
- **Reliability Hypothesis:** (1) First, under the ConfQA setting, the samples that the model chooses to answer should have a high accuracy rate, indicating that the model provides high-quality outputs only when it is genuinely confident. (2) Furthermore, the model's accuracy in the ConfQA setting should be significantly higher than in the DirQA setting. If the accuracy is roughly the same in both settings, it suggests that the model lacks awareness of its own knowledge boundaries and is unable to discern when it knows or does not know something. (3) Next, for instances labeled as "RAG_REQUIRED" that the model chooses to answer in the ConfQA setting, the corresponding accuracy in the DirQA setting should be relatively low. This would indicate that the model is indeed unfamiliar with such information, supporting the idea that these samples lie outside its intrinsic knowledge. (4) To summarize, points (2) and (3) suggest that there should be a significant accuracy gap between Confident Knowledge and the Knowledge Gap.

C.2 Results and Analyses

As shown in Figure 7 (a), the model produced an incorrect answer under the DirQA setting and then altered its response under the ConfQA setting for

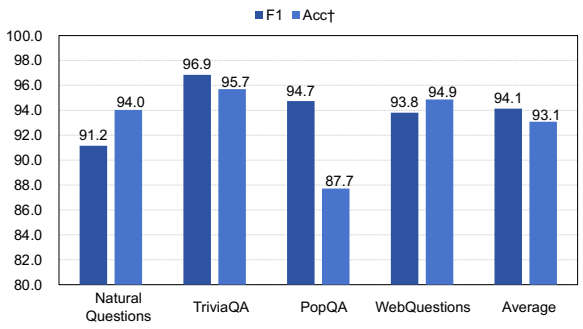

Figure 8: Results of model output consistency comparison based on GPT-4o.

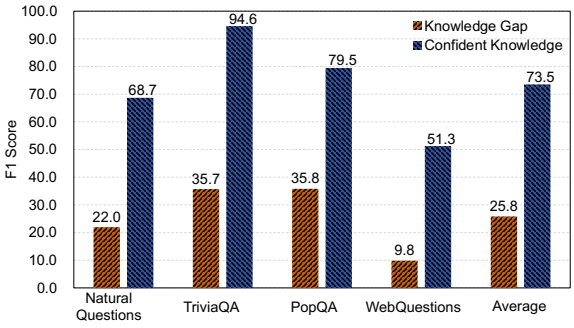

Figure 9: Accuracy Comparison of Confident Knowledge and Knowledge Gap Based on GPT-4o.

the same question, resulting in a different but still incorrect answer. Furthermore, as illustrated in Figure 7 (b), the model provided a correct answer under the DirQA setting but changed to an incorrect answer under the ConfQA setting. These behaviors are undesirable. We do not expect the model to change its answers simply due to variations in the prompt's content or tone, as such inconsistency reflects a lack of confidence in its own knowledge.

Therefore, for **Consistency Hypothesis**, we evaluate the consistency of model outputs under two settings: DirQA and ConfQA. Specifically, we measure the degree of consistency between the answers generated by the model for the same question under different prompting strategies, using F1 score and semantic accuracy as evaluation metrics. As shown in Figure 8, the model achieves an average F1 score of 94.1 and a semantic consistency score of 93.1 across the two settings, indicating a high degree of agreement in model outputs under varying prompts. These results support our proposed consistency hypothesis, which posits that when the model has high confidence in its answer to a question, it tends to produce consistent responses regardless of the prompting strategy.

For **Reliability Hypothesis** (1) and (2), we further compare the answer accuracy between the DirQA and ConfQA settings. Results are shown in Table 4. In the ConfQA approach, the model outperforms the DirQA method across all four datasets. Specifically, the average score of the ConfQA method reaches 70.8, significantly higher than the 59.9 achieved by DirQA, indicating that the ConfQA strategy contributes to improved answer accuracy. Within the ConfQA framework, the model's output is referred to as a Confident Knowledge. Notably, when the model encounters questions for which it is uncertain or lacks sufficient knowledge, it returns a special token, RAG_REQUIRED, rather than generating an answer forcibly. Although the model is not explicitly trained to recognize "I don't know" scenarios, the incentives introduced in the ConfQA setup encourage a form of knowledge boundary awareness.

For **Reliability Hypothesis** (3) and (4). We constructed a knowledge gap subset by extracting from the DirQA setting those questions for which the model output RAG_REQUIRED under the ConfQA setting. Another subset is the Confident Knowledge obtained under the ConfQA setting. Results are shown in Figure 9. On the knowledge gap subset, the model's average accuracy under the DirQA setting drops to just 25.8, significantly lower than the 73.5 accuracy of the Confident Knowledge and there's a significant accuracy gap of 47.7 points. This finding provides further support for our hypothesis regarding the model's epistemic awareness: the model achieves higher accuracy when it self-identifies as confident in its answer, whereas responses generated in the presence of knowledge gaps tend to be less accurate and more error-prone. Moreover, the accuracy gap between these two conditions is substantial.

In summary, our empirical evidence suggests that current models exhibit a certain level of knowledge boundary awareness and can actively identify the scope of their certainty under the ConfQA strategy. We argue that this capability is highly valuable for downstream tasks, especially in the context of the RAG framework. By combining internally confident knowledge with external sources, it is possible to achieve more reliable and higher-quality question answering.

D Generalization Study

To evaluate the adaptability and generalization ability of our proposed framework across language models of varying scales, we further conduct knowledge boundary awareness experiments

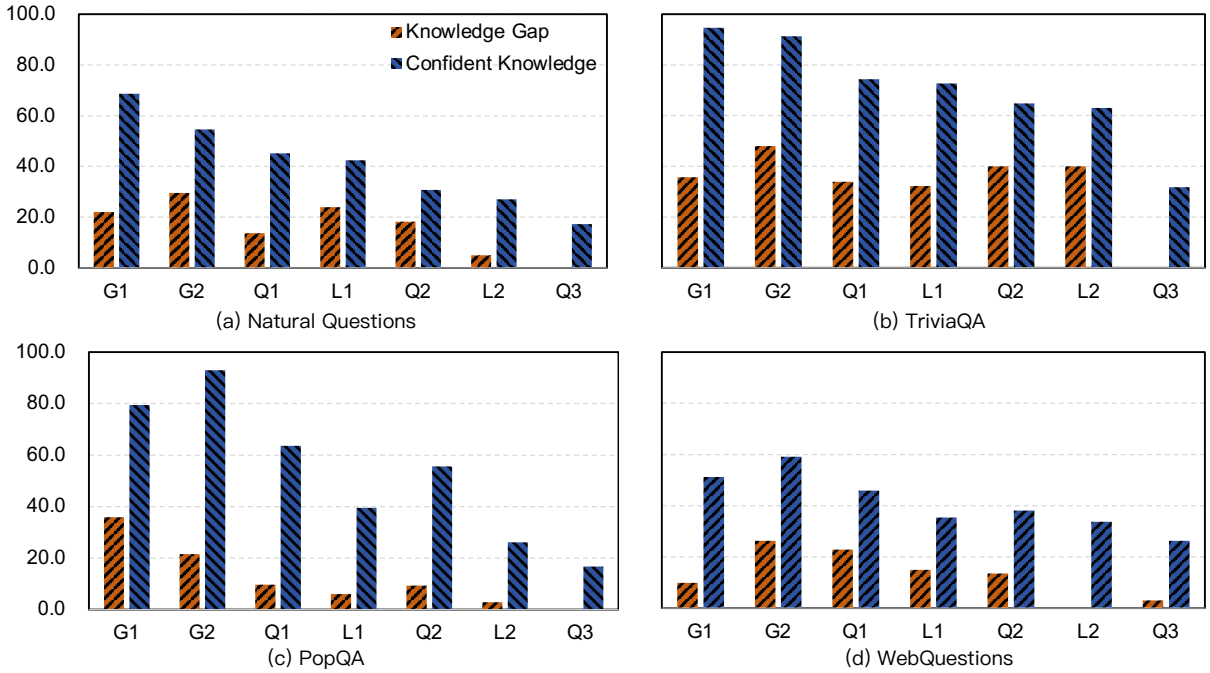

Figure 10: Accuracy comparison of different models on Confident Knowledge and Knowledge Gap samples. The vertical axis represents the F1 score, and the horizontal axis corresponds to the different models.

on multiple large language models. Given that our framework relies on a model's ability to perceive its own knowledge boundaries in order to actively decide whether to retrieve external information, we aim to investigate whether models other than GPT-4o also possess this capability, enabling effective integration with the framework. The experimental settings were kept consistent with those described in the previous section to ensure comparability. Specifically, we evaluated the output consistency of different models under the ConfQA and DirQA configurations, with the results summarized in Figure 11. Furthermore, we measured the models' accuracy on two distinct types of samples, namely Confident Knowledge and Knowledge Gap, as shown in Figure 10. In both figures, G1 corresponds to GPT-4o, G2 to GPT-4o-mini, Q1 to Qwen-2.5-14B-Instruct, L1 to Llama-3.1-8B-Instruct, Q2 to Qwen-2.5-7B-Instruct, L2 to Llama-3.2-3B-Instruct, and Q3 to Qwen-2.5-1.5B Instruct.

As illustrated in Figures 11 and 10, models with 7B parameters or more exhibit a clear sense of knowledge boundary awareness, demonstrating capabilities comparable to those of GPT-4o. For instance, Qwen-2.5-14B-Instruct and Llama-3.1-8B-Instruct show relatively stable performance under both the ConfQA and DirQA settings, with consistency rates exceeding 80%. This indicates their ability to maintain reliable knowledge judg-

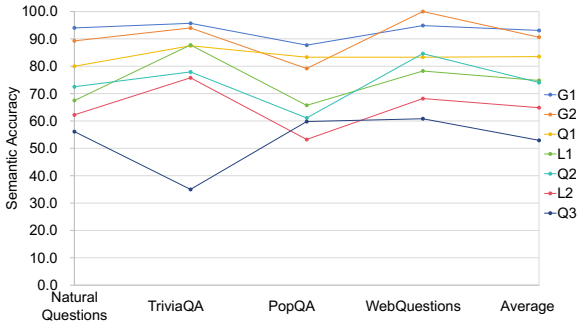

Figure 11: Comparative results of output consistency across different models under the DirQA and ConfQA settings. The vertical axis represents the semantic accuracy score, and the horizontal axis corresponds to the different datasets.

ments despite variations in question formulation. Moreover, their accuracy on Confident Knowledge instances approaches or exceeds 50% (with GPT-4o reaching approximately 70%), suggesting that these models can provide reliable responses when encountering familiar knowledge, while tending to express uncertainty when facing unfamiliar content. Such behavior is crucial for the effectiveness of our proposed framework, as basic knowledge boundary awareness is a prerequisite for the model to proactively trigger the retrieval module when necessary, thereby enabling more effective incorporation of external information.

However, models with fewer than 7B parameters exhibit several significant limitations.

- First, as illustrated in Figure 11, their output consistency under both the ConfQA and DirQA settings remains relatively low, with average semantic accuracy scores below 70. This indicates a strong sensitivity to prompt variations, often resulting in inconsistent responses, which is an undesirable characteristic for robust knowledge reasoning.
- Second, as shown in Figure 10, these models achieve low accuracy on Confident Knowledge samples, with mean F1 scores falling below 40, suggesting that responses produced with high confidence are frequently incorrect.
- Third, the difference in accuracy between Confident Knowledge and Knowledge Gap samples is minimal, implying that these models struggle to distinguish between known and unknown information.

In summary, we conclude that language models with 7B parameters or more are well-suited to our framework, exhibiting emerging capabilities in knowledge boundary awareness and behavioral patterns that align with those of GPT-4o. In contrast, models below the 7B parameter scale suffer from severe hallucination phenomena and possess an imprecise understanding of their own knowledge boundaries, thereby limiting their ability to collaborate effectively within the proposed framework.

E Datasets

E.1 Multi-hop Question Answering Datasets

HotpotQA (Yang et al., 2018) is a large-scale dataset featuring complex, multi-hop questions that require reasoning across multiple documents. It aims to improve question answering systems' multi-hop inference and answer interpretability.

2WikiMultiHopQA (Ho et al., 2020) includes complex multi-hop questions constructed from Wikidata. Models are required to integrate and reason over information from multiple Wikipedia documents to answer questions related to Wikipedia entities.

MuSiQue (Trivedi et al., 2022) constructs high-difficulty, multi-hop questions by compositing multiple single-hop questions. It aims to facilitate research and evaluation of multi-hop reasoning in question answering models.

Bamboogle (Press et al., 2023) is a small, manually curated dataset designed to evaluate language models' ability to handle compositional reasoning. It consists of two-hop questions that require effective combination and reasoning over disparate pieces of information.

Examples of these datasets could be found in Table 9.

Statistic	Value
# The data scale	9589
# The data scale of **Decomposition**	1900
# The average length of input instruction	252.4
# The average length of output	77.6
# The data scale of **Question Answering**	4821
# The average length of input instruction	358.4
# The average length of output	8.5
# The data scale of **Summarization**	2868
# The average length of input instruction	125.9
# The average length of output	25.9

Table 5: Statistics of the instruction-following synthetic dataset.

E.2 Single-hop Question Answering Datasets

Natural Questions (Kwiatkowski et al., 2019) comprises real user queries submitted to Google Search, primarily focused on content from Wikipedia pages and covering domains such as news and general knowledge.

TriviaQA (Joshi et al., 2017) dataset contains real-world trivia questions accompanied by supporting documents retrieved from Wikipedia and web search results. It is designed to enhance the capabilities of machine reading comprehension and question answering systems.

PopQA (Mallen et al., 2023) leverages fact triples derived from Wikidata to generate natural language questions spanning various relation types. It serves to evaluate question answering systems on their ability to understand and reason over entity-centric information.

WebQuestions (Berant et al., 2013) consists of natural language questions posed by real users, with answers grounded in the Freebase knowledge base. It is widely used for tasks in knowledge base question answering and semantic parsing.

Examples of these datasets could be found in Table 10.

F Training Data and Training

F.1 Training Data

To enhance the capabilities of models in complex instruction following and knowledge boundary

Statistic	Value
# The data scale	3096
# The data scale of **Confident Knowledge**	1267
# The average length of input instruction	150.1
# The average length of output	3.9
# The data scale of **Knowledge Gap**	1829
# The average length of input instruction	147.1
# The average length of output	4.0

Table 6: Statistics of the knowledge boundary-awareness synthetic dataset.

awareness, we design and construct two targeted datasets. These datasets are respectively aimed at improving the model's instruction-following ability and its capacity to recognize the limits of its own knowledge.

Given the current limitations of models in instruction execution accuracy and task comprehension, we constructed a set of training samples specifically designed to strengthen their instruction-following capabilities. The construction process is as follows: we randomly sampled 2,000 training examples each from the HotpotQA and 2WikiMultihopQA datasets, and processed these examples using our framework. If a model achieved a perfect prediction on a question (i.e., F1 = 1.0), the key intermediate outputs generated during reasoning were retained as training data. These include decomposed sub-instructions, question-answer pairs grounded in the provided documents, and extracted summaries from tree-structured reasoning steps. Importantly, to avoid injecting excessive factual knowledge into the model, we excluded data related to answers derived from the model's internal knowledge and retained only the parts that reflect task understanding and procedural execution. Table 5 presents the statistics of the training data. Table 12 presents several representative training examples constructed through this process.

Specifically, both in the training data and during inference, we use the same prompt format for two types of reasoning steps: answering sub-questions based on the provided documents, and answering complex questions using the summarized tree-structured information produced in the final step of our framework. This is because both types of questions are answered solely based on given information.

In addition, to improve the model's ability to recognize knowledge gap, we built a separate set of training samples focused on developing knowledge

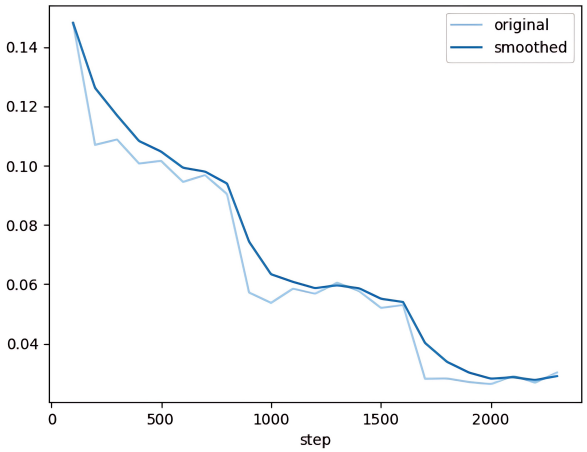

Figure 12: Training loss.

boundary awareness. Specifically, we randomly sampled 1,000 examples each from the training sets of the Natural Questions, PopQA, and WebQuestions datasets. Using the DirQA setting, the model was prompted to answer each question directly. When the model produces a correct answer, we combine the question with the ConfQA template and prompt the model to continue answering. If the model gives an incorrect answer, we still combine the question with the ConfQA template, but instruct the model to output a specific retrieval indicator, RAG_REQUIRED. These two types of training samples generated in this manner are used for fine-tuning the model's knowledge boundary-awareness capability. This approach aims to equip the model with the ability to accurately detect and articulate its knowledge limitations when faced with potentially underspecified or out-of-scope queries, thereby enhancing its robustness and trustworthiness. Table 6 presents the statistics of the training data. Table 13 shows several concrete examples of the generated training samples.

F.2 Training

We conduct model training using the LLaMA-Factory (Zheng et al., 2024) framework. Based on prior empirical insights, the LoRA-related hyperparameters are set as follows: the LoRA rank is set to 16 and the LoRA alpha to 32. During training, the learning rate is set to 5e-5, with 3 epochs and a batch size of 2. Gradient accumulation is applied with an accumulation step of 8. The training is performed on an NVIDIA RTX A6000 GPU and lasts approximately four hours. The training loss over time is illustrated in Figure 12.

G Semantic Accuracy Evaluation

Semantic accuracy is widely used in the evaluation of RAG systems. Consistent with existing work, we employ LLMs to assess semantic accuracy. The task is defined as follows: given a question, the model's predicted answer, and the reference answer, the LLM must determine whether the predicted answer can imply the reference answer and output only **Yes** or **No**. The prompt we used is shown below.

Prompt for Semantic Accuracy Evaluation

In the following task, you are given a Question, a model Prediction for the Question, and a Ground-truth Answer to the Question. You should decide whether the model Prediction implies the Ground-truth Answer.

Question
{question}

Prediction
{prediction}

Ground-truth Answer
{answer}

Does the Prediction imply the Ground-truth Answer? Output Yes or No and do not output any other words:

Unlike previous studies that employed GPT-series models for semantic accuracy evaluation, we utilized Llama-3.1-8B-Instruct as an alternative to GPT-4o for this task due to computational resource constraints. The significant performance gap that once existed between GPT-series models and other models has notably diminished, with major LLMs now demonstrating increasingly comparable capabilities. This observation led us to hypothesize that for relatively simple and well-defined tasks, evaluation decisions should be largely consistent across different models.

To test this hypothesis, we designed a comparative experiment: First, we randomly sampled 100 instances from each of the four test sets generated by the model (400 samples in total). These samples were then evaluated for semantic accuracy by three distinct models: GPT-4o, Qwen-2.5-7B-Instruct, and Llama-3.1-8B-Instruct. Subsequently, we employed statistical analysis to measure the agreement between the two open-source models and GPT-4o.

The experimental results presented in Table 7 demonstrate a high degree of consistency between the open-source models and GPT-4o, with Qwen-2.5-7B-Instruct achieving 96.75% agreement and Llama-3.1-8B-Instruct reaching 96.25%. These results strongly support our initial hypothesis, demonstrating that under resource-constrained conditions, replacing GPT-4o with an open-source model for semantic accuracy evaluation is both a feasible and effective solution. Considering the minimal performance gap between the two open-source models and aiming to maintain consistency with the model used in our main experiments, we employed Llama-3.1-8B-Instruct for the evaluation.

Dataset	Data Size	GPT-Qwen	GPT-Llama
HotpotQA	100	97.00	97.00
2WikiMultihopQA	100	97.00	96.00
MuSiQue	100	95.00	94.00
Bamboogle	100	98.00	98.00
Average	100	96.75	96.25

Table 7: The results of the consistency comparison in semantic accuracy evaluation among the Llama, Qwen, and GPT models.

H Reasoning Continuation Mechanism

H.1 Details

To prevent interruptions in the reasoning chain during the question-answering process, we introduce a Reasoning Continuation Mechanism. When the model fails to identify the answer to a sub-question within the retrieved context, this mechanism activates a direct inquiry strategy based on a Direct QA template (see Table 17), prompting the language model to provide an answer directly. Specifically, we define a set of trigger words, including "found", "mention", "provide" and so on. If the model's response contains a negation (e.g., "not") alongside any of these trigger words, it indicates that relevant information for the sub-question is not found in the current document. In such cases, the system bypasses the current retrieval content and directly queries the model using the Direct QA template to generate an alternative answer, thereby preserving the continuity of the reasoning chain.

According to our statistics (see Figure 13), this mechanism is triggered in approximately 5% of the samples. Although the resulting answers are

Models	2WikiMultihopQA			Bamboogle		
	F1	EM	Acc†	F1	EM	Acc†
w/	64.2	53.8	67.2	73.9	65.6	76.0
w/o	60.9 ($\downarrow_3$)	49.8 ($\downarrow_4$)	62.8 ($\downarrow_4$)	71.7 ($\downarrow_2$)	63.2 ($\downarrow_2$)	72.8 ($\downarrow_3$)

Table 8: The results of the ablation study for reasoning continuation mechanism.

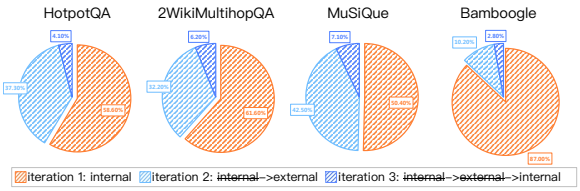

Figure 13: The proportion of iterations the model uses to solve each sub-question. An iteration count of 1 indicates that the model chooses to answer using confident knowledge; an iteration count of 2 indicates that the model opts for a RAG-based response; and an iteration count of 3 indicates that the model, unable to answer through RAG, resorts to a direct response.

not supported by external retrievals, they serve as a suboptimal yet effective strategy in scenarios of information absence, significantly enhancing the robustness of the system and ensuring the continuity of multi-step reasoning.

H.2 Ablation Study

To evaluate the specific impact of the Reasoning Continuation Mechanism on system performance, we conducted an ablation study by removing this module from the experimental framework. In this setting, when the model determines that the retrieved documents do not support answering the question, the system no longer performs any further processing or generates an answer based on the model's own knowledge; instead, it terminates the current reasoning process.

We tested the ablated system on two multi-hop question answering datasets: 2WikiMultihopQA and Bamboogle. The experimental results indicate a slight performance degradation: a drop of 11 points on 2WikiMultihopQA and 7 points on Bamboogle (see Table 8). Although the performance decline is relatively modest, the results suggest that the Reasoning Continuation Mechanism plays a supportive role in maintaining overall system effectiveness.

We argue that while current models tend to adopt a RAG approach under the Confident QA setting, there are scenarios in which the model's internal knowledge may already contain the correct answer. Therefore, when external documents fail to provide the necessary information for a sub-question, leveraging the model's internal knowledge to continue generating an answer helps preserve the integrity of the reasoning chain and contributes to improved system stability and robustness.

I Error Analyses

Although the framework proposed in this study demonstrates strong performance in most scenarios, failures may still occur under certain conditions.

Internal Knowledge Error: Despite the high accuracy exhibited by large language models such as GPT-4o in most tasks, they are still prone to factual errors in their generated outputs. When internal knowledge is misrepresented or flawed, the model may produce inaccurate content, thus compromising the reliability of the overall framework. For instance, as shown in Table 15, when addressing the question "*How did Nora Brockstedt die?*", the model generated an incorrect answer, leading to a failure in solving the problem.

Overall, due to Internal Knowledge Error, our framework may still produce errors.

J Additional Details

To facilitate a deeper understanding of our proposed framework, we provide additional materials as follows.

Table 11 offers a comprehensive comparison between our approach and the baselines, summarizing the key characteristics of each method to highlight the distinctions and advantages of our design.

Table 14 presents a representative inference example of our framework on the 2WikiMultihopQA dataset, illustrating how the model processes multi-hop reasoning in a real scenario.

Table 16 presents several examples of decomposed questions.

Dataset	Type	Num	Question	Answer
HotpotQA	bridge	386	Geoff LaTulippe is an American writer whose best-known work was directed by whom?	Nanette Burstein
	comparison	114	Which lake is located further south, Dal Lake or Waterton Lake?	Dal Lake
2WikiMultihopQA	comparison	123	Which film came out earlier, Watermark (Film) or Sofia'S Last Ambulance?	Sofia'S Last Ambulance
	compositional	201	Where was the performer of song Feelin' Myself (Nipsey Hussle Song) born?	Crenshaw
	inference	69	Who is Duke Siegfried August In Bavaria's maternal grandmother?	Princess Clémentine of Orléans
	bridge comparison	107	Which film has the director who was born earlier, Hostage For A Day or Buck And The Preacher?	Buck And The Preacher
MuSiQue	2-hop	283	Who is the spouse of the creator of Absolutely Fabulous?	Adrian Edmondson
	3-hop	150	Who is the owner of the record label that the performer of Trojans belongs to?	Warner Music Group
	4-hop	67	What is the capital of the county that shares a border with the county where KRSU is licensed to broadcast?	Green Bay
Bamboogle	2-hop	125	Who was president of the United States in the year that Citibank was founded?	james madison

Table 9: Details for multi-hop QA datasets used for evaluation.

Dataset	Question	Answer
Natural Questions	how many seasons of prison break are on netflix	five
TriviaQA	Which Scotsman became the first European to reach the River Niger in 1796?	Mungo Park
PopQA	Who was the screenwriter for A Teacher?	Hannah Fidell
WebQuestions	who does ronaldinho play for now 2011?	Clube de Regatas do Flamengo

Table 10: Examples for single-hop QA datasets.

Method	Iterative	Dynamic	Retrieval Strategy	Retrieval Query
Direct QA	No	No	-	-
CoT	No	No	-	-
OneR	No	No	-	-
RetGen	Yes	No	-	-
Self-Ask	Yes	No	-	-
FLARE	Yes	Yes	Token probability in model output sentence	Tokens with probability above threshold
CompAct	Yes	No	-	-
GenGround	Yes	No	-	-
DRAGIN	Yes	Yes	Real-time information needs	Query generation based on self-attention
DyPlan	Yes	Yes	Strategy selection based on question	Complex input question
Search-R1	Yes	No	-	-
TAIKC	Yes	Yes	Model's knowledge boundary awareness	Sub-questions decomposed from complex questions

Table 11: Overview of methods and their use of iterative and dynamic retrieval strategies.

Input: You are an AI assistant adept at dissecting intricate queries into a series of straightforward, logically interlinked questions, and adeptly representing each of these as discrete objects within a JSON structure. Decompose the following complex question into smaller, logically connected queries. Each query should be represented as a separate object in a JSON format. For questions that require further follow-up (e.g., sub-queries about the results), nest those queries under the parent query. Rules: 1. Each question should be split into logical, simpler questions. 2. Use "query1," "query2," etc., for the main and sub-queries. 3. If a sub-query depends on the answer to a previous query, use a reference in the form of "#queryX" (e.g., "What is the birth year of #query1?") to indicate the dependency. The number should correspond to the query being referenced. 4. When decomposing a multi-hop question, only generate essential factual sub-questions needed to answer the main query, and avoid any comparison, intersection, or synthesis questions. Decompose the following question without any additional explanation or words. Are Han Sai Por and Gabriele Oettingen of the same nationality?
Output: { "query1": { "question": "What is the nationality of Han Sai Por?", "children": { "query2": { "question": "What is the nationality of Gabriele Oettingen?" } } }}

Input: You will receive a question and several documents. The following are given documents: passage: Princess Gabriele of Oettingen-Oettingen and Oettingen-Spielberg, known professionally as Gabriele Oettingen, (born G̈abriele Elisabeth Aloisia Notgera Prinzessin zu Oettingen-Oettingen und Oettingen-Spielberg¨; July 22, 1953, Munich) is a German academic and psychologist. She is a professor of psychology at New York University and the University of Hamburg. Her research focuses on how people think about the future, and how this impacts cognition, emotion, and behavior. passage: Oettingen-Spielberg is a noble family and former principality in modern-day eastern Baden-Württemberg and western Bavaria, Germany. passage: Oettingen was born on July 22, 1953 in Munich, Germany to Alois Philipp Joseph Maria Notger, 9th Prince of Oettingen-Oettingen and Oettingen-Spielberg and Countess Elisabeth Gabriele zu Lynar. She is a member of the German princely House of Oettingen-Spielberg. passage: Alexander Konstantin von Oettingen (–) was a Baltic German Lutheran theologian and statistician. passage: Oettingen lives and works in New York City and Munich. On August 10, 1990 she married Peter Gollwitzer. They had a religious ceremony on January 22, 1994. They have two children, Anton and Jakob. Question: What is the nationality of Gabriele Oettingen? Please provide the answer directly and do not output any other words:
Output: German

Input: Summarize the information provided in the json text, which consists of several query-and-answer pairs. Please do not omit any information in the question-and-answer pairs. Summarize the following query-and-answer pairs into a sentence: {"query1": {"question": "What is the nationality of Han Sai Por?", "answer": "Singaporean", "children": {"query2": {"question": "What is the nationality of Gabriele Oettingen?", "answer": "German"}}}} Output your summary below:
Output: Han Sai Por is Singaporean, while Gabriele Oettingen is German.

Input: You will receive a question and several documents. The following are given documents: Han Sai Por is Singaporean, while Gabriele Oettingen is German. Question: Are Han Sai Por and Gabriele Oettingen of the same nationality? Please provide the answer directly and do not output any other words:
Output: No.

Table 12: Examples for instruction following training data. The red text indicates the core requirements of the task, the blue text represents the data to be processed, and the teal text denotes the resulting output.

Input: Next, I will provide a question. Please output only the exact answer to the question directly, without adding any additional explanations, guiding phrases, or background information. If you are uncertain about the answer or unable to provide an exact answer directly, please respond with "RAG_REQUIRED" so that I can proceed with further information retrieval (RAG). If there is any uncertainty about the correctness of your answer, even slightly, you must respond with "RAG_REQUIRED." Only provide an answer if you are absolutely certain it is correct. Here is the question you need to handle: secretary of state for communities and local government of the united kingdom Please provide the answer directly and do not output any other words:
Output: RAG_REQUIRED

Input: Next, I will provide a question. Please output only the exact answer to the question directly, without adding any additional explanations, guiding phrases, or background information. If you are uncertain about the answer or unable to provide an exact answer directly, please respond with "RAG_REQUIRED" so that I can proceed with further information retrieval (RAG). If there is any uncertainty about the correctness of your answer, even slightly, you must respond with "RAG_REQUIRED." Only provide an answer if you are absolutely certain it is correct. Here is the question you need to handle: who played tom cruises wife in the firm Please provide the answer directly and do not output any other words:
Output: Jeanne Tripplehorn

Input: Next, I will provide a question. Please output only the exact answer to the question directly, without adding any additional explanations, guiding phrases, or background information. If you are uncertain about the answer or unable to provide an exact answer directly, please respond with "RAG_REQUIRED" so that I can proceed with further information retrieval (RAG). If there is any uncertainty about the correctness of your answer, even slightly, you must respond with "RAG_REQUIRED." Only provide an answer if you are absolutely certain it is correct. Here is the question you need to handle: what countries are located in the netherlands? Please provide the answer directly and do not output any other words:
Output: RAG_REQUIRED

Input: Next, I will provide a question. Please output only the exact answer to the question directly, without adding any additional explanations, guiding phrases, or background information. If you are uncertain about the answer or unable to provide an exact answer directly, please respond with "RAG_REQUIRED" so that I can proceed with further information retrieval (RAG). If there is any uncertainty about the correctness of your answer, even slightly, you must respond with "RAG_REQUIRED." Only provide an answer if you are absolutely certain it is correct. Here is the question you need to handle: what type of cancer did donna summer died from? Please provide the answer directly and do not output any other words:
Output: Lung cancer

Table 13: Examples for knowledge boundary recognition training data. The red text indicates the core requirements of the task, the blue text represents the data to be processed, and the teal text denotes the resulting output.

An Example of Reasoning on 2WikiMultihopQA Dataset

Question:
Which film has the director who is older than the other, The Carousel Of Death or Nameless Star?

Decompose:

```
{
  "query1": {
    "question": "Who directed the film The Carousel Of Death?",
    "children": {
      "query2": {
        "question": "What is the birth year of #query1?"
      }
    }
  },
  "query3": {
    "question": "Who directed the film Nameless Star?",
    "children": {
      "query4": {
        "question": "What is the birth year of #query3?"
      }
    }
  }
}
```

Answer:
Query1: Who directed the film The Carousel Of Death?
Confident QA: RAG_REQUIRED
RAG: Heinz Paul
Query2: What is the birth year of Heinz Paul?
Confident QA: RAG_REQUIRED
RAG: 1918
Query3: Who directed the film Nameless Star?
Confident QA: Mihail Kozakov
Query4: What is the birth year of Mihail Kozakov?
Confident QA: 1934

Summarize:
The film "The Carousel Of Death" was directed by Heinz Paul, who was born in 1918.
The film "Nameless Star" was directed by Mihail Kozakov, who was born in 1934.

Predict:
The Carousel Of Death ✓

Ground truth:
The Carousel Of Death

Table 14: An example of reasoning on 2WikiMultihopQA dataset.

An Example of Internal Knowledge Error for Error Analyses

Question:
Why did the performer of song Sommer I Palma die?

Decompose:

```
{
  "query1": {
    "question": "Who is the performer of the song Sommer I Palma?",
    "children": {
      "query2": {
        "question": "How did Nora Brockstedt die?"
      }
    }
  }
}
```

Answer:
Query1: Who is the performer of the song Sommer I Palma?
Confident QA: RAG_REQUIRED
RAG: Nora Brockstedt
Query2: How did Nora Brockstedt die?
Confident QA: Natural causes. ✗

Summarize:
Nora Brockstedt is the performer of the song "Sommer I Palma" and she died of natural causes.

Predict:
natural causes ✗

Ground truth:
illness

Table 15: An example of internal knowledge error for error analyses. The red text indicates that an error has occurred with the problem and the module.

Demonstrations of Question Decomposition

Which magazine was started first Arthur's Magazine or First for Women?

```
{
  "query1": {
    "question": "When was the magazine Arthur's Magazine started?"
    },
  "query2": {
    "question": "When was the magazine First for Women started?"
    }
}
```

What nationality was James Henry Miller's wife?

```
{
  "query1": {
    "question": "Who was James Henry Miller's wife?",
    "children": {
      "query2": {
        "question": "What is the nationality of #query1?"
      }
    }
}
```

Which film whose director is younger, Charge It To Me or Danger: Diabolik?

```
{
  "query1": {
    "question": "Who directed the film Charge It To Me?",
    "children": {
      "query2": {
        "question": "What is the birth year of #query1?"
      }
    }
  },
  "query3": {
    "question": "Who directed the film Danger: Diabolik?",
    "children": {
      "query4": {
        "question": "What is the birth year of #query3?"
      }
    }
  }
}
```

Table 16: Demonstrations of question decomposition.

Prompt for Direct QA and Confident QA

Prompt for Direct QA:

Next, I will provide a question. Please output only the exact answer to the question directly, without adding any additional explanations, guiding phrases, or background information.

Here is the question you need to handle:
{question}

Please provide the answer directly and do not output any other words:

Prompt for Confident QA:

Next, I will provide a question. Please output only the exact answer to the question directly, without adding any additional explanations, guiding phrases, or background information.

If you are uncertain about the answer or unable to provide an exact answer directly, please respond with "RAG_REQUIRED" so that I can proceed with further information retrieval (RAG).

If there is any uncertainty about the correctness of your answer, even slightly, you must respond with "RAG_REQUIRED." Only provide an answer if you are absolutely certain it is correct.

Here is the question you need to handle:
{question}

Please provide the answer directly and do not output any other words:

Table 17: Prompt for Direct QA and Confident QA.