Debate over Mixed-knowledge: A Robust Multi-Agent Reasoning Framework for Incomplete Knowledge Graph Question Answering

Jilong Liu[1] , Pengyang Shao[* 2], Wei Qin[1] , Fei Liu[1 3] , Yonghui Yang[2] , Richang Hong[* 1]
[1]Hefei University of Technology
[2]National University of Singapore
[3]The Key Laboratory of Knowledge Engineering with Big Data (the Ministry of Education of China)
{liujilong0116, shaopymark, qinwei.hfut}@gmail.com,feiliu@mail.hfut.edu.cn, {yyh.hfut, hongrc.hfut}@gmail.com

Abstract

Knowledge Graph Question Answering (KGQA) aims to improve factual accuracy by leveraging structured knowledge. However, real-world Knowledge Graphs (KGs) are often incomplete, leading to the problem of Incomplete KGQA (IKGQA). A common solution is to incorporate external data to fill knowledge gaps, but existing methods lack the capacity to adaptively and contextually fuse multiple sources, failing to fully exploit their complementary strengths. To this end, we propose Debate over Mixed-knowledge (DoM), a novel framework that enables dynamic integration of structured and unstructured knowledge for IKGQA. Built upon the Multi-Agent Debate paradigm, DoM assigns specialized agents to perform inference over knowledge graphs and external texts separately, and coordinates their outputs through iterative interaction. It decomposes the input question into sub-questions, retrieves evidence via dual agents (KG and Retrieval-Augmented Generation, RAG), and employs a judge agent to evaluate and aggregate intermediate answers. This collaboration exploits knowledge complementarity and enhances robustness to KG incompleteness. In addition, existing IKGQA datasets simulate incompleteness by randomly removing triples, failing to capture the irregular and unpredictable nature of real-world knowledge incompleteness. To address this, we introduce a new dataset, Incomplete Knowledge Graph WebQuestions, constructed by leveraging real-world knowledge updates. These updates reflect knowledge beyond the static scope of KGs, yielding a more realistic and challenging benchmark. Through extensive experiments, we show that DoM consistently outperforms state-of-the-art baselines.

Introduction

Recent advances in Knowledge Graph-based Question Answering (KGQA) have shown that augmenting larges language models (LLMs) with structured, semantically rich knowledge graphs (KGs) can improve the factual reliability of model outputs (Luo et al. 2024b; Chen et al. 2024; Tan et al. 2025; Ma et al. 2025). These methods typically retrieve KG subgraphs relevant to the input query and feed them into the LLM in a multi-step manner, thereby improving answer quality (Sun et al. 2024). Although effective, these methods tend to rely on the completeness of the underlying knowledge graphs—conditions that are difficult to satisfy in practice due to the high cost of KG construction and maintenance (Hur, Janjua, and Ahmed 2021). Previous studies have recognized the challenge of KG incompleteness (Min et al. 2013; Ren, Hu, and Leskovec 2020; Pflueger, Tena Cucala, and Kostylev 2022), which has led to the emergence of **Incomplete Knowledge Graph Question Answering (IKGQA)** as a distinct research task.

To address the IKGQA issue, existing approaches can be broadly categorized to: (i) KG-internal completion, and (ii) external information augmentation. For category (i), KG-internal completion methods typically aim to predict missing links by learning embedding representations of entities and relations, modeling relational patterns among existing triples within the KG (Saxena, Kochsiek, and Gemulla 2022; Guo et al. 2023). However, such methods inherently rely on existing KG structure and thus struggle to capture changes in the external world, as real-world KG incompleteness often arises from evolving events. Category (ii) includes methods that mitigate KG incompleteness by incorporating knowledge sources beyond the KG itself. some approaches leverage external textual corpora, such as Wikipedia, to construct question-specific subgraphs and enrich the KG with supplementary contextual information (Sun, Bedrax-Weiss, and Cohen 2019; Lv et al. 2020). Others treat the parametric knowledge embedded in LLMs as an auxiliary information source, using it to infer missing triples (Xu et al. 2024). While these methods demonstrate the value of incorporating external information to mitigate KG incompleteness, they still exhibit notable limitations. Approaches that integrate structured and unstructured data typically rely on training models with substantial amounts of aligned data, making them sensitive to the quality and coverage of training resources. On the other hand, methods that rely exclusively on the parametric knowledge encoded in LLMs may yield incorrect inferences when the missing KG facts fall beyond the coverage of the LLMs' pre-training. These limitations raise a central question: How to leverage inference capabilities of LLM to better integrate mixed knowledge for the IKGQA issue?

To this end, we propose Debate over Mixed-knowledge (DoM). Given the need to effectively integrate external and KG-based knowledge, we adopt the Multi-Agent Debate

*Corresponding authors

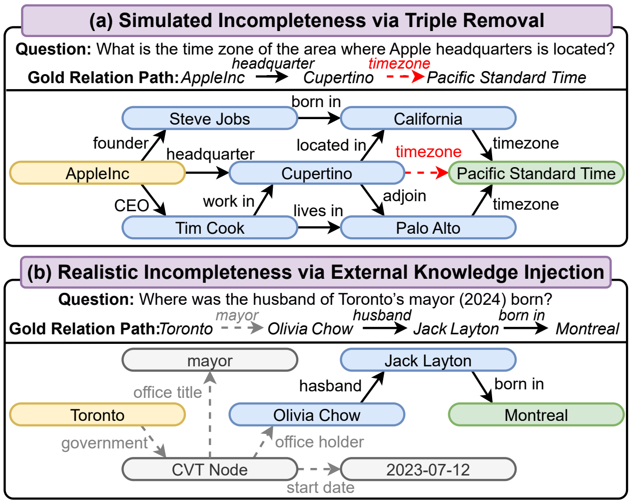

Figure 1: Illustration of two strategies for constructing incomplete KG scenarios. (a) Simulated incompleteness by removing triples from the KG; red dashed arrows denote the deleted facts. (b) Realistic incompleteness by incorporating external knowledge; gray dashed arrows indicate newly introduced information. CVT (Compound Value Type) nodes are used to represent multi-entity relations in Freebase.

(MAD) framework, where independent agents specialize in performing inference over different knowledge sources and collaboratively refine their outputs through interaction and alignment. Specifically, to effectively combine MAD with IKGQA, we introduce a three-stage framework : (1) Initialization: to prepare for effective coordination of mixed-knowledge, we decompose the input question into semantically coherent sub-questions. These sub-questions are dynamically updated throughout the inference process. (2) Sub-question Inference: to integrate heterogeneous knowledge, we design dual retrieval agents, including a KG Agent for structured KG and a Retrieval-Augmented Generation (RAG) Agent for unstructured external knowledge. Each agent independently retrieves candidate evidence, and a Judge Agent evaluates and integrates their outputs to produce an intermediate answer and update the inference plan. This design enables mutual correction and complementarity between knowledge sources. (3) Final Answer Generation: to ensure global consistency, DoM prompts an LLM to consolidate intermediate results into a final answer. However, existing IKGQA datasets typically simulate incompleteness by randomly removing gold-path triples. This synthetic pattern fails to capture the irregular and evolving nature of real-world incomplete KG scenarios. To address this limitation, we construct a new dataset, Incomplete Knowledge Graph WebQuestions (IKGWQ), by revisiting existing QA benchmarks (CWQ and WebQSP) and regenerating question-answer pairs using up-to-date knowledge. This design naturally introduces both missing triples and missing entities, thereby capturing the evolving nature of real-world KG incompleteness. Finally, extensive experiments demonstrate the effectiveness of our proposed DoM. Our main contributions are summarized as follows:

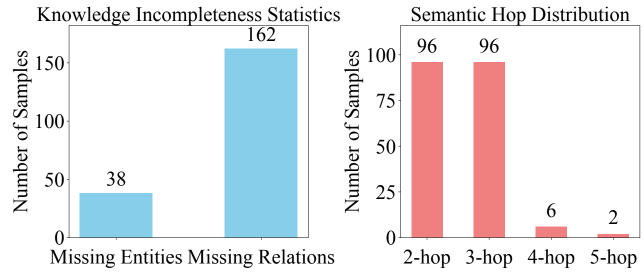

Figure 2: Statistics of knowledge incompleteness and semantic hop distribution in the IKGWQ dataset. The left subfigure shows the number of samples with missing entities and missing relations. The right subfigure presents the distribution of semantic inference hops.

1. To better integrate multi-source knowledge to address the IKGQA challenge, we propose a multi-agent debate framework, Debate over Mixed-knowledge (DoM), which enables dynamic and complementary inference from incomplete KGs and external knowledge.
2. We construct a new dataset, IKGWQ, addressing the limitations of existing IKGQA datasets by introducing real-world factual updates, better reflecting the irregularity and unpredictability of KG incompleteness.
3. DoM achieves consistent gains over strong baselines, with up to 13.6% relative improvement in Hits@1 on existing IKGQA datasets and 70.7% on IKGWQ.

Data Description

Existing IKGQA datasets often simulate incompleteness by removing triples from gold relation paths, as illustrated in Figure 1(a). To better capture real-world KG dynamics, we construct IKGWQ by revisiting samples from CWQ and WebQSP and rebuilding corresponding question-answer using up-to-date knowledge, as shown in Figure 1(b).

Specifically, we revisit each selected sample by retrieving up-to-date information for its topic entity from reliable sources and constructing question-answer pairs that reflect facts missing or outdated in the original Freebase (Bollacker et al. 2008) KG. First, we extract topic entities from the original datasets and retrieve their updated descriptions via the Wikipedia API. We then guide an LLM to generate question-answer pairs based on knowledge extracted from these texts—focusing particularly on facts beyond the scope of the original KG. Finally, we conduct manual filtering, question rewriting, and human annotation to ensure the quality and correctness of the resulting dataset. This pipeline naturally introduces both missing triples and missing entities, offering a more realistic simulation of the dynamic and uncertain nature of KG incompleteness in real-world scenarios.

To better understand the characteristics and challenges posed by the IKGWQ dataset, we conduct a detailed analysis in two key dimensions: knowledge incompleteness and inference complexity. The results are summarized in Figure 2. The dataset comprises 200 samples, including 38 instances of missing entities and 162 instances of missing relations. Note that missing-entity cases inherently subsume relation incompleteness, whereas missing-relation cases do not involve any entity omission. In terms of inference complexity, a substantial portion of questions require multi-hop inference: 48% of the samples involve 3-hop inference, with some extending up to 5-hop. Note that hop count is defined based on the number of semantic inference steps needed to answer a question, excluding auxiliary nodes such as Compound Value Type nodes in Freebase. The actual KG traversal steps may be higher due to these intermediate structures.

Preliminary

Incomplete Knowledge Graph Question Answering

IKGQA is a generalization of the standard KGQA task. In KGQA, the goal is to predict the correct answer entities $A_q \subseteq E$ for a given natural language question q, based on a complete knowledge graph $G = (E, R, T)$. Here, E denotes the set of entities, R denotes the set of relations, and $T = (e_h, r, e_t) \mid e_h, e_t \in E, r \in R$ represents the set of factual triples. Each triple consists of a head entity e_h, a relation r, and a tail entity e_t. This task assumes that all topic entities $T_q \subseteq E$ and the necessary relation paths are present in G. Formally, KGQA can be defined as a function $f : (q, G) \mapsto A_q$.

In real-world applications, however, KGs are often incomplete due to limitations in construction and maintenance. To address this, IKGQA relaxes the requirement for full KG coverage by enabling the reasoning process to leverage external knowledge—either retrieved from textual sources or derived from the internal knowledge embedded in LLMs. Following GoG (Xu et al. 2024), IKGQA can be formulated as $f : (q, G, \mathcal{R}) \mapsto A_q$, where $\mathcal{R}$ denotes external knowledge that supplements the incomplete KG G during the inference process.

Notation for LLM Modules

We denote the outputs of task-specific LLM calls using the notation $\text{LLM}_{\text{role}}(\cdot)$, where the subscript indicates the role or function performed by the LLM in the inference process. For example, LLM_{plan} represents the planning module for sub-question decomposition.

Debate over Mixed-knowledge (DoM)

The workflow of DoM is showed in Figure 3. At the inference stage, DoM adopts the MAD framework to enhance the utilization of mixed knowledge in IKG scenarios. It constructs a KG Agent and a RAG Agent to independently infer from structured KGs and unstructured external data. Their outputs are then aligned and integrated by a Judge Agent. This pipeline consists of three phases: **Initialization**, **Sub-question Inference**, and **Final Answer Generation**.

Initialization

Given a natural language question q, we prompt the LLM to decompose it into a sequence of semantically meaningful sub-questions, forming an ordered list Q:

$$Q = \text{LLM}_{plan}(q) = \{q_1, q_2, \ldots, q_n\}, \tag{1}$$

Each sub-question q_i is treated as an intermediate goal and will be addressed sequentially in later stages. This decomposition transforms a complex question into a controllable step-by-step inference trajectory, serving as the structural foundation for the subsequent multi-agent debate. We also initialize a system memory M to store intermediate inference results across sub-questions; its structure will be detailed in the final answer generation phase.

Sub-question Inference

After initialization, DoM enters the inference phase, iteratively integrating structured and unstructured knowledge to solve sub-questions. Each iteration consists of three main steps: **knowledge graph inference**, **external knowledge inference**, and **debate-based integration**.

Formally, at the i-th iteration, the current inference context is centered around a set of topic entities, denoted as $E_i^{\text{topic}} = \{\hat{e}_i^1, \hat{e}_i^2, \ldots, \hat{e}_i^m\}$. Based on these entities, the system retrieves:

- A set of KG relation paths $P_i = \{T_i^1, T_i^2, \ldots, T_i^m\}$, where each path $T_i^j = \{t_i^{j,1}, t_i^{j,2}, \ldots, t_i^{j,l}\}$ is a sequence of triples starting from entity $\hat{e}_i^j$. Each triple $t_i^{j,k} = (h, r, t)$ denotes a factual relation in the KG.
- A set of external textual evidence chunks $C_i = \{C_i^1, C_i^2, \ldots, C_i^o\}$ retrieved by the RAG Agent.

These two sources of knowledge are independently processed by the KG Agent and the RAG Agent. Each agent infers from its respective evidence and proposes a candidate answer to q_i. The Judge Agent integrates these outputs and determines the result, based on which the system updates the next sub-question q_{i+1}.

Knowledge Graph Inference This process focuses on answering the sub-question q_i using structured knowledge, facilitated by the **KG Agent**. This KG-based processing pipeline consists of two phases: **entity linking** and **iterative exploration**. Although E_i^{topic} may contain multiple entities, we select a representative $\hat{e}_i$ to illustrate these two steps.

Entity Linking In each iteration (except the first), the topic entities are derived from the previous sub-question's answer, which may originate from KG, external sources, or the LLM's internal knowledge. Since non-KG-originated entities may not align with the KG, we perform entity linking to map them to corresponding machine identifiers.

This process involves two steps: (1) retrieving top candidate KG entities via embedding-based name similarity search, and (2) prompting the LLM to select the most suitable entity based on relational context in KG.

Formally, for the ungrounded entity mention $\hat{e}_i$, we retrieve the top_k KG entities E_i' whose names are most similar to $\hat{e}_i$ in the embedding space, i.e., $E_i' = top_k(\text{sim}(E(\hat{e}_i), E(G)))$, where $E(\cdot)$ denotes the entity name encoder, $\text{sim}(\cdot, \cdot)$ denotes embedding similarity. Then, for each candidate entity in E_i', we retrieve its associated KG description. The final linked entity e_i is selected by the LLM:

$$e_i = \text{LLM}_{entity\_select}(\hat{e}_i, E_i'). \tag{2}$$

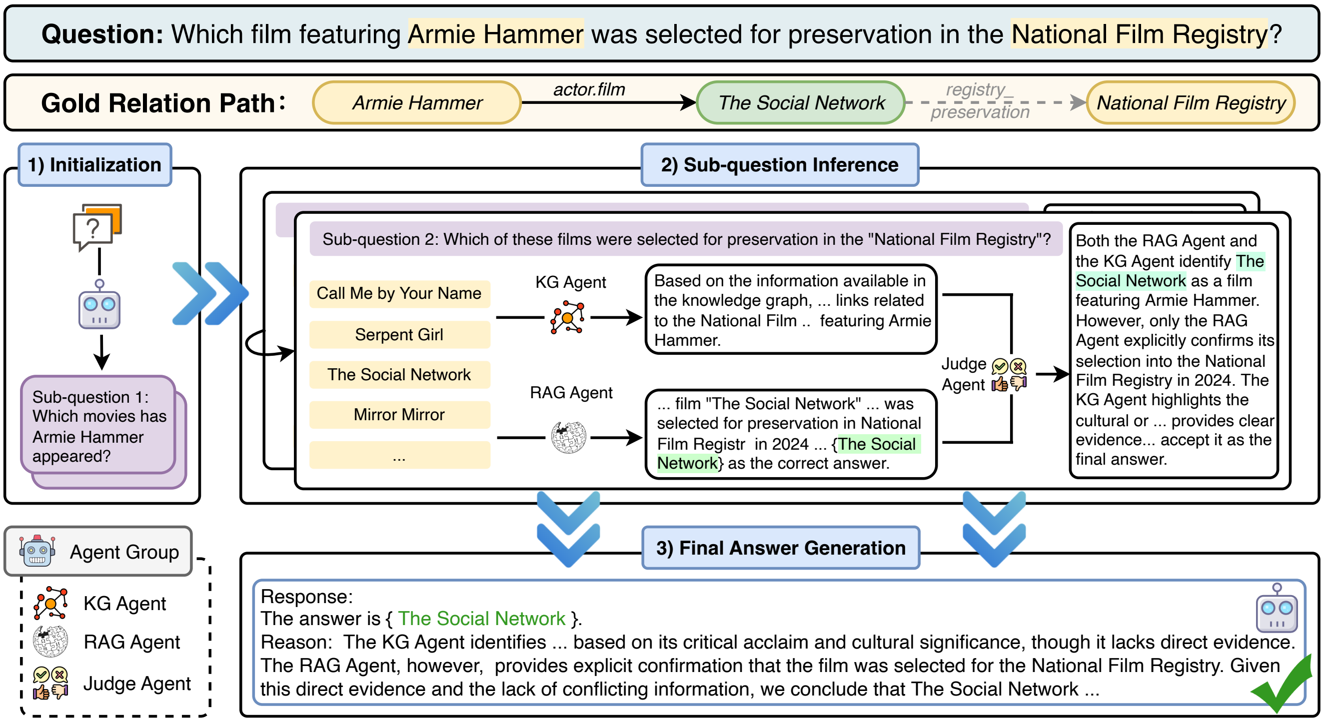

Figure 3: Overview of the DoM framework. DoM first decomposes the input question into sub-questions. For each sub-question, the KG Agent and RAG Agent independently infer over structured and unstructured knowledge, and the Judge Agent integrates their outputs through iterative debate. This interaction continues until sufficient evidence is gathered for final answer generation.

If all candidate entities are deemed unsuitable, we assume the target entity is missing from the KG. In this case, KG exploration is skipped for this sub-question, and the LLM resorts to internal CoT inference, as shown in Equation 5.

Iterative KG Exploration Once the entity e_i is linked to the KG, we initialize the KG inference process by setting $e_i^1 = e_i$, and perform an iterative search for supporting evidence. At the w-th KG inference step for q_i, we first retrieve all 1-hop outbound relations of the current entity e_i^w—including both outbound and inbound edges (i.e., inverse relations)—via SPARQL, denoted as $S_r(e_i^w)$. The LLM then selects the top_k relations most relevant to the current sub-question q_i:

$$R_i^w = \text{LLM}_{relation\_select}(S_r(e_i^w), q_i), \tag{3}$$

For each selected relation, we extract the associated triples via $S_t(e_i^w, R_i^w)$, and incrementally update the candidate evidence set as $P_i \leftarrow P_i \cup T_i^w$. The union of retrieved triples across iterations constitutes the KG relation path for sub-question q_i. The evidence set P_i is initialized as an empty set and progressively expanded over iterations.

The LLM determines whether the current evidence set P_i is sufficient to answer the sub-question q_i. If so, it directly produces the answer $\hat{a}_i^{kg}$; otherwise, it selects a new entity from P_i to update the topic entity e_i^{w+1} for the next iteration:

$$\text{LLM}_{kg\_inference}(P_i, q_i, M) = \begin{cases} \hat{a}_i^{kg}, & \text{sufficient;} \\ e_i^{w+1}, & \text{otherwise.} \end{cases} \tag{4}$$

The process continues iteratively until an answer is generated or a maximum of W steps is reached. If no answer is derived within W steps, or if entity linking fails, the LLM resorts to CoT inference:

$$\hat{a}_i^{kg} = \text{LLM}_{CoT}(q_i, M), w = W \text{ or } \hat{e}_i \text{ missing in KG.} \tag{5}$$

External Knowledge Inference This process aims to answer the sub-question q_i using external unstructured knowledge, facilitated by the **RAG Agent**. To ensure factual reliability, we adopt Wikipedia as the external knowledge source.

Given the topic entity $\hat{e}_i$, we retrieve the top-k most relevant Wikipedia articles, which are then segmented into text chunks to construct the candidate evidence set C_i. We then compute relevance scores between each chunk $c \in C_i$ and the sub-question q_i using an embedding model. The top-k scored chunks $C_i' = \{c_i^1, \ldots, c_i^k\}$ are selected as the contextual input for LLM-based inference.

Similar to the KG Agent, if the retrieved evidence is insufficient, the RAG Agent resorts to the LLM's internal knowledge via CoT inference to preserve the continuity of the inference process. We formalize the RAG Agent's inference process as follows: given the candidate evidence set C_i, the top-k relevant chunks are selected as $C_i' = top_k(\text{sim}(E(C_i), E(q_i)))$,

$$\hat{a}_i^{rag} = \begin{cases} \text{LLM}_{rag\_inference}(C_i', q_i, M), & \text{sufficient;} \\ \text{LLM}_{CoT}(q_i, M), & \text{otherwise.} \end{cases} \tag{6}$$

Debate-based Integration For each sub-question q_i, the KG Agent and RAG Agent independently generate answers accompanied by their respective inference chains. The **Judge Agent** then serves as an arbiter to evaluate, compare, and integrate these outputs, producing the consolidated sub-answer a_i.

A new iteration is triggered either when sub-questions remain, or when existing information is deemed insufficient to answer the original query. The current sub-answer a_i is used to update the topic entity set. If the next sub-question q_{i+1} exists, it is revised based on the context; otherwise, a new sub-question is adaptively generated. This step is defined as:

$$a_i, q_{i+1}^{new} = \text{LLM}_{judge}(q, q_i, q_{i+1}, \hat{a}_i^{kg}, \hat{a}_i^{rag}, M). \quad (7)$$

This mechanism enables a dynamic and adaptive inference loop, guided by accumulated evidence. To ensure termination, the number of sub-question iterations is capped by a predefined threshold I.

Final Answer Generation

After all sub-questions have been processed or the maximum number of inference iterations I has been reached, the system proceeds to generate the final answer. This step is based on the accumulated inference memory M, which records intermediate results from each sub-question in the form $M = [[q_1, \hat{a}_1^{kg}, \hat{a}_1^{rag}, a_1], ...]$. The final answer a_{final} is computed as:

$$a_{final} = \begin{cases} \text{LLM}_{verifier}(q, M), & \text{sufficient;} \\ \text{LLM}_{CoT}(q), & \text{otherwise.} \end{cases} \quad (8)$$

Experiments

In this section, we empirically investigate the following five Research Questions (RQ): **RQ1**: Does our proposed DoM outperform baselines under KG incompleteness? **RQ2**: How robust is DoM when facing varying levels of KG incompleteness? **RQ3**: How does DoM perform when instantiated with different LLM backbones? **RQ4**: How does the core components contribute to the effectiveness of DoM? **RQ5**: How efficient is DoM in terms of token usage and runtime compared with baselines?

Experimental Setup

Dataset We evaluate DoM on the following datasets:

- A collection of datasets proposed by Xu et al. (Xu et al. 2024), consisting of 1,000 sampled questions from each of the CWQ and WebQSP. For each dataset, four degrees of incomplete KGs are generated by randomly removing 20%, 40%, 60%, or 80% crucial triples, which appear in the gold relation path. These variants, denoted as IKG-20%/40%/60%/80%, are used to assess model robustness under varying degrees of KG incompleteness.
- A new dataset, IKGWQ, constructed by updating entity knowledge with recent facts and generating questions grounded in information missing from the existing KG.

Method	IKGWQ	CWQ	WebQSP
w.o. Knowledge Graph (DeepSeek-v3)			
IO	28.5	50.1	68.8
CoT	51.0	54.3	66.3
w.t. Knowledge Graph / Fine-tuned[1]			
RoG	–	54.2	78.2
ChatKBQA	–	39.3	49.5
w.t. Knowledge Graph / Not-Training (DeepSeek-v3)			
ToG	46.5	52.0	70.2
PoG	27.5	55.9	78.0
GoG	49.5	60.4	78.1
DoM (Ours)	**84.5**	**62.0**	**81.7**

Table 1: Performance on IKGWQ and IKG-40% versions of CWQ and WebQSP. All results are reported as Hits@1 (%).

Baseline We evaluate DoM against a comprehensive set of baselines spanning three major categories in KGQA: (1) prompt-based LLM methods, including IO prompt (Brown et al. 2020) and CoT prompt (Wei et al. 2022); (2) fine-tuned LLM methods, such as ChatKBQA (Luo et al. 2024a) and RoG (Luo et al. 2024b); and (3) retrieval-augmented inference frameworks, including ToG (Sun et al. 2024), PoG (Tan et al. 2025), and GoG (Xu et al. 2024).

Evaluation Metrics Following previous works (Li et al. 2024; Luo et al. 2024b,a; Xu et al. 2024), exact match accuracy (Hits@1) is used as evaluation metric for all datasets.

Parameter Settings In our experiments, we adopt five LLMs as the backbones: DeepSeek-v3, Qwen-max, Qwen2.5-72B, GPT-3.5 and GPT-4o[2]. All LLMs are accessed via their official APIs. The maximum token length is set to 512 for each call. The temperature is set to 0.4 during the information retrieval and exploration stages, and reduced to 0 during final answer generation to ensure deterministic outputs. The maximum KG search depth per sub-question (W) is set to 3, and the maximum number of sub-question iterations (I) is set to 6. For RAG retrieval, chunk size is 500 tokens, and the retrieval top-k is set to 3.

Main Results (RQ1)

Table 1 reports Hits@1 scores of DoM and various baselines across multiple IKGQA datasets. As shown, DoM consistently outperforms all baselines.

On IKGWQ, DoM outperforms all evaluated baselines, demonstrating the effectiveness of its multi-source integration. The core challenge of IKGWQ lies in its faithful simulation of real-world KG incompleteness, characterized by irregularity and unpredictability. The unpredictability of miss-

[1]Fine-tuned LLM backbones: RoG (LLaMA2-Chat-7B); ChatKBQA (Llama-2-7B).

[2]Model versions: DeepSeek-v3 (default release); Qwen-max (qwen-max-2025-01-25); Qwen2.5-72B (qwen2.5-72b-instruct); GPT-3.5 (gpt-3.5-turbo-0613); GPT-4o (gpt-4o-2024-08-06).

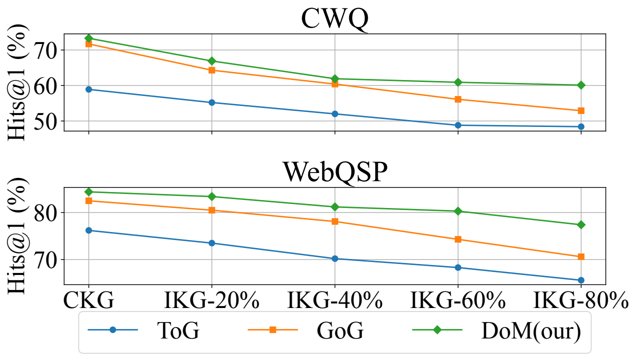

Figure 4: Performance on CWQ and WebQSP under varying KG incompleteness. CKG denotes a complete KG.

ing knowledge fundamentally limits the effectiveness of existing KGQA methods. In IKGWQ, gold relation paths are often incomplete in the KG due to missing crucial relations or entities, posing significant challenges for inference. Even GoG, specifically designed for IKGQA, suffers significant performance degradation under these conditions. Due to missing facts in the KG, the retrieved context may lack essential information. This can mislead the LLM into generating incorrect answers, occasionally underperforming simpler prompting strategies such as CoT. In contrast, DoM introduces external unstructured information and employs a debate mechanism to integrate it with structured KG evidence. Benefiting from this multi-source integration, DoM surpasses the strongest baseline by 70.7% on IKGWQ.

Beyond IKGWQ, DoM also achieves state-of-the-art performance on existing IKGQA datasets. While RoG outperforms several retrieval-based methods (e.g., GoG with DeepSeek-v3) on WebQSP, this primarily highlights that the performance of LLM-based inference methods depends heavily on the strength of the underlying backbone. In contrast, DoM consistently outperforms these baselines with the same backbone, demonstrating greater robustness and reasoning capability in IKG settings.

Performance Under Varying KG Incompleteness (RQ2)

To evaluate the robustness of DoM under different degrees of KG incompleteness, we conduct experiments on CWQ and WebQSP using DeepSeek-v3 as the backbone. The results, shown in Figure 4, demonstrate that DoM maintains consistent superiority across all degrees of incompleteness.

While all methods suffer performance degradation as more triples are removed, DoM exhibits notably slower decline, indicating enhanced resilience to missing knowledge. Even under extreme conditions (e.g., 80% of critical triples removed), DoM still outperforms GoG by a large margin (with a 13.6% relative improvement on CWQ IKG-80%), highlighting its robustness to severe knowledge sparsity.

The improved stability of DoM stems from its ability to integrate heterogeneous evidence sources through structured agent interaction. In particular, the KG-based and RAG-based agents independently retrieve and infer from different modalities, while the Judge Agent coordinates their outputs. This architecture enables DoM to recover from incomplete retrieval and supports reliable multi-hop inference even under sparse KG conditions.

Backbone	**IKGWQ**		
	IKG		NKG
DeepSeek-v3	**84.5**		51.0
Qwen-Max	78.0		**52.5**
Qwen2.5-72B	69.0		35.5
	CWQ		
	CKG	IKG-40%	NKG
DeepSeek-v3	**73.3**	**62.0**	**54.3**
Qwen-Max	67.2	59.7	51.2
Qwen2.5-72B	62.4	54.9	47.7

Table 2: Performance of DoM with different backbone models on IKGWQ and CWQ. CKG denotes a complete KG, and NKG corresponds to CoT reasoning without KG access.

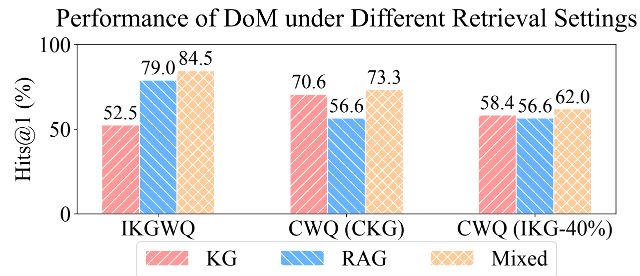

Figure 5: Performance of DoM with different retrieval agents on IKGWQ and CWQ. KG-retriever and RAG-retriever involve only the respective retrieval agent, with the Judge Agent reduced to a simple planner. Mixed-retriever activates all agents for full collaboration.

Performance with Different Backbones (RQ3)

We evaluate how DoM performs with different LLM backbones to assess its adaptability to various models. As shown in Table 2, DeepSeek-v3, which has the largest parameter size and strongest empirical performance among the evaluated models, achieves the best results on both IKG datasets. The performance gap between Qwen-Max and Qwen2.5-72B further shows that DoM benefits significantly from stronger backbones. This indicates that DoM can effectively scale with the capabilities of the underlying LLM.

Ablation Study (RQ4)

To assess the contribution of different retrieval agents in DoM, we conduct an ablation study using DeepSeek-v3 as the backbone, as shown in Figure 5. On the IKGWQ dataset, the KG Agent alone exhibits limited effectiveness, yielding performance comparable to that of LLM-based baselines reported in Table 1. This can be attributed to the irregular and unpredictable missing patterns in IKGWQ, which hinder the

	ToG	PoG	GoG	DoM
token / k	613	693	645	712
time / min	45	232	52	61

Table 3: Inference cost in terms of token usage and runtime across different methods.

KG Agent from consistently retrieving the necessary facts for accurate inference. In contrast, the KG Agent performs better on the CWQ dataset, where incompleteness is introduced in a controlled and predictable manner, making it easier to retrieve relevant facts. While the single-agent variants (KG-only and RAG-only) achieve reasonable performance individually, integrating their outputs via the Judge Agent yields significantly better results. These results underscore the complementary nature of structured and unstructured retrieval and validate the effectiveness of our modular design. It is notable that the KG-only variant of DoM outperforms the GoG baseline on IKGWQ. This can be attributed to the inherent nature of IKGWQ, where the missing information is more complex and unpredictable compared to the variants of CWQ and WebQSP. GoG relies heavily on retrieving relevant triples to complete the knowledge graph when missing facts need to be inferred. However, in IKGWQ, there often exists a significant gap between the retrieved triples and the missing facts required to answer the question, making it difficult for GoG to complete the inference process. In contrast, DoM's KG Agent can resort to LLM-based CoT inference when retrieval fails, leveraging LLM's internal knowledge to answer sub-questions. This adaptability allows DoM to handle the unpredictable incompleteness in IKGWQ more robustly, which explains its superior performance compared to GoG. This ablation confirms the necessity of both retrieval agents and the central role of the Judge Agent in coordinating complementary sources.

Computational Cost Analysis (RQ5)

To evaluate the computational overhead introduced by DoM, we analyze the inference cost on 100 samples from the CWQ IKG-40% setting. As shown in Table 3, DoM introduces a moderate increase in token usage and runtime compared with GoG, primarily due to the additional retrieval and debate steps involving external evidence. Nevertheless, this overhead remains substantially lower than that of PoG, while yielding significantly better performance. Overall, although the integration of external textual knowledge inevitably adds some cost, the increase is modest and well compensated by the notable improvements in accuracy and robustness. These results demonstrate that DoM achieves an effective balance between computational efficiency and performance gains.

Related Work

LLM-based KGQA

To enhance faithfulness, recent advances have explored how LLMs can be combined with KGs in question answering (Liu et al. 2020; Huang et al. 2024). These methods generally follow two main approaches: knowledge-internalization, where KGs are embedded into LLMs via fine-tuning to improve factual reasoning (Li et al. 2023), and knowledge-interaction, where LLMs query and perform inference on over external KGs (Sun et al. 2024).

Despite their strong performance, these methods often rely on complete KGs, which are difficult to realize due to the high cost of constructing and maintaining KGs (Hur, Janjua, and Ahmed 2021). This motivates growing interest in IKGQA (Min et al. 2013; Pflueger, Tena Cucala, and Kostylev 2022). Existing IKGQA approaches can be broadly categorized into two classes. The first class focuses on KG-internal completion by predicting missing links based on relational patterns among existing triples (Saxena, Kochsiek, and Gemulla 2022; Zan et al. 2022; Guo et al. 2023). However, such methods exhibit limited effectiveness when dealing with newly emerged or out-of-KG knowledge. The second class addresses KG incompleteness by incorporating external information beyond the KG itself, e.g., retrieving unstructured textual corpora to provide supplementary evidence or construct question-specific subgraphs (Sun, Bedrax-Weiss, and Cohen 2019; Lv et al. 2020), or utilizing LLMs as auxiliary knowledge sources (Xu et al. 2024).

Multi-Agent Debate

Multi-Agent Debate (MAD) enhances the reliability and diversity of LLM outputs by coordinating multiple agents under a judge's supervision, outperforming direct prompting on complex tasks (Liang et al. 2024; Chan et al. 2024; Qian et al. 2024). Existing MAD systems typically follow two paradigms: adversarial debate, where agents present conflicting viewpoints to promote critical reasoning (Liang et al. 2024; Liu et al. 2024); and collaborative planning, where specialized agents cooperate to solve complex problems (Du et al. 2023; Haase and Pokutta 2025).

Motivated by MAD's success, recent studies apply it to question answering (Mao, Yang, and Fu 2025; Ma et al. 2025; Hu et al. 2024). For example, DoG employs a MAD framework where agents reason over KG subgraphs for multi-hop questions (Ma et al. 2025). However, these works focus on inference and generation quality, with limited attention to addressing KG incompleteness.

Conclusion

To enable more effective inference under IKG scenarios, we proposed DoM, a MAD framework for IKGQA. By coordinating structured and unstructured evidence through agent collaboration, DoM effectively leverages their complementary strengths. Additionally, we constructed the IKGWQ dataset by revisiting samples from CWQ and WebQSP. We rebuild the corresponding QA pairs using up-to-date knowledge retrieved from reliable sources. This provides a more realistic benchmark for evaluating IKGQA systems. Extensive experiments demonstrate that DoM consistently outperforms strong baselines across multiple settings. We released both the IKGWQ dataset and our code to support future research.[3]

[3]https://github.com/liujilong0116/DoM

Acknowledgments

This work was supported by the National Key Research and Development Program under Grant 2023YFC2506800, and by the the National Natural Science Foundation of China under Grant No. 62406096.

References

Bollacker, K.; Evans, C.; Paritosh, P.; Sturge, T.; and Taylor, J. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, 1247–1250.

Brown, T.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J. D.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33: 1877–1901.

Chan, C.-M.; Chen, W.; Su, Y.; Yu, J.; Xue, W.; Zhang, S.; Fu, J.; and Liu, Z. 2024. Chateval: Towards better llm-based evaluators through multi-agent debate. In *International Conference on Representation Learning*, volume 2024.

Chen, L.; Tong, P.; Jin, Z.; Sun, Y.; Ye, J.; and Xiong, H. 2024. Plan-on-graph: Self-correcting adaptive planning of large language model on knowledge graphs. *Advances in Neural Information Processing Systems*, 37: 37665–37691.

Du, Y.; Li, S.; Torralba, A.; Tenenbaum, J. B.; and Mordatch, I. 2023. Improving factuality and reasoning in language models through multiagent debate. In *Forty-first International Conference on Machine Learning*.

Guo, Q.; Wang, X.; Zhu, Z.; Liu, P.; and Xu, L. 2023. A knowledge inference model for question answering on an incomplete knowledge graph. *Applied Intelligence*, 53(7): 7634–7646.

Haase, J.; and Pokutta, S. 2025. Beyond Static Responses: Multi-Agent LLM Systems as a New Paradigm for Social Science Research. *arXiv preprint arXiv:2506.01839*.

Hu, Z.; Yang, P.; Li, B.; and Wang, Z. 2024. Multi-agents based on large language models for knowledge-based visual question answering. *arXiv preprint arXiv:2412.18351*.

Huang, R.; Wei, W.; Qu, X.; Xie, W.; Mao, X.; and Chen, D. 2024. Joint Multi-Facts Reasoning Network For Complex Temporal Question Answering Over Knowledge Graph. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 10331–10335. IEEE.

Hur, A.; Janjua, N.; and Ahmed, M. 2021. A survey on state-of-the-art techniques for knowledge graphs construction and challenges ahead. In *2021 IEEE Fourth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE)*, 99–103. IEEE.

Li, W.; Wei, W.; Qu, X.; Mao, X.-L.; Yuan, Y.; Xie, W.; and Chen, D. 2023. TREA: Tree-Structure Reasoning Schema for Conversational Recommendation. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Li, X.; Zhao, R.; Chia, Y. K.; Ding, B.; Joty, S.; Poria, S.; and Bing, L. 2024. Chain-of-Knowledge: Grounding Large Language Models via Dynamic Knowledge Adapting over Heterogeneous Sources. In *ICLR*.

Liang, T.; He, Z.; Jiao, W.; Wang, X.; Wang, Y.; Wang, R.; Yang, Y.; Shi, S.; and Tu, Z. 2024. Encouraging Divergent Thinking in Large Language Models through Multi-Agent Debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 17889–17904.

Liu, D.; Qu, X.; Dong, J.; and Zhou, P. 2020. Reasoning step-by-step: Temporal sentence localization in videos via deep rectification-modulation network. In *Proceedings of the 28th International Conference on Computational Linguistics*, 1841–1851.

Liu, T.; Wang, X.; Huang, W.; Xu, W.; Zeng, Y.; Jiang, L.; Yang, H.; and Li, J. 2024. Groupdebate: Enhancing the efficiency of multi-agent debate using group discussion. *arXiv preprint arXiv:2409.14051*.

Luo, H.; Haihong, E.; Tang, Z.; Peng, S.; Guo, Y.; Zhang, W.; Ma, C.; Dong, G.; Song, M.; Lin, W.; et al. 2024a. ChatKBQA: A Generate-then-Retrieve Framework for Knowledge Base Question Answering with Fine-tuned Large Language Models. In *ACL (Findings)*.

Luo, L.; Li, Y.; Haffari, G.; and Pan, S. 2024b. Reasoning on Graphs: Faithful and Interpretable Large language Model Reasoning. In *ICLR 2024: The Twelfth International Conference on Learning Representations*. ICLR.

Lv, S.; Guo, D.; Xu, J.; Tang, D.; Duan, N.; Gong, M.; Shou, L.; Jiang, D.; Cao, G.; and Hu, S. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 8449–8456.

Ma, J.; Gao, Z.; Chai, Q.; Sun, W.; Wang, P.; Pei, H.; Tao, J.; Song, L.; Liu, J.; Zhang, C.; et al. 2025. Debate on graph: a flexible and reliable reasoning framework for large language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 24768–24776.

Mao, T.; Yang, S.; and Fu, B. 2025. A Multi-Agent Framework for Multi-Source Manufacturing Knowledge Integration and Question Answering. In *Companion Proceedings of the ACM on Web Conference 2025*, 1687–1695.

Min, B.; Grishman, R.; Wan, L.; Wang, C.; and Gondek, D. 2013. Distant supervision for relation extraction with an incomplete knowledge base. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 777–782.

Pflueger, M.; Tena Cucala, D. J.; and Kostylev, E. V. 2022. GNNQ: A neuro-symbolic approach to query answering over incomplete knowledge graphs. In *International Semantic Web Conference*, 481–497. Springer.

Qian, C.; Xie, Z.; Wang, Y.; Liu, W.; Zhu, K.; Xia, H.; Dang, Y.; Du, Z.; Chen, W.; Yang, C.; et al. 2024. Scaling Large Language Model-based Multi-Agent Collaboration. In *The Thirteenth International Conference on Learning Representations*.

Ren, H.; Hu, W.; and Leskovec, J. 2020. Query2box: Reasoning Over Knowledge Graphs In Vector Space Using Box

Embeddings. In *International Conference on Learning Representations (ICLR)*.

Saxena, A.; Kochsiek, A.; and Gemulla, R. 2022. Sequence-to-Sequence Knowledge Graph Completion and Question Answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2814–2828.

Sun, H.; Bedrax-Weiss, T.; and Cohen, W. 2019. PullNet: Open Domain Question Answering with Iterative Retrieval on Knowledge Bases and Text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 2380–2390.

Sun, J.; Xu, C.; Tang, L.; Wang, S.; Lin, C.; Gong, Y.; Ni, L. M.; Shum, H.-Y.; and Guo, J. 2024. Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph. In *ICLR 2024: The Twelfth International Conference on Learning Representations*. ICLR.

Tan, X.; Wang, X.; Liu, Q.; Xu, X.; Yuan, X.; and Zhang, W. 2025. Paths-over-graph: Knowledge graph empowered large language model reasoning. In *Proceedings of the ACM on Web Conference 2025*, 3505–3522.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Xia, F.; Chi, E.; Le, Q. V.; Zhou, D.; et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35: 24824–24837.

Xu, Y.; He, S.; Chen, J.; Wang, Z.; Song, Y.; Tong, H.; Liu, G.; Zhao, J.; and Liu, K. 2024. Generate-on-Graph: Treat LLM as both Agent and KG for Incomplete Knowledge Graph Question Answering. In *2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024*, 18410–18430. Association for Computational Linguistics (ACL).

Zan, D.; Wang, S.; Zhang, H.; Zhou, K.; Wu, W.; Zhao, W. X.; Wu, B.; Guan, B.; and Wang, Y. 2022. Complex question answering over incomplete knowledge graph as n-ary link prediction. In *2022 International Joint Conference on Neural Networks (IJCNN)*, 1–8. IEEE.