Reflection on Knowledge Graph for Large Language Models Reasoning

Anonymous ACL submission

Abstract

Recent research shows that supplementing Large Language Models (LLMs) with knowledge graphs can enhance their performance. However, existing methods often introduce noise in the retrieval and reasoning pipeline, hindering LLMs' ability to effectively integrate external knowledge for complex multi-hop question answering. To address this, we propose RefKG, a novel framework designed to enhance the reasoning capabilities of LLMs through reflective engagement with knowledge graphs. RefKG autonomously conduct retrieval and reflection on knowledge graphs. It consists of three modules: Query Decoupling, LLM-Driven Knowledge Graph Exploration, and Inference with Knowledge Reconstruction. We also introduce a multi-task tuning strategy that not only integrates external knowledge into LLMs but also trains them to leverage this knowledge for answering questions. This significantly improves their performance on knowledge-intensive tasks. Experiments on fact verification and knowledge graph question answering demonstrate RefKG's effectiveness.

1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities in solving various NLP tasks, such as machine translation (Zhang et al., 2023) and information extraction (Sainz et al., 2023). However, given the ever-evolving nature of real-world knowledge (Zhang et al., 2023), LLMs exhibit limitations in domain-specific expertise or in timely updating their knowledge bases. This shortfall often results in hallucinations within their responses, where the generated content deviates from factual accuracy (Huang et al., 2023).

To alleviate the issue of hallucinations in LLMs on knowledge-intensive tasks such as Knowledge Graph Question Answering (KGQA) (Gupta et al., 2018), a promising strategy involves augmenting LLMs with external knowledge sources (Tan et al., 2023), like knowledge graphs (KGs) (Luo et al., 2018; Hu et al., 2018). This approach retrieves relevant facts from knowledge bases to help LLMs generate more accurate responses. However, existing solutions still suffer from several shortcomings.

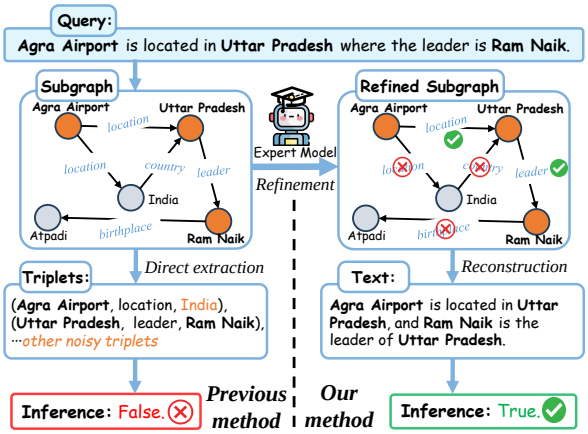

Figure 1: Comparison between previous method and our method. While conventional methods often introduce noisy knowledge during retrieval, our method employs an Expert Model for knowledge refinement, significantly reducing the acquisition of noisy information.

First, due to the scale and complexity of knowledge graphs, retrieval and reasoning processes often introduce irrelevant or noisy information, complicating the model's ability to answer complex queries (Lan et al., 2021), as illustrated in Figure 1. Second, recent investigations (Li et al., 2023b; Nie et al., 2023) have predominantly performed black-box testing on proprietary models such as ChatGPT. These studies often employ in-context learning techniques (Liu et al., 2022), where external knowledge is incorporated into the prompts to steer the model's response generation.

Although these training-free methods enable the integration of external knowledge, they do not enhance the interactive capabilities between LLMs and knowledge graphs, thereby limiting the potential of LLMs to efficiently acquire and deploy knowledge, especially when supervised signals are

available. Additionally, black-box models cannot be deployed privately, which significantly limits their flexibility and adaptability.

In this paper, we introduce RefKG, an innovative framework specifically crafted to enhance the reasoning capabilities of LLMs through reflective engagement with knowledge graphs. In particular, RefKG is structured as a three-step framework: 1) A *Query Decoupling Module* that decouples a complex query into multiple sub-queries that share a common knowledge background. 2) A *LLM-Driven Knowledge Graph Exploration Module* that iteratively and reflectively retrieves relevant evidence subgraphs from a knowledge base and refines the knowledge through an expert model. 3) An *Inference with Knowledge Reconstruction Module* that transforms structured knowledge into a natural language format that is more easily understood by the LLM, and integrates it with the question to derive the answer. Compared to approaches that directly use retrieved results in prompts (Kim et al., 2023a), our approach maximizes the reflection capabilities of LLMs (Asai et al., 2023) to critically assess and refine the evidence subgraph. Furthermore, we have formulated a knowledge-driven multi-task tuning strategy that provides RefKG with foundational expertise in knowledge-intensive reasoning. This is achieved by fine-tuning the model on a specially synthesized corpus, equipping it with the necessary skills for advanced reasoning tasks. Together, the three-step process enables our approach to autonomously retrieve, reflect, and utilize knowledge in solving knowledge-intensive tasks. In summary, our main contributions are three-fold:

- We propose RefKG, a novel framework crafted to enhance the reasoning capabilities of LLMs through reflective engagement with knowledge graphs. In particular, our approach simplifies complex queries through decomposition, enabling effective retrieval, reflection and reasoning within knowledge graphs.
- We develop an LLM-Generated corpus for knowledge-intensive multi-task tuning, equipping LLMs with initial expertise in knowledge-intensive reasoning, setting the stage for advanced task-specific learning.
- We extensively evaluate RefKG on fact verification and knowledge graph question answering tasks. The experimental results affirm that RefKG outperforms previous KG-augmented methods across various open-source LLMs.

2 Related Work

KG Retrieval-Augmented Methods. Knowledge graphs (KGs) organize relationships between entities in a structured manner, and leveraging KG retrieval to enhance large language models (LLMs) has proven effective in mitigating hallucination issues (Agrawal et al., 2023; Pan et al., 2023). Recent research in KG retrieval can be broadly classified into two categories: (1) *Semantic Parsing-Based Methods*: For example, SSKGQA (Li and Ji, 2022) generates query graphs based on questions to eliminate incorrect query structures, while RnG-KBQA (Ye et al., 2022) ranks and generates logical forms (LFs) from candidate queries. However, these methods require generating executable SPARQL statements and additional label information. (2) *Information Retrieval-Based Methods*: For instance, UniK-QA (Oguz et al., 2022) combines retrieved triplets with questions in a Seq2Seq framework to generate answers. However, it rely heavily on the accuracy of the retrieved subgraphs or triplets, lacking mechanisms to filter irrelevant results, which can lead to error accumulation. DiFaR (Baek et al., 2023a) improves retrieval accuracy by leveraging query-triplet similarity but struggles with complex multi-hop questions. In contrast, our approach enhances the KG retrieval process by utilizing LLMs' semantic capabilities to guide retrieval and applies robust quality control to the results.

LLMs Reasoning for KGQA. Recent work has focused on using LLMs for Knowledge Graph Question Answering (KGQA), with methods like KAPING (Baek et al., 2023b) and KGGPT (Kim et al., 2023a) prompting LLMs to generate answers by inserting retrieved triplets into predefined templates. However, these methods neglect the challenge that triplets from knowledge graphs are not always in natural language, complicating LLM inference. Additionally, they rely on black-box APIs, limiting model training and deployment. Retrieve-Rewrite-Answer (Wu et al., 2023) addresses this by fine-tuning open-source LLMs on KG-to-text corpora, converting triplets into more readable text. However, none of these methods filter the extracted triplets, potentially introducing irrelevant information and leading to incorrect results.

3 Methodology

As shown in Figure 2, our proposed RefKG is structured with three modules: Query Decoupling,

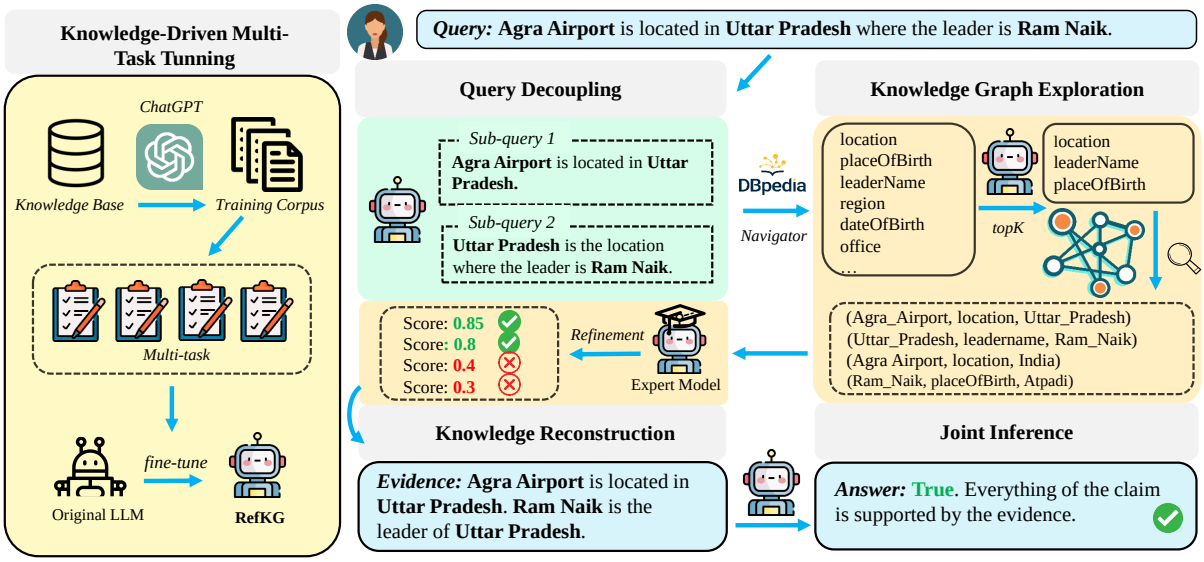

Figure 2: An overview of our proposed framework RefKG. The framework consists of three modules: *Query Decoupling*, *Evidence Subgraph Retrieval* and *inference with Knowledge Reconstruction*. We enhance the model's ability to utilize knowledge through Knowledge-Driven Multi-Task Tuning, enabling the decoupling, navigation, refinement, and reconstruction of knowledge.

LLM-Driven Knowledge Graph Exploration, Inference with Knowledge Reconstruction. Besides, we enhance the model's ability to utilize knowledge through Knowledge-Driven Multi-Task Tuning.

3.1 Query Decoupling

Inspired by the divide-and-conquer paradigm, we initially decouple a complex query into multiple sub-queries, each of which shares the contextual semantics but contains only a single-hop atomic query. In knowledge-intensive tasks, we assume entities contain the essential information necessary for the decomposition process. By anchoring entities, LLMs can capture the underlying mechanisms of knowledge-intensive problem decoupling.

Specifically, given a knowledge-intensive query q, a collection of relevant knowledge entities E , and a predefined decoupling template P, our goal is to predict the hop number H, derive a sequence of sub-queries $q_{sub} = [q_1, ..., q_H]$, and identify the corresponding entity subsets $E_{sub} = [e_1, ..., e_H]$ for each subquery. It can be formulated as:

$$\{q_i, e_i\}_{i=1}^{H} = \mathbf{LLM}(p'), \quad p' = \mathbf{P}(q, e), \tag{1}$$

3.2 LLM-Driven Knowledge Graph Exploration

As shown in Figure 3, the evidence subgraph retrieval consists of *Evidence Subgraph Retrieval* and *Knowledge Refinement*.

Evidence Subgraph Retrieval. Our approach leverages the LLM as a navigator, encouraging it to autonomously select the search trajectory on the related subgraph $\mathcal{G}_{\text{sub}}$, continuously advancing to form a chain of reasoning. Specifically, we divide the retrieval reasoning process into multiple iterations, ultimately forming a complete chain $\mathcal{P}_t$, formulated as:

$$\begin{aligned}\mathcal{P}_t = \{(e_1^{head}, r_1, e_1^{tail}) \xrightarrow{LLM} \dots \xrightarrow{LLM} \\ (e_T^{head}, r_T, e_T^{tail}), (e_t^{head}, r_t, e_t^{tail}) \in \mathcal{G}_{sub}\}\end{aligned} \tag{2}$$

For each iteration, the LLM conducts interpretable reasoning on the graph by targeting relationships as objectives for selecting paths. We formulate the relation selection task as an optimization problem, with the objective of maximizing the probability of extracting a set of relationships r from the knowledge graph $\mathcal{G}$ by generating an inference chain $\mathcal{P}_t$:

$$\begin{aligned}P_\theta(r|q, e, \mathcal{G}) = \sum_{p_{t-1} \in \mathcal{P}_{t-1}} P_\theta(r|p_{t-1}, q, e, \mathcal{G}) \\ \cdot P_\theta(p_{t-1}|q, e, \mathcal{G}),\end{aligned} \tag{3}$$

The new relation r are incorporated into the reasoning path to form new reasoning paths p_t, with N such paths together constituting a complete evidence subgraph $\mathcal{G}_{evi} = \{p_t^n\}_{n=1}^{N}$. To improve the stability and coverage of relation selection, our approach incorporates the Top-k most relevant relations into the reasoning chain, rather than relying on a single relation.

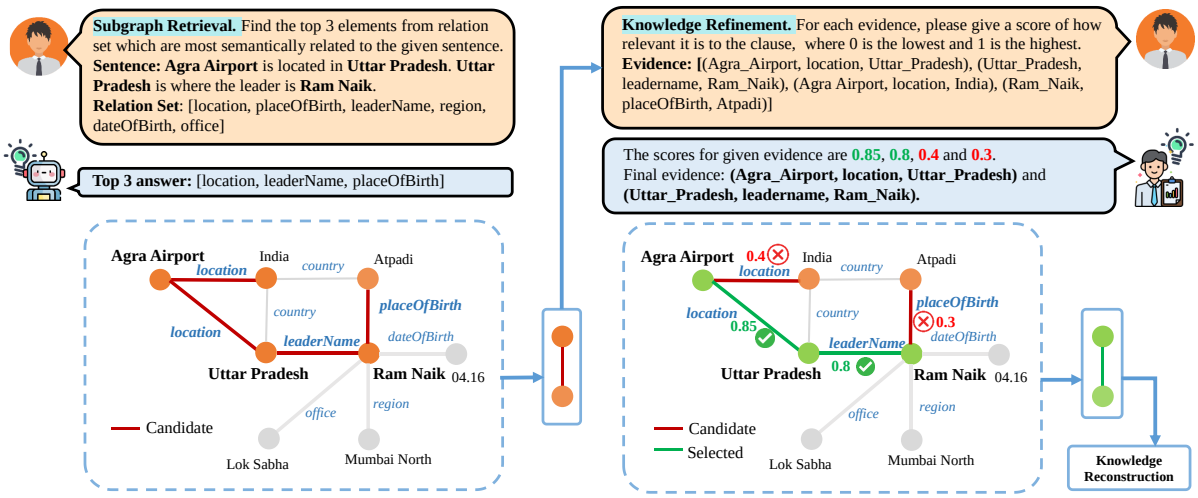

Figure 3: In the Evidence Subgraph Retrieval process, RefKG initiates from entities within the related subgraph to select the most probable relations, thereby constructing an inference pathway in triplets-form. In the Knowledge Refinement phase, RefKG uses a trained expert model to score and rerank the retrieved knowledge, filtering out noisy triplets.

Evidence Subgraph Refinement. To address the noisy data of the evidence subgraph retrieved from large knowledge graphs, we trained an expert model to refine and rerank the generated evidence subgraph, enhancing both the accuracy and effectiveness of the external knowledge.

Given a sub-query q_i and its corresponding evidence subgraph $\mathcal{G}_{sub,i} \in \mathcal{G}_{sub}$,we utilize an LLM to jointly encode the query and subgraph together, resulting in a hidden layer state h_i. Then we integrate a single Multi-Layer Perceptron (MLP) after an LLM for regression training, aimming to map the hidden layer state h_i to a corresponding score s_i (more details in Appendix A.6). The formula for this mapping is expressed as follows:

$$s_i = \mathbf{MLP}(h_i),\ h_i = \mathbf{LLM}(q_i, \mathcal{G}_{sub,i}) \quad (4)$$

We use the Mean Squared Error (MSE) loss as the objective function, formulated as:

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(s_i - \hat{s}_i)^2 \quad (5)$$

where s_i represents the actual scores, and $\hat{s}_i$ denotes the predicted scores by the model. Then, we rerank the obtained evidence triplets by score and set a threshold α to filter out triplet reasoning paths that are irrelevant to the question.

3.3 Inference with Knowledge Reconstruction

To improve the LLM's capacity to integrate external knowledge, we reconstruct the evidence subgraph into a natural language format.

For an evidence subgraph $\mathcal{G}_{evi}$ containing n triplets, We transform them into a textual prompt p' by a predefined template P, and then transform the input p' into a trained LLM to generate the textual evidence evi:

$$evi = \mathbf{LLM}(p'),\ p' = \mathbf{P}(\mathcal{G}_{evi}), \quad (6)$$
$$\mathcal{G}_{evi} = \{(e_n^{head}, r_n, e_n^{tail})\}_{n=1}^{N},$$

Then we perform reasoning in two types of knowledge-intensive tasks, question answering tasks and fact verification tasks. We unify them into a single probabilistic model, formulated as:

$$P_\theta(a|q,\mathcal{G}) = P_\theta(a|evi,q,\mathcal{G})P_\theta(evi|q,\mathcal{G}) \quad (7)$$

where a denote the answer, evi denote the evidence transformed from knowledge graph. And the details of prompts templates for each step of RefKG are thoroughly outlined in Appendix A.10.

3.4 Knowledge-Driven Multi-Task Tunning

3.4.1 Training Corpus

Corpus Generation. To address the limitations of existing corpora that do not fully meet our training needs, we have developed a multi-task approach for corpus generation. To create training data, we focus on three specific tasks: (1) Query Decoupling, (2) Evidence Subgraph Retrieval, and (3) Inference with Knowledge Reconstruction. For each task, we design a pre-defined template T and insert relevant feature elements x, forming a text

prompt p. This prompt p is then processed by ChatGPT[1] to generate the corresponding training data y. Further details are provided in Appendix A.3.

Quality Control. In light of the lack of explicit labels and the challenge of applying general metrics, we have developed specific evaluation methods for assessing the quality of the generated outcomes: (1) For *Query Decoupling*: We evaluate the decoupling quality based on the entity set E extracted from the original question and the entity sets $E_{div} = [e_{div,1}, ..., e_{div,H}]$ derived from the decomposed sub-queries. The criteria for considering the decoupling results as high-quality are as follows: (a) $E \neq \emptyset$. (b) $E = \bigcup_{i=1}^{H} e_{div,i}$. (c) If $|E_{\text{div}}| > 1$, then $\forall e_{div,i} \in E_{\text{div}}, e_{div,i} \subsetneq E$. If $|E_{\text{div}}| = 1$, then $E_{\text{div}} = E$. (2) For *Inference with Knowledge Reconstruction*: We perform a unified assessment of the two-step pipeline process. For the answers A generated through these two steps, we identify instances where the feedback from the generator corresponds with the factual ground truth as indicators of high-quality data.

3.4.2 Multi-Task Tunning

Previous research has demonstrated that multi-task learning is effective when tasks are diverse but related, particularly when they share a common knowledge background, despite requiring different skills (Ni et al., 2023). Based on this insight, we have designed tasks that are related in knowledge but involve distinct skill sets. In the training phase, we synergistically infuse both linguistic and entity knowledge into LLMs by focusing on the optimization of three tasks: *Query Decoupling*, *Evidence Subgraph Retrieval*, and *Inference with Knowledge Reconstruction*

The auto-regressive training objective focuses on training the LLM to predict subsequent tokens based on previous tokens. Specifically, for the prompt p_i of different tasks, the objective function for generating the target answer $z = [z_1, ..., z_T]$ is:

$$\mathcal{L}_{\text{i}}(\theta) = -\sum_{t=1}^{T} \log p_\theta(z_t | \boldsymbol{z}_{<t}, \boldsymbol{p}_i) \tag{8}$$

4 Experiments

4.1 Experimental Setup

Datasets. We evaluate RefKG on a fact-verification benchmark: FactKG (Kim et al., 2023b), and two KGQA benchmarks: MetaQA (Zhang et al., 2018) and WebQSP (Yih et al., 2016). FactKG and WebQSP are both highly challenging benchmarks, while MetaQA is relatively less difficult. Further dataset details are provided in Appendix A.2.

Baselines. For FactKG, we compare RefKG with two types of baselines: (1) *Claim Only*: These baselines utilize the claim as the input without any evidence retrieved from the knowledge graph, including classifiers trained on the training set such as BERT, BlueBERT, and popular LLMs. (2) *With Evidence*: These baselines incorporate both the claim and retrieved evidence as inputs. This group includes fully supervised models like GEAR (Zhou et al., 2019) and 12-shot model KG-GPT (Kim et al., 2023a). For MetaQA and WebQSP, we compare RefKG with four types of baselines: 1) *Embedding-based methods*. 2) *Retrieve-augmented methods*. 3) *Prompting-based LLMs methods*, and 4) *Fine-tuned LLMs methods*. The details of each baseline are described in Appendix A.4.

Implementation Details. We perform our experiments across a diverse range of LLMs, including Llama-2 7B (Touvron et al., 2023), Bloom 7B (Workshop et al., 2022), Baichuan-2 7B (Yang et al., 2023) and Internlm-2 7B (Team, 2023). For evidence subgraph retrieval, we configure the number of relations k to be either 2 or 5. For knowledge refinement, we establish a score threshold α of 0.6. See Appendix A.5 for more details.

4.2 Main Results

Results on FactKG. The results are shown in Table 1. We can make the following observations:

First, RefKG with Bloom-7B outperforms all baseline methods in terms of the overall accuracy, attaining a new state-of-the-art status on this benchmark. This success can be attributed to our framework's dual strategy of employing knowledge graphs as external resources and harnessing the innate reasoning powers of LLMs. By encouraging LLMs to engage deeply with and reflect on retrieved information, RefKG significantly enhance the performance.

Second, fine-tuned 7B-parameter LLMs exhibit much better performance in fact verification tasks than LLMs without fine-tuning. Notably, RefKG enhances the performance of Llama 2, Bloom, Internlm 2 and Baichuan 2 by 71.62%, 73.67%, 41.64% and 56.01%, respectively.

[1]ChatGPT is from https://openai.com/

Method	One-hop	Conjunction	Existence	Multi-hop	Negation	Overall
Claim Only						
BERT∗	69.64	63.31	61.84	70.06	63.62	65.20
BlueBERT∗	60.03	60.15	59.89	57.79	58.90	59.93
Flan-T5∗	62.17	69.66	55.29	60.67	55.02	62.70
Baichuan-2 7B	29.88	26.21	18.55	18.43	17.73	24.29
Llama-2 7B	13.17	2.58	20.40	10.08	24.35	9.64
Internlm-2 7B	39.98	40.54	28.71	48.00	34.55	40.40
Bloom 7B	3.24	16.61	2.16	13.80	7.69	10.37
With Evidence						
KG-GPT†[EMNLP23]	-	-	-	-	-	72.68
GEAR∗[ACL19]	83.23	77.68	**81.61**	68.84	79.41	77.65
RefKG (Ours)						
Baichuan-2 7B	81.14	83.75	80.83	73.52	77.63	80.30(+2.65)
Llama-2 7B	84.13	**88.46**	72.83	71.83	**83.64**	81.26(+3.61)
Internlm-2 7B	84.18	86.12	76.15	76.41	80.06	82.04(+4.39)
Bloom 7B	**85.65**	87.94	81.14	**77.81**	82.80	**84.04**(+6.39)

Table 1: Performance of different models on the FactKG benchmark. Performance marked with ∗ are sourced from (Kim et al., 2023b) and those marked with † are sourced from (Kim et al., 2023a). We applied our method, RefKG, to experiments on four open-source large language models (Baichuan-2, Llama-2, Internlm-2, Bloom), testing it against five types of questions (One-hop, Conjunction, Existence, Multi-hop, Negation). The green numbers indicate the improvement values compared to the GEAR method, **Bold** numbers represent the highest values, and underlined numbers represent the second-highest values.

Third, in the context of knowledge graph retrieval methods, RefKG outperforms KG-specific supervised models like GEAR and training-free approaches such as KG-GPT. This underscores the effectiveness of our approach, which involves fine-tuning LLMs with a rich set of instructions. Moreover, RefKG demonstrates commendable results across all five tasks, with the exception of the Existence category. This exception might stem from the limited entity information available, which poses challenges for effective query decoupling.

Results on WebQSP. As shown in Table 2, RefKG demonstrates competitive performance, achieving a Hits@1 score of 85.2% within fine-tuned LLMs methods. Moreover, unlike prompting-based LLMs methods that typically rely on carefully crafted prompts to guide black-box large models in generating answers, RefKG surpasses them by fine-tuning a 7B-parameter LLM.

Results on MetaQA. As shown in Table 3, RefKG reaches state-of-the-art performance on the Hop-1 test set, recording a 98.1% accuracy. This exceptional performance is attributed to the richer relational context available in Hop-1 compared to Hop-2 and Hop-3, suggesting that the strategic use of LLMs for relation selection minimizes errors at this juncture, thereby enhancing overall results. Additionally, RefKG achieves performances close to SOTA on the Hop-2 and Hop-3 test sets, underscoring its versatility and robust adaptability across a variety of tasks. This demonstrates RefKG's consistent and reliable performance across both single-hop and multi-hop question answering tasks.

Method	Hits@1
Embedding	
EmbedKGQA(Saxena et al., 2020)[ACL20]	66.6
NSM(He et al., 2021)[WSDM21]	68.7
TransferNet(Shi et al., 2021)[EMNLP21]	71.4
Retrieval	
GraftNet(Sun et al., 2018)[EMNLP18]	66.4
PullNet(Sun et al., 2019)[EMNLP19]	68.1
SR+NSM(Zhang et al., 2022)[ACL22]	68.9
LLM (Prompting)	
KAPING(Baek et al., 2023b)[NLRSE23]	73.9
KB-BINDER(Li et al., 2023b)[ACL23]	74.4
ChatGPT+ToG(Sun et al., 2024)[ICLR24]	76.2
FRAG(Gao et al., 2025)	76.7
GPT4+ToG(Sun et al., 2024)[ICLR24]	82.6
LLM (Fine-tuned)	
InstructGraph(Yu et al., 2022)[ACL24]	73.3
UniKGQA(Jiang et al., 2022)[ICLR23]	77.2
Retrieve-Rewrite(Wu et al., 2023)[IJCKG23]	79.4
DECAF(Yu et al., 2022)[ICLR23]	82.1
RefKG (Ours)	**85.2**

Table 2: The performance of the models on WebQSP. The best results are in bold.

4.3 Ablation Study

Is text form better than triplet form? As illustrated in Table 4, our experiments suggest that

Methods	1-hop	2-hop	3-hop	Avg.
Embedding				
KVMemNN(Xu et al., 2019)[NAACL19]	96.2	82.7	48.9	75.9
EmbedKGQA(Saxena et al., 2020)[ACL20]	97.5	98.8	94.8	97.0
NSM(He et al., 2021)[WSDM21]	97.1	**99.9**	98.9	98.6
Retrieval				
GraftNet(Sun et al., 2018)[EMNLP18]	97.0	94.8	77.7	89.9
PullNet(Sun et al., 2019)[EMNLP19]	97.0	**99.9**	91.4	96.1
LLM (Prompting)				
ChatGPT	60.0	23.0	38.7	40.6
KG-GPT(Kim et al., 2023a)[EMNLP23]	96.3	94.4	94.0	94.9
StructGPT(Jiang et al., 2023)[EMNLP23]	97.1	97.3	87.0	93.8
KB-BINDER[ACL23]	93.5	99.6	96.4	96.5
LLM (Fine-tuned)				
UniKGQA(Jiang et al., 2022)[ICLR22]	97.5	99.0	**99.1**	98.5
Retrieve-Rewrite(Wu et al., 2023)[IJCKG23]	-	97.7	-	97.7
RefKG (Ours)	**98.1**	99.4	99.0	**98.8**

Table 3: The performance of the models on MetaQA (Hits@1). The best results are in bold.

Methods	Accuracy	Rate (%)
RefKG (full)	**81.26**	0.00
-triplet only	61.15	-24.74
-w/o Knowledge Refinement	78.55	-3.33
-w/o Knowledge Reconstruction	68.99	-15.10
-w/o JI tunning	50.62	-37.71
RefKG (Lora-ft)	72.45	-10.84

Table 4: Ablation study on the FactKG using llama2-7b. "Rate" quantifies the reduction in accuracy. JI tunning denotes the Joint Inference tunning.

Model	Base Model	RefKG (Ours)	Difference
Llama-2 7B	34.12	81.26	-47.14
Bloom 7B	37.65	84.04	-46.39
Internlm-2 7B	39.41	82.04	-42.63
Baichuan-2 7B	31.73	80.30	-48.57
Average	35.73	81.84	-46.11

Table 5: Comparison of Multi-task Tuning and untrained base model.

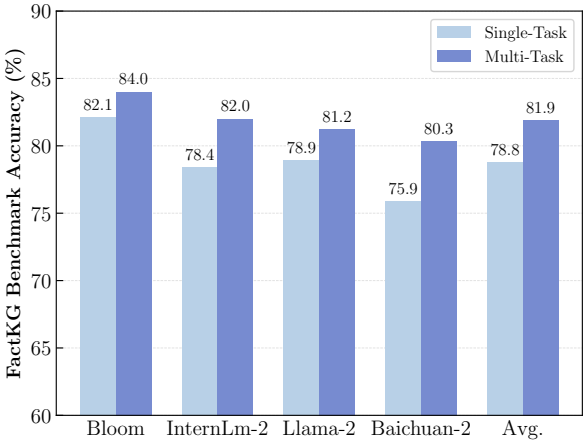

Figure 4: Comparison of Multi-task Tuning and Single-task Tuning.

replacing natural language text with triplets results in a performance decline of about 20%. We hypothesize that triplets may lack crucial semantic details, hindering the model's ability to process the information effectively. In contrast, providing evidence in natural language aligns better with the LLM's pre-training corpus, enhancing the model's ability to utilize the information more efficiently.

How does Knowledge Refinement enhance inference performance? As illustrated in Table 4, removing Knowledge Refinement from the RefKG framework results in a performance drop of approximately 2.71%. This highlights the importance of reflection in the model's analytical process. Through reflection, the model can independently identify and discard incorrect relations and evidence, leading to more accurate inferences.

What role does multi-task tuning play in RefKG? We first conducted comparison experiments using an untrained base model for the entire process, with results presented in Table 5. The findings reveal a 46.11% average performance drop, indicating that untrained models struggle with our multi-task framework and lack the capacity to handle complex knowledge-based tasks.

Then we conducted an experiment to evaluate the performance gap between multi-task fine-tuning and single-task fine-tuning. Specifically, we train the LLM on multiple independent single tasks and then combine the trained LLMs into a unified system for inference. We refer to this approach as single-task tuning. As shown in Figure 4, single-task tuning weakens the model's overall capabilities compared to multi-task tuning, leading to a decline in task performance, with an average accuracy drop of 3.21%.

We attribute the advantages of multi-task tuning to three key factors: (1)Multi-task tuning enables the model to share hidden layers across different tasks, thereby facilitating the sharing of learned features and representations. (2)Training on multiple tasks simultaneously mitigates overfitting, enhancing the model's ability to generalize. (3)Multi-task tuning optimizes data utilization and lowers computational costs.

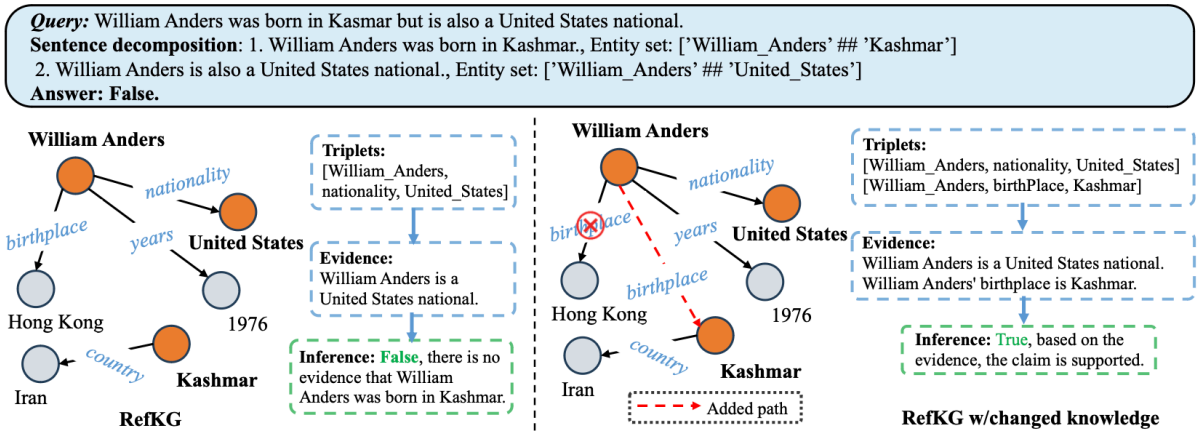

Figure 5: A case study on FactKG. The left figure illustrates the process of RefKG handling a claim, while the right figure depicts the modification made to the knowledge graph, resulting in the change of RefKG's response.

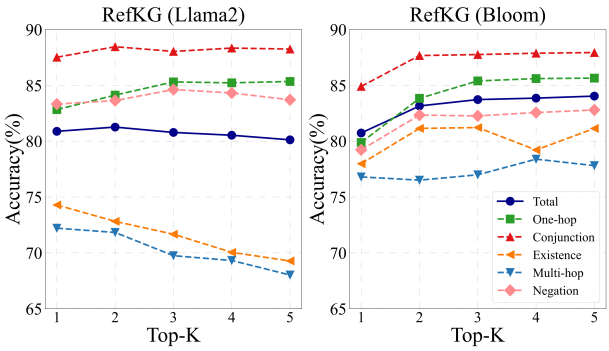

Figure 6: Impact of varying the number of Top-K retrieval with Llama-2 and Bloom on FactKG.

4.4 Further Analysis

Impact of numbers of Top-K Retrieval. As shown in Figure 6, we investigated the impact of Top-K values ranging from 1 to 5. The Bloom model consistently improves as the Top-K value increases, while the Llama model shows a decline in performance with higher Top-K settings. This suggests that an increase in the number of paths selected, and consequently, more evidence being generated, may overwhelm the Llama model, complicating its ability to distill crucial information from an extensive pool of evidence. Interestingly, when the Top-K value is set to 1, where only the most probable relation is chosen, RefKG still performs well. This indicates that the LLM's ability to select the most relevant relation from a limited set is often sufficient for accurate results.

Qualitative Analysis We conduct a case study as presented in Figure 5. Based on the given statement, our method RefKG performs sentence decomposition to identify triplets and transform them into evidence. Since no relevant fact is found in the knowledge graph for the statement *"William Anders was born in Kashmar"*, our model outputs *"False"*. This underscores RefKG's capability to precisely detect the absence of supporting evidence for incorrect statements and to consequently deliver an accurate verdict.

Furthermore, we explore whether RefKG can adapt to newly updated knowledge. By manually adding a new path into the original KG, our model adeptly identifies and processes the triplets into evidence, resulting in a diametrically opposed conclusion. This case demonstrates the model's ability to seamlessly adjust to updated factual knowledge, negating the necessity for further training or adjustments. This flexibility highlights RefKG's potential for maintaining relevance and accuracy in the face of evolving knowledge bases.

5 Conclusion

In this paper, we proposed the RefKG framework, which engages with knowledge graphs in a reflective manner to identify the most likely relational paths and evidence, using this curated evidence to derive answers. To Infuse the LLM with the abilities to decouple, navigate, refine, reconstruct, and reason over knowledge, we developed a knowledge-driven multi-task tuning approach and built a corresponding training corpus. The experimental results prove its effectiveness on fact verification and knowledge graph question answering. Our method can be deployed on any open-source LLM, and the experimental results indicate that it achieves excellent performance in fact verification and knowledge graph question answering.

Limitations

In this section, we faithfully discuss the limitations of our approach and potential avenues for future research.

Cumulative error effect. Although our framework can handle some complex multi-hop or negation questions, it involves multiple subtasks. The workflow of the pipeline generates a cumulative error effect. For example, if the model misidentifies entities in the first step of sentence decomposition, subsequent answers obtained will inevitably be incorrect. Therefore, future work could focus on reducing error rates by introducing efficient and accurate retrieval methods or instruction fine-tuning methods.

Larger model sizes. Limited by computational resources, we only applied RefKG to the 7B LLM and conducted full-parameter fine-tuning of the model under this configuration without testing larger models. We hope to conduct experiments on models with larger parameter sizes such as OPT (175B) in the future.

Ethical Considerations

Our approach RefKG has been validated on publicly available datasets FactKG and MetaQA. However, it is unclear how RefKG performs on other specific datasets or domains. Therefore, using RefKG in some highly sensitive and high-risk datasets or domains may result in the generation of offensive information or other unexpected consequences. We recommend practitioners to conduct thorough testing and inspection before applying our method to real-world scenarios.

References

Garima Agrawal, Tharindu Kumarage, Zeyad Alghami, and Huan Liu. 2023. Can Knowledge Graphs Reduce Hallucinations in LLMs? : A Survey. *arXiv preprint arXiv:2311.07914*.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection. *arXiv preprint arXiv:2310.11511*.

Jinheon Baek, Alham Fikri Aji, Jens Lehmann, and Sung Ju Hwang. 2023a. Direct fact retrieval from knowledge graphs without entity linking. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 10038–10055.

Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023b. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. In *Proceedings of the Workshop on Natural Language Reasoning and Structured Explanations (NLRSE)*, pages 78–106.

Zengyi Gao, Yukun Cao, Hairu Wang, Ao Ke, Yuan Feng, Xike Xie, and S Kevin Zhou. 2025. Frag: A flexible modular framework for retrieval-augmented generation based on knowledge graphs. *arXiv preprint arXiv:2501.09957*.

Vishal Gupta, Manoj Chinnakotla, and Manish Shrivastava. 2018. Retrieve and re-rank: A simple and effective IR approach to simple question answering over knowledge graphs. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*.

Gaole He, Yunshi Lan, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Improving Multi-hop Knowledge Base Question Answering by Learning Intermediate Supervision Signals. *arXiv preprint arXiv:2101.03737*.

Sen Hu, Lei Zou, and Xinbo Zhang. 2018. A state-transition framework to answer complex questions over knowledge base. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2023. A Survey on Hallucination in Large Language Models: Principles, Taxonomy, Challenges, and Open Questions. *arXiv preprint arXiv:2311.0523*.

Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Xin Zhao, and Ji-Rong Wen. 2023. StructGPT: A general framework for large language model to reason over structured data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing(EMNLP)*.

Jinhao Jiang, Kun Zhou, Wayne Xin Zhao, and Ji-Rong Wen. 2022. UniKGQA: Unified Retrieval and Reasoning for Solving Multi-hop Question Answering Over Knowledge Graph. *arXiv preprint arXiv:2212.00959*.

Jiho Kim, Yeonsu Kwon, Yohan Jo, and Edward Choi. 2023a. KG-GPT: A general framework for reasoning on knowledge graphs using large language models. In *Findings of the Association for Computational Linguistics: EMNLP*, pages 9410–9421.

Jiho Kim, Sungjin Park, Yeonsu Kwon, Yohan Jo, James Thorne, and Edward Choi. 2023b. FactKG: Fact verification via reasoning on knowledge graphs. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 16190–16206.

Yunshi Lan, Gaole He, Jinhao Jiang, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Complex Knowledge Base Question Answering: A Survey. *arXiv preprint arXiv:2108.06688*.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick van Kleef, S. Auer, and Christian Bizer. 2015. Dbpedia - a large-scale, multilingual knowledge base extracted from wikipedia. *Semantic Web*, pages 167–195.

Junyi Li, Xiaoxue Cheng, Xin Zhao, Jian-Yun Nie, and Ji-Rong Wen. 2023a. HaluEval: A large-scale hallucination evaluation benchmark for large language models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Mingchen Li and Shihao Ji. 2022. Semantic structure based query graph prediction for question answering over knowledge graph. In *Proceedings of the International Conference on Computational Linguistics (COLING)*, pages 1569–1579.

Tianle Li, Xueguang Ma, Alex Zhuang, Yu Gu, Yu Su, and Wenhu Chen. 2023b. Few-shot in-context learning on knowledge base question answering. In *Proceedings of Annual Meeting of the Association for Computational Linguistics (ACL)*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. What makes good in-context examples for GPT-3? In *Proceedings of Deep Learning Inside Out (DeeLIO 2022)*.

Kangqi Luo, Fengli Lin, Xusheng Luo, and Kenny Zhu. 2018. Knowledge base question answering via encoding of complex query graphs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Jingwei Ni, Zhijing Jin, Qian Wang, Mrinmaya Sachan, and Markus Leippold. 2023. When does aggregating multiple skills with multi-task learning work? a case study in financial nlp. *arXiv preprint arXiv:2305.14007*.

Zhijie Nie, Richong Zhang, Zhongyuan Wang, and Xudong Liu. 2023. Code-Style In-Context Learning for Knowledge-Based Question Answering. *arXiv preprint arXiv:2309.04695*.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. UniK-QA: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of the Association for Computational Linguistics: NAACL*, pages 1535–1546.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2023. Unifying Large Language Models and Knowledge Graphs: A Roadmap. *arXiv preprint arXiv:2306.08302*.

Oscar Sainz, Iker García-Ferrero, Rodrigo Agerri, Oier Lopez de Lacalle, German Rigau, and Eneko Agirre. 2023. GoLLIE: Annotation Guidelines improve Zero-Shot Information-Extraction. *arXiv preprint arXiv:2310.03668*.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 4498–4507.

Jiaxin Shi, Shulin Cao, Lei Hou, Juanzi Li, and Hanwang Zhang. 2021. TransferNet: An effective and transparent framework for multi-hop question answering over relation graph. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Haitian Sun, Tania Bedrax-Weiss, and William Cohen. 2019. PullNet: Open domain question answering with iterative retrieval on knowledge bases and text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2380–2390.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4231–4242.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Heung-Yeung Shum, and Jian Guo. 2024. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph.

Md Tahmid Rahman Laskar, Mizanur Rahman, Israt Jahan, Enamul Hoque, and Jimmy Huang. 2023. CQSumDP: A ChatGPT-Annotated Resource for Query-Focused Abstractive Summarization Based on Debatepedia. *arXiv preprint arXiv:2305.06147*.

Yiming Tan, Dehai Min, Yu Li, Wenbo Li, Nan Hu, Yongrui Chen, and Guilin Qi. 2023. Can ChatGPT Replace Traditional KBQA Models? An In-depth Analysis of the Question Answering Performance of the GPT LLM Family. *arXiv preprint arXiv:2303.07992*.

InternLM Team. 2023. Internlm: A multilingual language model with progressively enhanced capabilities. https://github.com/InternLM/InternLM.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Batra, et al. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. *arXiv preprint arXiv:2307.09288*.

Jianing Wang, Junda Wu, Yupeng Hou, Yao Liu, Ming Gao, and Julian McAuley. 2024. Instructgraph: Boosting large language models via graph-centric instruction tuning and preference alignment. *arXiv preprint arXiv:2402.08785*.

Keheng Wang, Feiyu Duan, Sirui Wang, Peiguang Li, Yunsen Xian, Chuantao Yin, Wenge Rong, and Zhang Xiong. 2023. Knowledge-driven cot: Exploring faithful reasoning in llms for knowledge-intensive question answering. *arXiv preprint arXiv:2308.13259*.

BigScience Workshop, :, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Sasha Luccioni, et al. 2022. BLOOM: A 176B-Parameter Open-Access Multilingual Language Model. *arXiv preprint arXiv:2211.05100*.

Yike Wu, Nan Hu, Sheng Bi, Guilin Qi, Jie Ren, Anhuan Xie, and Wei Song. 2023. Retrieve-Rewrite-Answer: A KG-to-Text Enhanced LLMs Framework for Knowledge Graph Question Answering. *arXiv preprint arXiv:2309.11206*.

Kun Xu, Yuxuan Lai, Yansong Feng, and Zhiguo Wang. 2019. Enhancing key-value memory neural networks for knowledge based question answering. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 2937–2947.

Aiyuan Yang, Bin Xiao, Bingning Wang, Borong Zhang, Ce Bian, Chao Yin, Chenxu Lv, Da Pan, Dian Wang, Dong Yan, Fan Yang, Fei Deng, Feng Wang, Feng Liu, Guangwei Ai, Guosheng Dong, Haizhou Zhao, Hang Xu, Haoze Sun, Hongda Zhang, Hui Liu, Jiaming Ji, Jian Xie, JunTao Dai, Kun Fang, Lei Su, Liang Song, Liu, et al. 2023. Baichuan 2: Open Large-scale Language Models. *arXiv preprint arXiv:2309.10305*.

Xi Ye, Semih Yavuz, Kazuma Hashimoto, Yingbo Zhou, and Caiming Xiong. 2022. RNG-KBQA: Generation augmented iterative ranking for knowledge base question answering. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 6032–6043.

Scott Wen-tau Yih, Matthew Richardson, Chris Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Donghan Yu, Sheng Zhang, Patrick Ng, Henghui Zhu, Alexander Hanbo Li, Jun Wang, Yiqun Hu, William Wang, Zhiguo Wang, and Bing Xiang. 2022. Decaf: Joint decoding of answers and logical forms for question answering over knowledge bases. *arXiv preprint arXiv:2210.00063*.

Biao Zhang, Barry Haddow, and Alexandra Birch. 2023. Prompting Large Language Model for Machine Translation: A Case Study. *arXiv preprint arXiv:2301.07069*.

Jing Zhang, Xiaokang Zhang, Jifan Yu, Jian Tang, Jie Tang, Cuiping Li, and Hong Chen. 2022. Subgraph retrieval enhanced model for multi-hop knowledge base question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (ACL)*.

Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alexander J Smola, and Le Song. 2018. Variational reasoning for question answering with knowledge graph. In *AAAI*.

Zihan Zhang, Meng Fang, Ling Chen, Mohammad-Reza Namazi-Rad, and Jun Wang. 2023. How do large language models capture the ever-changing world knowledge? a review of recent advances. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Jie Zhou, Xu Han, Cheng Yang, Zhiyuan Liu, Lifeng Wang, Changcheng Li, and Maosong Sun. 2019. GEAR: Graph-based evidence aggregating and reasoning for fact verification. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 892–901.

A Appendix

A.1 Large Language Models

We conducted extensive experiments on multiple 7B open-source LLMs, including popular models such as *Llama-2*, *Bloom*, *Vicuna*, *Internlm-2* and *Baichuan-2*.

Llama-2 is an LLM optimized for dialogue scenarios based on Llama 2, particularly suitable for handling KGQA tasks. *Vicuna* is an LLM fine-tuned based on Llama 1.

Bloom is an LLM trained on the Megatron-LM GPT2, utilizing unique decoder structures, normalization of the word embedding layer, linear bias attention position encoding with the GeLU activation function, and other advanced techniques.

Baichuan-2 is developed by Baichuan Intelligence, is a highly influential AI large-scale model. It integrates intent understanding, information retrieval, and reinforcement learning technologies, achieving high-performance results through supervised fine-tuning and alignment with human intent.

Internlm-2 is capable of efficiently supporting ultra-long contexts of up to 200,000 characters, achieving a leading level among open-source models in tasks such as Longbench and E-eval. Its comprehensive capabilities have shown all-around advancements over the previous generation of Internlm, and it possesses strong code interpretation and data analysis abilities.

A.2 Datasets

We conduct extensive experiments on three datasets: FactKG (Kim et al., 2023b), MetaQA (Zhang et al., 2018) and WebQSP (Yih et al., 2016).

FactKG is a fact-verification benchmark based on KG, containing 108K natural language statements verifiable via DBpedia (Lehmann et al., 2015), categorized into five reasoning types: One-hop, Conjunction, Existence, Multi-hop, and Negation.Furthermore, FactKG contains various linguistic patterns, including colloquial style statements as well as written style statements, to increase practicality.

MetaQA is a comprehensive benchmark for assessing question-answering systems, particularly those utilizing knowledge graphs. It comprises over 400K questions, including one-hop, two-hop, and three-hop reasoning. This dataset is crucial for evaluating knowledge graph-based question answering, especially in handling complex multi-hop reasoning and noisy inputs.

WebQuestionsSP is a KGQA benchmark containing full semantic parses in SPARQL queries for 4,737 questions (3,098 train, 1,639 test). It is built on Freebase and includes multi-hop questions, linked through topic entities, reasoning chains, and SPARQL queries. It provides semantic parses in SPARQL with standard Freebase entity identifiers, which can be directly executed on Freebase to return answers to questions.

A.3 Corpus Generation

Recognizing ChatGPT's exceptional abilities in understanding and generating text, as highlighted in recent research (Li et al., 2023a; Tahmid Rahman Laskar et al., 2023), we use the GPT-3.5-turbo API ($0.002 / 1K tokens) to generate training corpora, with the following steps:

Query Decoupling. Given a question q and a set of entities e, we insert them into a predefined generation template p_{dec} to obtain a text prompt. This text prompt is then input into ChatGPT to produce an output sequence $z = [z_1, ..., z_T]$, which includes sub-queries and their respective entity subsets.

Knowledge Reconstruction. Given an evidence subgraph $\mathcal{G}evi$ stored in triplet form, we first linearize it into a text format by concatenating the head entity, relation word, and tail entity to form textual triplets. We insert this sequence of triplets into a predefined template p_{evi}: "Your task is to transform a knowledge graph in triplets (or tuples) format into a single sentence, preserving the original words or expressions from the triplets as much as possible. The knowledge graph is: {graph}. The sentence is:". This prompt is then fed into ChatGPT, resulting in an output sequence $z = [z_1, ..., z_T]$ that contains the textualized evidence.

Joint Inference. Given a query q and an evidence sequence evi, we insert both into a predefined template p_{inf}, input it into ChatGPT, and the model will produce inference results and explanations based on the input.

A.4 Baselines

We compare RefKG with four types of baselines: 1) *Embedding-based methods*, 2) *Retrieve-augmented methods*, 3) *Prompting-based LLMs methods*, and 4) *Fine-tuned LLMs methods*. The details of each baseline are described below.

Embedding-based methods.

- KVMemNN (Xu et al., 2019) utilizes a Key-Value memory network to store triples and conducts multi-hop reasoning through iterative operations on the memory.
- EmbedKGQA (Saxena et al., 2020) approaches reasoning on knowledge graphs as a sequential link prediction problem by leveraging the embeddings of both entities and questions.
- NSM (He et al., 2021) employs a sequential model to replicate the multi-hop reasoning process.
- TransferNet (Shi et al., 2021) uses a graph neural network to capture the relevance between entities and questions for reasoning. process.

Retrieve-augmented methods.

- GraftNet (Sun et al., 2018) retrieves relevant subgraphs from knowledge graphs using entity linking.
- PullNet (Sun et al., 2019) trains a retrieval model that combines an LSTM and a graph neural network to retrieve a question-specific subgraph.
- SR+NSM (Zhang et al., 2022) introduces a relation-path retrieval mechanism to retrieve subgraphs for multi-hop reasoning.

Prompting-based LLMs methods.

- KB-Binder (Li et al., 2023b) is the first to enable few-shot in-context learning over KBQA tasks.
- KAPING (Baek et al., 2023b) propose to augment the knowledge directly in the input of LLMs.
- KG-GPT (Kim et al., 2023a) is a multi-purpose framework leveraging LLMs for tasks employing KGs. It comprises three steps: Sentence Segmentation, Graph Retrieval, and Inference, each aimed at partitioning sentences, retrieving relevant graph components, and deriving logical conclusions, respectively.
- StructGPT (Jiang et al., 2023) proposes an invoking linearization-generation procedure to support LLMs in reasoning on the structured data.
- ToG (Sun et al., 2024) enables LLM agent to interactively explore related entities and relations on KGs and perform reasoning based on the retrieved knowledge.
- FRAG (Gao et al., 2025) is a flexible modular KG-RAG framework that enhances LLM reasoning by estimating query complexity and applying tailored retrieval pipelines.

Fine-tuned LLMs methods.

- KD-CoT (Wang et al., 2023) retrieves pertinent knowledge from knowledge graphs to formulate faithful reasoning plans for LLMs.
- UniKGQA (Jiang et al., 2022) integrates graph retrieval and reasoning into a unified model with LLMs, achieving state-of-the-art performance on KGQA tasks.
- DECAF (Yu et al., 2022) synergizes semantic parsing and LLMs reasoning to jointly generate answers, achieving notable performance on KGQA tasks.
- Retrieve-Rewrite-Answer (Wu et al., 2023) propose an answer-sensitive KG-to-Text approach that can transform KG knowledge into well-textualized statements most informative for KGQA. A. Also, they propose a KG-to-Text enhanced LLMs framework for solving the KGQA task.
- InstructGraph (Wang et al., 2024) is a framework that empowers LLMs with the abilities of graph reasoning and generation by instruction tuning and preference alignment.

A.5 Implementation Details

The details of training hyperparameters are presented in Table 6.

Hyper-parameters	FactKG	MetaQA	WebQSP
training strategy	full	full	lora
epoch	3	3	50
sequence length	2048	256	2048
learning rate	1e-5	2e-5	5e-5
batch size	1	1	16
gradient accumulation	4	4	1
optimizer	AdamW	AdamW	AdamW
weight decay	0.01	0.01	0.01
deepSpeed stage	3	3	3

Table 6: Hyper-parameters of training.

FactKG. We extracted a subset of 40,000 data from the training set to generate our training corpus. Following quality control measures, we produced a total of 86,786 data instances, divided into three categories: 31,999 for question decomposition, 29,702 for evidence generation, and 24,085 for evidence reasoning. For the task of evidence subgraph retrieval, we configure the number of relations k to be either 2 or 5, and the score threshold α to 0.6. For a full-parameter fine-tuning of a 7b model using two A800-80G graphics cards, the memory consumption is approximately 140G, and it takes about 24 hours.

MetaQA. We extracted a subset of 30,000 data from the training set to create our training corpus. In the hyperparameter configuration, we set the number of selected relations k to 3, and the score threshold α to 0.7. Since the number of entities related to each question in the WebQSP dataset is smaller compared to FactKG, we directly treat the topic entity as the sole member of the entity set, in order to train the LLM's ability to predict the number of hops. For a full-parameter fine-tuning of a 7b model using two A800-80G graphics cards, the memory consumption is approximately 140G, and it takes about 16 hours.

WebQuestionsSP. We first extract SPARQL queries and their corresponding topic entities from the training set. Next, we parse these SPARQL queries and decompose them into multiple hops. By designing precise SPARQL query statements, we perform searches in Freebase, thereby obtaining inference chains represented in the form of triplets. By populating predefined task templates with the obtained ground truth data, we construct training datasets for each stage. And we set the number of selected relations k to 3, and the score threshold α to 0.6. Since the number of entities related to each question in the WebQSP dataset is smaller compared to FactKG, we directly treat the topic entity as the sole member of the entity set, in order to train the LLM's ability to predict the number of hops. Due to the small size of our training dataset, which contains only 3,098 entries, we use Lora for fine-tuning to prevent overfitting during the training process. For a lora fine-tuning of a 7b model using four A800-80G graphics cards, the memory consumption is approximately 240G, and it takes about 14 hours.

Stage	Total	Existence	Multi-hop	Other
Query Decoupling	62	10	18	34
Evidence Subgraph Retrieval	13	7	1	5
Joint Inference	25	6	3	16

Table 7: Statistics on 100 incorrect samples.

A.6 Training Details for Expert Model

We trained the Expert LLM using 30,000 annotated data entries, as detailed below:

Evidence score annotation. For each sub-query q_i and triplet format evidence t, we first employ the semantic similarity model DistilBERT to assign a similarity score, denoted as s, to represent the supportiveness of the evidence triplet toward the query.

For each sub-query q_i, we sort all evidence triplets t based on their scores, from highest to lowest. This set includes triplets that are relevant to the query as well as some that are noise. We then match these triplets with the ground truth. If a triplet from the ground truth is ranked among the top k, we retain it as part of the training data; otherwise, we filter it out.

It's important to note that we don't directly use all the collected annotated data for training. Instead, we first conduct a complete inference process with this data. If the final inference result is correct, we retain the annotated data as the gold score; if it's incorrect, we discard it. This approach ensures the high quality of the annotated data. Additionally, to minimize the influence of noise during the training process, we have eliminated anomalously high and low scores.

A.7 Error Analysis.

For the error analysis of the FactKG, see Table 7.

To explore the execution efficiency of each step, we perform an error analysis on FactKG. It was noted that errors predominantly arise in the Query

Decoupling stage, primarily due to the model's struggle in correctly identifying entities within sentences, a difficulty that is particularly pronounced in Multi-hop claims. This issue can lead to the alteration of entities mentioned in a sentence. A potential solution to mitigate such errors involves enhancing the model's sensitivity towards entity recognition.

A.8 Noise Analysis

We randomly selected 100 samples from the FactKG dataset and conducted a detailed analysis of the noise introduction and reduction in the decoupling, retrieval, scoring, and reconstruction steps, as shown in Table 8. We defined three statistical metrics:

- **Noise introduction**: Refers to the introduction of incorrect knowledge, conflicting knowledge, or loss of correct information at a particular step.
- **Noise reduction**: Refers to successfully removing incorrect or irrelevant knowledge at a particular step.
- **Correctness**: Indicates whether the current knowledge information contains correct knowledge.

The details of the noise flow in the four stages are as follows:

In the Query Decoupling: A small number of cases may experience partial entity information loss, leading to the introduction of noise.

In the Subgraph Retrieval: Since we aim to retrieve as much relevant knowledge as possible, it is inevitable to introduce some irrelevant knowledge and even knowledge that conflicts with correct information. Among them, some conflicting information may interfere with the results, while some irrelevant information has a minor impact.

In the Knowledge Refinement: Some incorrect and irrelevant triples are scored and removed, but there is also a small possibility that a few correct answers may be mistakenly filtered out.

In the Knowledge Reconstruction: While converting triples into textual information, the model performs implicit reasoning. During this process, the model may actively discard some incorrect or conflicting information and even correct erroneous information, but this may also result in the loss of a few correct pieces of information.

Type	Decoupling	Retrieval	Refinement	Reconstruction
Noise Introduction	9	24	2	3
Noise Reduction	-	-	16	6
Correctness	92	87	85	84

Table 8: Noise analysis of the FactKG.

Dataset	Number of triplets	Number of calls	Inference time
FactKG	10.11	4.8	2.4
WebQSP	19.76	4.4	2.1
MetaQA	-	5.1	1.9
Average	-	4.8	2.1

Table 9: Evaluation of computational efficiency.

The statistical results show that noise introduction is often difficult to completely avoid when handling complex problems. Through the collaborative operation of various tasks, particularly during the **Knowledge Refinement** and **Knowledge Reconstruction** stages, we effectively control noise, significantly mitigating its cumulative effects across tasks and reducing its impact on overall performance. This further validates the robustness and effectiveness of our approach in complex knowledge reasoning scenarios.

A.9 Evaluation of Computational Efficiency

We randomly selected 100 samples from each of the three datasets for the efficiency analysis, as shown in Table 9. We computed the average number of triples involved in each question, the average number of LLM calls, and the average inference time (in seconds).

A.10 Prompts

The 9-shot prompt templates for Query Decoupling, Evidence Subgraph Retrieval, and Joint Inference are respectively presented in Table 10, Table 11, and Table 12.

A.11 Qualitative Analysis

More qualitative results on FactKG and MetaQA are respectively presented in Table 13 and Table 14.

Prompt for query decoupling
Please decompose the given sentence into multiple single-hop sub-sentences, which can be represented by a triplet. Each entity subset should contain no more than two elements, entities can be duplicated across different subsets, and the union of multiple subsets should equal the original entity set. Generate the results in the format of (number). (Sentence), (entity set), using "##" to separate different entities. Refer to the following examples to complete the task: Examples) Sentence A: The City of Soldotna is the owner of the AIDAluna. Entity set: ['AIDAluna' ## awareaware'"The City of Soldotna"'] Answer: 1. The City of Soldotna is the owner of the AIDAluna., Entity set: ['AIDAluna' ## '"The City of Soldotna"'] Sentence B: Born in Gevelsberg, Alan Shepard was awarded the "Distinguished Service Medal". Entity set: ['Alan_Shepard' ## 'Distinguished_Service_Medal_(United_States_Navy)' ## 'Gevelsberg'] Answer: 1. Alan Shepard was awarded the "Distinguished Service Medal"., Entity set: ['Alan_Shepard' ## 'Distinguished_Service_Medal_(United_States_Navy)'] 2. Alan Shepard was born in Gevelsberg., Entity set: ['Alan_Shepard' ## 'Gevelsberg'] ..... Your Task) Query: ««QUERY»» Entity set: ««ENTITY_SET»» Answer:

Table 10: Prompt for query decomposition. ««QUERY»» and ««ENTITY_SET»» will be replaced with the corresponding query and entity set in the FactKG dataset.

Prompt for evidence subgraph retrieval
I will give you a set of words. Find the top ««K»» elements from relational words set which are most semantically related to the given sentence. You may select up to ««K»» words. If there is nothing that looks semantically related, pick out any ««K»» elements and give them to me. Examples) Sentence A: The City of Soldotna is the owner of the AIDAluna. Words set: ['status', 'owner', 'builder', 'shipOwner', 'shipBuilder', 'operator', 'shipOperator', 'shipClass'] Top 2 Answer: ['owner', 'shipOwner'] Sentence B: Born in Gevelsberg, Alan Shepard was awarded the "Distinguished Service Medal". Relational words set: ['birthPlace', 'mission', 'awards', 'rank', 'region', 'state', 'birthYear', 'country', 'type'] Top 2 Answer: ['birthPlace', 'awards'] ... Now let's find the top ««K»» elements. Query: ««QUERY»» Relational words set: ««RELATION_SET»» Top ««K»» Answer:

Table 11: Prompt for subgraph retrieval. «<QUERY»» and ««ENTITY_SET»» will be replaced with the corresponding query and Relational words set in the FactKG dataset. ««K»» will be replaced with the chosen hyperparameter k.

Prompt for joint inference
You should verify the claim based on the textual evidence. Each evidence is derived from one or several sentences generated from knowledge graph triplets. Verify the claim based on the evidence. (True means that everything contained in the claim is supported by the evidence.) Choose one of {True, False}, and give me one sentence explanation. Examples) Claim A: The City of Soldotna is the owner of the AIDAluna. Evidence: Lack of evidence. Answer: {False}, there is no evidence that The City of Soldotna is the owner of the AIDAluna. Claim B: Brandon Carter was born in England and graduated from the University of Cambridge where the current Chancellor is Leszek Borysiewicz. Evidence: Brandon Carter attended the University of Cambridge.Brandon Carter was born in England.Leszek Borysiewicz served as the Vice-Chancellor of the University of Cambridge. Answer: {True}, everything of the claim is supported by the evidence. Now let's verify the Claim based on the Evidence. Query: ««QUERY»» Evidence: ««EVIDENCE»» Answer:

Table 12: Prompt for joint inference.««QUERY»» and ««EVIDENCE»» will be replaced with the corresponding query on the FactKG dataset and evidence set generated in 3.3.

Type	Claim	Evidence Subgraph Graph	Textual Evidence generation	Prediction
One-hop	Do you know Agra Airport IATA Location Identifier is AGR.	[Agra_Airport, iataLocationIdentifier, "AGR"],	Agra Airport has an IATA location identifier of "AGR".	True
Conjunction	Doris Bures is the leader of Austria where Alfons Gorbach died in Styria.	[Austria, leader, Doris_Bures], [Alfons_Gorbach, placeOfDeath, Styria], [Doris_Bures, birthPlace, Austria]	Austria is the leader and birthplace of Doris Bures. Alfons Gorbach was born and died in Styria.	True
Existence	At least Dawn Butler had a successor!	[Dawn_Butler, successor, Paul_Boateng], [Dawn_Butler, birthPlace, England], [Dawn_Butler, predecessor, Sarah_Teather]	Dawn Butler has a successor named Paul Boateng. Dawn Butler was born in England. Dawn Butler has a predecessor named Sarah Teather.	True
Negation	I understand that Acura is not a division of Honda.	[Acura, owningCompany, Honda], [Honda, division, Acura], [Acura, owner, Honda]	Acura is owned by Honda and is also a division of Honda.	False
Multi-hop	It is located in Alan B Miller Hall in the United States.	[Alan_B_Miller_Hall, location, Williamsburg,_Virginia], [Williamsburg,_Virginia, country, United_States]	Alan B Miller Hall locates in Williamsburg,Virginia. Williamsburg,Virginia is in the United States.	True

Table 13: Qualitative results from FACTKG.

Task	Question	Evidence Subgraph Graph	Textual Evidence generation	Prediction
1-hop	Brian Backer appears in which movies?	(Moving Violations, starred_actors, Brian Backer)	Moving Violations, starring Brian Backer, is a film.	Moving Violations
	who is the writer of the film Habit?	(Habit, written_by, Larry Fessenden)	Habit, written by Larry Fessenden.	Larry Fessenden
	what kind of film is The Old Dark House?	(The Old Dark House, has_genre, Comedy) (The Old Dark House, has_genre, Horror)	The Old Dark House is a Comedy. The Old Dark House is a Horror genre.	Comedy
2-hop	what genres do the movies written by Anders Nilsson fall under?	(The Third Wave, written_by, Anders Nilsson), (The Third Wave, has_genre, Action)	The Third Wave, written by Anders Nilsson, is an action genre.	Action
	what are the primary languages in the movies directed by David Mandel	(EuroTrip, directed_by, David Mandel) (EuroTrip, in_language, German)	EuroTrip, directed by David Mandel, is a film in the German language.	German
	who is listed as director of Joseph Stein written films	(Fiddler on the Roof, written_by, Joseph Stein), (Fiddler on the Roof, written_by, Joseph Stein)	Fiddler on the Roof, written by Joseph Stein and directed by Norman Jewison, is a film.	Norman Jewison
3-hop	what genres do the films that share writers with Karate-Robo Zaborgar fall under?	(Karate-Robo Zaborgar, written_by, Noboru Iguchi), (RoboGeisha, written_by, Noboru Iguchi), (RoboGeisha, has_genre, Action)	Karate-Robo Zaborgar and RoboGeisha are written by Noboru Iguchi and they both belong to the genre of Action.	Action
	the movies that share writers with the movie Naqoyqatsi were released in which years?	(Naqoyqatsi, written_by, Godfrey Reggio), (Powaqqatsi, written_by, Godfrey Reggio), (Powaqqatsi, release_year, 1988)	Naqoyqatsi and Powaqqatsi were written by Godfrey Reggio and were released in 1988.	1988
	who is listed as screenwriter of the movies directed by the The Battle of Shaker Heights director?	(The Battle of Shaker Heights, directed_by, Kyle Rankin), (Infestation, directed_by, Kyle Rankin), (Infestation, written_by, Kyle Rankin)	The Battle of Shaker Heights and Infestation, directed by Kyle Rankin, were written by Kyle Rankin.	Kyle Rankin

Table 14: Qualitative results from MetaQA.