RGR-KBQA: Generating Logical Forms for Question Answering Using Knowledge-Graph-Enhanced Large Language Model

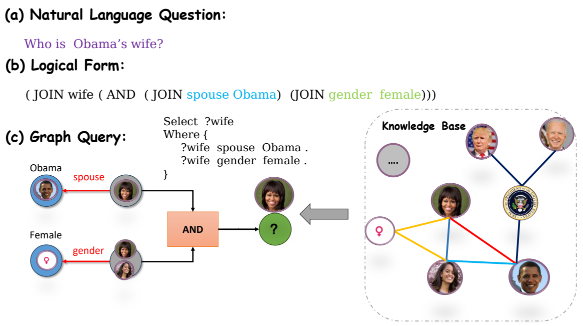
Tengfei Feng[1,2] , **Liang He**[1,2,3*]
[1] School of Computer Science and Technology, Xinjiang University, Urumqi 830017, China
[2] Xinjiang Key Laboratory of Signal Detection and Processing, Urumqi 830017, China
[3] Department of Electronic Engineering, and Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China
aryafeng53@gmail.com, heliang@mail.tsinghua.edu.cn

Abstract

In the field of natural language processing, Knowledge Base Question Answering (**KBQA**) is a challenging task that involves accurately retrieving answers from structured knowledge. Existing methods often face issues when generating query statements using LLMs, as the knowledge introduced may be imprecise and the models themselves may exhibit hallucination problems, leading to low accuracy, particularly when dealing with complex questions. To address these challenges, we introduce a novel semantic parsing approach called RGR-KBQA, which adopts a **R**etrieve-**G**enerate-**R**etrieve framework. The first retrieval step introduces factual knowledge from a knowledge graph to enhance the semantic understanding capabilities of LLMs, thereby improving generation accuracy of logical form. The second step uses a fine-tuned model to generate the logical form, and the final step involves unsupervised relation and entity retrieval to further enhance generation accuracy. These two retrieval steps help alleviate the hallucination problems inherent in LLMs. Experimental results show that RGR-KBQA demonstrate promising performance on CWQ and WebQSP datasets.

1 Introduction

Knowledge Base Question Answering (KBQA) is a classical task in Natural Language Processing (NLP) that involves answering natural language questions based on facts from large-scale knowledge bases, such as Freebase (Bollacker et al., 2008), Wikidata (Vrandečić and Krötzsch, 2014) and DBpedia (Auer et al., 2007). These Knowledge bases are typically organized as sets of factual triples, such as (h, r, t), which indicate that there is a relation r between the head entity h and the tail entity t. Previous KBQA methods have primarily adopted two main approaches: information retrieval and semantic parsing.



```
Select ?wife
Where {
    ?wife spouse Obama .
    ?wife gender female .
}
```

Figure 1: An example of KBQA task to answer a natural language question by converting the question to a graph query.

Information Retrieval-based methods (IR) (He et al., 2021; Zhang et al., 2022; Ye et al., 2022) directly retrieve and rank answers from the knowledge base by considering the information conveyed in the questions. These methods typically follow a pipeline approach, consisting of two key steps: (1) Subgraph Extraction. A subgraph relevant to the question is extracted from the knowledge base based on the entities and relations. (2) Answer Ranking. Potential answers are ranked based on their relevance to the question. IR methods are generally simpler to implement but often suffer from lower accuracy.

Semantic Parsing-based methods (SP) aim to convert unstructured natural language question into structured logical forms, such as S-expression (Gu et al., 2021). As shown in Figure 1, S-expression is a special type of logical form which can be converted into an executable graph database query such as SPARQL (Pérez et al., 2009) to obtain answers. In the early work of semantic parsing-based knowledge graph question answering (Yih et al., 2015; Hu et al., 2018), answering question primarily relied on various manually defined rules, features, and templates.

[*] Corresponding author

Proceedings of the 31st International Conference on Computational Linguistics, pages 3057–3070
January 19–24, 2025. ©2025 Association for Computational Linguistics

Improvements in performance of these methods mainly depended on traditional linguistics and human expertise, rather than high-dimensional features extracted by deep learning models. However, for complex question and large-scale datasets, this methods requires a higher level of setting for template rules, often leading to suboptimal performance and difficulty in handling complex problems. Deep learning has opened up new avenues for research in semantic parsing, with neural network models playing a pivotal role. CBR-KBQA (Das et al., 2021) employs a memory module to store question-logical form pairs. For new questions, it retrieves similar pairs and leverages their logical forms for generation. This method, however, is memory-intensive and suffers from retrieval efficiency issues as the memory grows. RNG-KBQA (Ye et al., 2022) addresses this by using predefined rules to generate candidate logical forms, which are then refined by a generative model. However, this approach heavily relies on the quality of the predefined rules. (Cao et al., 2022; Xie et al., 2022; Yu et al., 2023; Zhang et al., 2023) utilize seq2seq models like T5 (Raffel et al., 2020) to either directly generate the corresponding logical forms or directly generate the answers. The former differs in how to improve the generation accuracy, but these methods all suffer from the problem that, as generative models, the final generated logical forms may not be executable. The latter generally has lower generation accuracy.

Large Language Models (LLMs) (Ouyang et al., 2022; OpenAI et al., 2023; Touvron et al., 2023) have demonstrated remarkable performance across various natural language processing tasks. These models can achieve satisfactory performance even in few-shot or zero-shot scenarios. some study (Jiang et al., 2023a; Gu et al., 2023; Sun et al., 2024) utilize the thinking capabilities of LLMs to find answer by retrieving from the graph in a step-wise manner. Chat-KBQA (Luo et al., 2024) proposed a novel pipeline:generate-then-retrieve. This approach can significantly improve the accuracy of the final generated logical forms. However, this method does not leverage information from the knowledge graph. In some cases, where the model's capability is weak, the accuracy of generation can be significantly affected.

In such a background, we propose a novel semantic parsing approach that leverages external factual knowledge from the knowledge graph to guide the model in generating better logical forms. Specifically, given a question and a knowledge base, We adopt Zhang's (Zhang et al., 2022) method to find the most relevant relation chain, which we define as the composition of intermediate triples with the highest score starting from the topic entity in the question and reaching the answer entity. Compared to other information retrieval methods, this method retrieves information that is more relevant to the question and more fine-grained. This relation is highly correlated with the subsequently generated logical form. Secondly, we fine-tune LLMs to generate logical forms. LLMs inherently contains a vast amount of knowledge, and after fine-tuning, they can better understand the question. By combining the information extracted from the knowledge graph, LLMs can produce better results. Finally, we perform unsupervised phrase-level semantic retrieval to further improve the generation accuracy.

In summary, we make the following contributions in this paper:

- We propose a novel retrieve-generate-retrieve framework: this approach first retrieves factual knowledge from a knowledge graph to enhance the semantic understanding of LLMs. Compared to previously added information in prompts, the information we add is more fine-grained and closer to logical forms. Then, we use a fine-tuned model to generate logical forms. Finally, through unsupervised relation and entity retrieval, we further improve the accuracy of generation. Since the final answer is obtained through SPARQL queries, the method itself has a certain degree of interpretability.

- Experimental results demonstrate that our method achieves highly competitive results on WebQSP (Yih et al., 2016) and CWQ (Talmor and Berant, 2018), with a significant advantage on the more complex CWQ dataset compared to other methods. This indicates that by incorporating relevant information from the knowledge graph, we can alleviate the hallucination problem of LLMs to some extent, while also inproving the model's comprehension ability and generation accuracy.

2 Related Work

Knowledge Base Question Answering

Existing Knowledge Base Question Answering (KBQA) methods can be broadly categorized into Information Retrieval-based(IR-based) and Semantic Parsing-based (SP-based) methods. Recently there have been some KBQA methods based on large language models (LLM-based) as well.

(a) **IR-based KBQA Methods** Information retrieval based on the knowledge base for question answering primarily involves retrieving relevant factual knowledge from the graph or other knowledge bases to answer given question. Existing research (Miller et al., 2016; Sun et al., 2019; Saxena et al., 2020; He et al., 2021; Shi et al., 2021; Zhang et al., 2022) focuses on retrieving relevant triples or text from knowledge bases based on natural language question to construct subgraph. Model then perform reasoning on these subgraph to obtain final answer.

(b) **SP-based KBQA Methods** These approaches aim to transform natural language questions into executable logical forms (e.g., SPARQL, query graphs) on knowledge bases. Some research (Yih et al., 2016; Lan et al., 2019; Chen et al., 2019; Bhutani et al., 2019; Lan and Jiang, 2020; Jiang et al., 2023b) utilize strategies of step-wise query graph generation and search for semantic parsing. This process begins by extracting initial entities from the query and then gradually expanding the graph by adding nodes and edges, followed by pruning irrelevant components and correcting errors. Alternatively, other studies (Das et al., 2021; Ye et al., 2022; Cao et al., 2022; Shu et al., 2022; Hu et al., 2022b; Xie et al., 2022; Yu et al., 2023; Zhang et al., 2023) have employed seq2seq models to directly generate S-expression. These approaches aim to enhance the generation of S-expression. While offering high automation and the ability to handle complex language inputs, these methods still have room for improvement in terms of accuracy.

(c) **LLM-based KBQA Methods** Thanks to LLMs' powerful few-shot even zero-shot capabilities, they can often provide correct answers to common-sense and factual question. some research (Jiang et al., 2023a; Gu et al., 2023; Sun et al., 2024) leverage large language model to extract relevant information and search for knowledge related to the query, then utilize this information in conjunction with the large language model to generate accurate responses.

In this paper, we propose RGR-KBQA, an SP-based method that enhances question answering accuracy by fine-tuning LLMs and incorporating factual knowledge from knowledge base. A novel retrieve-generate-retrieve approach is introduced to improve the quality of logical form generation. This method presents a new pipeline for leveraging large language model in semantic parsing for question answering tasks.

3 Preliminaries

In this section, we define three basic concepts of our work: knowledge graph, the optimal relation chain, SPARQL and the logical form, followed by the problem statement for KBQA tasks.

Definition 1: Knowledge Base (KB). A knowledge Base is a structured representation of facts, consisting of entities, relationships, and attributes. KBs are usually represented using the Resource Description Framework (RDF) format: $\mathcal{K} = \{(s, r, o) \mid s \in \mathcal{E}, r \in \mathcal{R}, o \in \mathcal{E} \cup \mathcal{L}\}$, where s is an entity, r is a relation, and o can be an entity or literal. Each entity $e \in \mathcal{E}$ in entity set $\mathcal{E}$ is represented by a unique Id, e.g. e. id = "m.02mjmr". We can perform the query operations to get the corresponding label or name. For example, e.label = "Barack Obama". Each relation $r \in \mathcal{R}$ in the relation set $\mathcal{R}$ has a label in a specific format, e.g. r = "people.person.place_of_birth".

Definition 2: Optimal Relation Chain. We believe that the probability of triples connected to the topic entity (the entity mentioned in the question) becoming the answer is very high. Among these connected triples, the ones with relationships most relevant to the question are more likely to be the answer. By training a scoring model that calculates the relevance between different relationships and the question. The optimal relation chain is determined by selecting the path with the maximum score that connects the topic entity and the answer.

Definition 3: SPARQL and Logical Form. SPARQL is a standard query language used for querying RDF data. It provides a flexible and expressive way to retrieve information stored in KBs. A logical form is a structured representation of a natural language question. Taking the S-expression as an example, a logical form is usually projection and various operators are composed. Projection operation refers to performing a one-hop query operation on the head entity s or the tail entity o in a triple (s, r, o), where $(?, r, o)$ is denoted as [JOIN r o], while $(s, r, ?)$ is denoted as [JOIN [R r] s]. There exists a mapping between logical forms and SPARQL query, allowing for a direct translation from logical form into equivalent SPARQL query language.

Problem Statement. For KBQA task, given a natural language question q and a knowledge base $\mathcal{K}$, We first need to obtain the optimal relation chain related to the question: $chain = SR(\mathcal{K}, q)$, where SR is the function for obtaining the most relevant relationship to the question. Then combining the previously extracted optimal relation chain, we convert the question q into a logical form $\mathcal{F} = SP(q, chain)$, where $SP(.)$ is a semantic parsing function. As noted above, the converted logical form can be transformed into an equivalent SPARQL query $q_{sparql} = Convert(\mathcal{F})$, where $Convert(.)$ is the fixed conversion function. Finally the final set of answers $A = Execute(q_{sparql} \mid \mathcal{K})$ is obtained by executing q against $\mathcal{G}$, where $Execute(.)$ is the query execution function.

4 Methodology

This section provides an overview of the RGR-KBQA framework as depicted in Figure 2. We delineate the processes of extracting optimal relation chain, fine-tuning a large language model, generating logical forms via fine-tuning, conducting unsupervised entity and relation retrieval, and performing explainable query execution.

4.1 Overview of RGR-KBQA

RGR-KBQA is a retrieve-generate-retrieve KBQA framework that leverages fine-tuned large language model (LLMs). Given a question and a knowledge base, RGR-KBQA first extracts the most relevant relation chain. It then efficiently fine-tunes open-source LLMs using a KBQA dataset consisting of question-answer pairs <natural language question + optimal relation chain, logical form>. The fine-tuned LLMs are subsequently used to transform natural language questions into corresponding logical form through semantic parsing. RGR-KBQA then performs phrase-level retrieval of entities and relations within these logical forms and searches for logical forms that can be converted into executable SPARQL query on the knowledge graph. Finally, by executing the converted SPARQL query, the final answer is generated.

4.2 Extract The Optimal Relation Chain

Refer to the trainable subgraph retrieval model proposed by (Zhang et al., 2022), for a question q with entity a as the answer, the goal of subgraph retrieval is to find a subgraph $\mathcal{G}$ that maximizes the probability $p(a \mid \mathcal{G}, q)$ as shown in Equation 1. Since common KBQA datasets only provide (question, answer) pairs without intermediate reasoning processes, the shortest path from the topic entity to the answer entity is used as a supervisory signal to guide model learning.

$$p(a \mid \mathcal{G}, q) = \sum_{\mathcal{G}_{sub}} p_\phi(a \mid q, \mathcal{G}_{sub}) p_\theta(\mathcal{G}_{sub} \mid q) \quad (1)$$

Specifically, starting from the topic entity, the similarity between relation r and question q is calculated using Equation 2, where $f(q)$ and $h(r)$ are embeddings obtained from the pre-trained RoBERTa (Liu et al., 2020) model.

$$score(q, r) = f(q) \cdot h(r) \quad (2)$$

During the subgraph expansion process, when select a new relation, the historical path influences the choice of current path. Therefore, SR incorporates previously selected paths when expanding the path. For example, at the t-th step of subgraph expansion, the previously selected relation chain $[r_1, r_2, ..., r_t]$ is combined with the question to jointly determine the path selection at the t-th step, i. e.,$f(q^{(t)}) = RoBERTa([q; r_1, r_2, ..., r_t])$. At the t-th hop, the probability of selecting relation r is as shown in Equation 3, where "END" is a virtual relation indicating the end. The expansion automatically stops when the probability is less than 0.5.

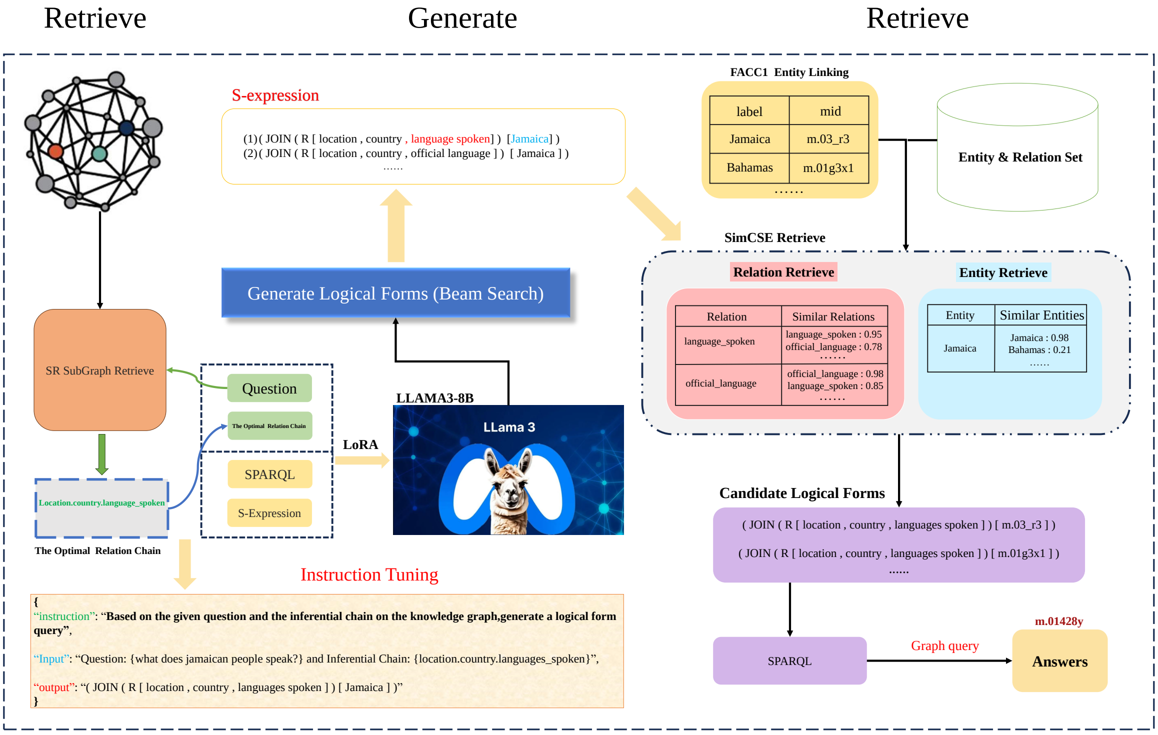
$$p(r \mid q^{(t)}) = \frac{1}{1 + exp(s(q^{(t)}, END) - s(q^{(t)}, r))} \quad (3)$$

The calculation score for path $path$ is shown in Equation 4. The path with the highest score is selected as the relation chain with the highest relevance to the question, and it is called the optimal relation chain.

$$p(path \mid q) = \prod_{t=1}^{|path|} p(r_t \mid q^{(t)}) \quad (4)$$

4.3 Fine-Tuning on LLMs

To construct an instruction-tuned dataset, we first preprocess the data. Specifically, we replace the


Retrieve
Generate
Retrieve
S-expression
(1)(JOIN (R [location , country , language spoken]) [Jamaica])
(2)(JOIN (R [location , country , official language]) [Jamaica])
……
Generate Logical Forms (Beam Search)
SR SubGraph Retrieve
Question
The Optimal Relation Chain
SPARQL
S-Expression
LoRA
LLAMA3-8B
LLama 3
Location.country.language_spoken
The Optimal Relation Chain
Instruction Tuning
FACC1 Entity Linking
Entity & Relation Set
SimCSE Retrieve
Relation Retrieve
Entity Retrieve
Candidate Logical Forms
(JOIN (R [location , country , languages spoken]) [m.03_r3])
(JOIN (R [location , country , languages spoken]) [m.01g3x1])
……
SPARQL
Graph query
m.01428y
Answers

label	mid
Jamaica	m.03_r3
Bahamas	m.01g3x1
……	

Relation	Similar Relations
language_spoken	language_spoken : 0.95 official_language : 0.78 ……
official_language	official_language : 0.98 language_spoken : 0.85 ……

Entity	Similar Entities
Jamaica	Jamaica : 0.98 Bahamas : 0.21 ……

```
{
"instruction": "Based on the given question and the inferential chain on the knowledge graph,generate a logical form query",

"Input": "Question: {what does jamaican people speak?} and Inferential Chain: {location.country.languages_spoken}",

"output": "( JOIN ( R [ location , country , languages spoken ] ) [ Jamaica ] )"
}
```

Figure 2: The overview of RGR-KBQA framework for retrieve-generate-retrieve KBQA method.The framework consists of three stages: retrieving relevant relationships, fine-tuning a large language model to generate a logical form, refining the logical form through retrieval, and finally executing a graph query to produce the answer.

semantically meaningless entity IDs in SPARQL with corresponding entity labels, such as converting "m.06w2sn5" to "Justin Bieber". Compared to ID, textual label can provide more information. Then, we take the question and the optimal relation chain as the input and the processed logical form as the corresponding output. As shown in Figure 3, we construct such a dataset and then fine-tune the LLMs. During fine-tuning, we employ Parameter Efficient Fine-Tuning (**PEFT**) techniques (Mangrulkar et al., 2022), including various efficient fine-tuning methods like LoRA (Hu et al., 2022a), QLoRA (Dettmers et al., 2023), P-tuning v2 (Liu et al., 2021), and Freeze (Geva et al., 2021), to minimize the cost of fine-tuning LLMs with a large number of parameters. After comprehensively considering model performance and computational resources, we choose LoRA as the primary fine-tuning technique and LLAMA3-8b (AI@Meta, 2024) as the base model for fine-tuning.

4.4 Unsupervised Retrieval of Entities and Relations

After fine-tuning, LLMs have acquired the knowledge for semantic parsing, enabling them to transform unstructured natural language question into structured logical form in most cases. For the fine-tuned model, approximately 63.5% of the generated logical form exactly match the ground truth logical form. If beam search is employed, about 77.4% of the candidate logical form generated by LLMs are correct. When we remove entities and relations from the generated logical forms, such as "(AND (JOIN [ ] [ ]) (JOIN (R [ ]) (JOIN (R [ ]) [ ])))", approximately 92% of the skeletons are correct. This indicates that the fine-tuned LLMs have achieved excellent semantic parsing results and can generate high-quality logical form.

Thanks to the powerful generation and comprehension capabilities of LLMs, and we adopt the (Luo et al., 2024) methods, we use an unsupervised approach for retrieval in the retrieval stage. This method involves phrase-level retrieval and replacement of entities and relations in candidate logical forms. Specifically, for a list of generated candidate logical forms $\mathcal{C}$, we iterate over all log-

Instruction	Based on the given question and the inferential chain on the knowledge graph, generate a logical form query.
Input	Question: {What is the name of Justin Biebers brother?} and Inferential Chain: {people. person. sibling_s}
Output	(AND (JOIN [people, person, gender] [Male]) (JOIN (R [people, sibling relationship, sibling]) (JOIN (R [people, person, sibling s]) [Justin Bieber])))

Figure 3: An example of instruction fine-tuning data.

ical forms. First, we perform entity retrieval. For each entity e in a logical form F, we calculate the similarity between this entity e and every entity e' in the entity set $\mathcal{E}$ of the knowledge base $\mathcal{K}$:

$$s_{e,e'} = SimiEntities(e, e') \tag{5}$$

The similarity calculation here employs an unsupervised phrase-level approach. Common unsupervised retrieval methods include SimCSE (Gao et al., 2021), Contriever (Izacard et al., 2022), and BM25 (Robertson and Zaragoza, 2009). In this paper, we choose SimCSE as the unsupervised retrieval method. Then, we rank the entities based on score and obtain the top_k entities. Similarly, we perform the same process for relations and obtain the top_k relations. The main reason for using an unsupervised retrieval method is that it does not require additional training and can directly retrieval the top_k most semantically similar candidates, simplifying the entire question answering task. After completing these steps, we obtained the candidate logical forms $\mathcal{C}'$. The $\mathcal{C}'$ will be transformed into equivalent SPARQL query in the subsequent steps.

4.5 Interpretable Query Execution

After retrieval, we obtain the final list of candidate logical forms, denoted as $\mathcal{C}'$. We iterate over each logical form F in this list and convert it into an equivalent SPARQL query using the function $q = Convert(F)$. During the conversion process, it's possible that the resulting SPARQL query is incorrect or cannot be executed. In such cases, we simply find the first executable query and execute it to obtain the final answer $A = Execute(q \mid \mathcal{K})$. Since SPARQL query statements explicitly provide the reasoning path, the entire reasoning process is interpretable. Overall, RGR-KBQA leverages information from external knowledge base to enhance the large language model's ability to generate logical forms, and the final execution result, being a SPARQL query executed on a knowledge base, is inherently interpretable.

5 Experiments

5.1 Experimental Setup

Datasets. All experiments were conducted on two standard question answering datasets: WebQuestionSP (WebQSP) (Yih et al., 2016) and ComplexWebQuestions (CWQ) (Talmor and Berant, 2018). WebQSP dataset contains a total of 4737 natural language questions with corresponding SPARQL queries, while CWQ dataset contains 34689 natural language questions. Both datasets rely on the Freebase knowledge base (Bollacker et al., 2008). Table 1 presents detailed statistics of these datasets.

Dataset	#Train	#Validation	#Test
WebQSP	2,848	250	1,639
CWQ	27,639	3,519	3,531

Table 1: Data statistics. The number of QA pairs for training, validing and testing are presented.

Evaluation Metrics. Following the previous work (Shu et al., 2022; Yu et al., 2023), we adopted the F_1 score, Hits@1, and Accuracy (Acc) as our evaluation metrics. Acc is used to measure the proportion of correct answers predicted by the system across all questions. Hits@1 is a standard evaluation that measures the ratio of the top-scoring entity among all test samples belonging to the correct answer. Since some questions have multiple answers, we also predict the answers by the optimal threshold searched on the validation set and evaluate their F_1 score.

Hyperparameters and Environment. Firstly, regarding the dataset, since we are conduction knowledge graph based question answering, we have excluded the 57 questions from the origin WebQSP dataset, whose answers are not entity type. Secondly, during the fine-tuning process of LLMs, the learning rate used for Llama3-8b is 1e-4, and for Llama2-7b it is 6e-5. The batch size is set to 4 for both. The LoRA rank is 8, LoRA alpha is 32, and LoRA dropout is 0.1. Finally, all experiments are conducted on a single NVIDIA 4090 GPU.

Baselines. In mainstream question-answering methods, there are information retrieval based methods (IR) and semantic parsing based methods (SP). Recently with the rapid development of large models, there has also been relevant research on question answering using LLMs. We select various models from different methods for comparison. For a detailed introduction to these baseline models, please refer to Appendix A

5.2 Results

Table 2 presents the experimental results of our retrieval-generation-retrieval framework for KBQA conducted on both the WebQSP and CWQ datasets. Our RGR-KBQA model achieves highly competitive results on both datasets. Notably, our model achieves the most significant improvements on the commonly used metrics of Hits@1 and F1. Compared to the best baseline model, our model achieves 1.1% increase in F1 and 2.5% increase in Hits@1 on the WebQSP dataset, and reaches a comparable F1 score to the best performing model on the CWQ dataset, with a 1.3% improvement in Hits@1. This indicates that incorporating relational information extracted from the knowledge graph can effectively help large language models generate corresponding logical forms.

Model	WebQSP			CWQ		
	F1	Hits@1	Acc	F1	Hits@1	Acc
IR-based KBQA Methods						
KV-Mem	34.5	46.7	-	15.7	21.1	-
PullNet	-	68.1	-	-	47.2	-
EmbedKGQA	-	66.6	-	-	44.7	-
NSM+h	67.4	74.3	-	44.0	48.8	-
TransferNet	-	71.4	-	-	48.6	-
Subgraph Retrieval	64.1	69.5	-	47.1	50.2	-
SP-based KBQA Methods						
STAGG	71.7	-	63.9	-	-	-
UHop	68.5	-	-	29.8	-	-
Topic Units	67.9	68.2	-	36.5	39.3	-
QGG	74.0	73.0	-	40.4	44.1	-
UniKGQA	72.2	77.2	-	49.4	51.2	-
CBR-KBQA	72.8	-	69.9	70.0	70.4	67.1
RnG-KBQA	75.6	-	71.1	-	-	-
Program Transfer	76.5	74.6	-	58.7	58.1	
TIARA	78.9	75.2	-	-	-	-
GMT-KBQA	76.6	-	**73.1**	**77.0**	-	72.2
UnifiedSKG	73.9	-	-	68.8	-	-
DecAF	78.8	82.1	-	-	70.4	-
FC-KBQA	76.9	-	-	56.4	-	-
LLM-based KBQA Methods						
StructGPT	72.6	-	-	-	-	-
Pangu	79.6	-	-	-	-	-
ToG	-	82.6	-	-	69.5	-
ChatKBQA*	78.3	81.5	70.7	75.4	80.7	71.1
RGR-KBQA(ours)	**80.7**	**84.5**	72.1	76.6	**82.0**	**72.2**

Table 2: KBQA comparison of RGR-KBQA with other baselines on WebQSP and CWQ datasets.* denotes that the results were replicated using our experimental setup and resources. The best results in each metric are in **bold**.

5.3 Ablation Study

5.3.1 Effectiveness of Optimal Relation Chain

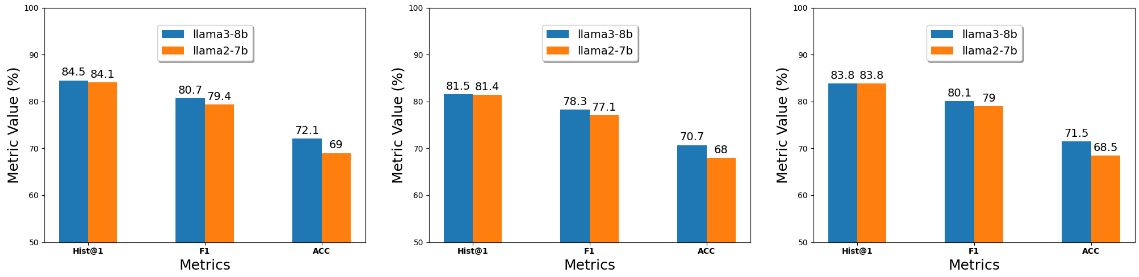
Unlike other approaches that use LLMs to generate logical forms and then retrieve answers, our method adds an extra step: extracting the most relevant relationship chain (**optimal relation chain**) from the question and knowledge bases. This optimal relation chain, along with the query, is jointly input into the LLM to refine logical form generation. To evaluate the effectiveness of the extracted optimal relation chain, we conducted ablation experiments under four conditions: using the optimal relation chain (RGR-KBQA), without it (w/o SR), using the two highest-scoring relation chains (two chains), and using a randomly selected relation chain (random). As shown in Table 3, omitting the optimal relation chain (w/o SR) and using only the question led to a decline in all metrics on both the WebQSP and CWQ datasets. This highlights the importance of relation chains in improving semantic understanding and generating accurate logical forms. Introducing random relation chains resulted in slight performance degradation, indicating that irrelevant information can impair model accuracy.

Model	WebQSP			CWQ		
	F1	Hits@1	Acc	F1	Hits@1	Acc
RGR-KBQA	**80.7**	**84.5**	**72.1**	**76.6**	**82.0**	**72.2**
w/o SR	78.3	81.5	70.7	75.4	80.8	71.1
two chains	80.1	83.8	71.5	76.2	81.8	71.9
random	77.5	81.4	68.5	75.2	80.7	71.0

Table 3: Ablation study on optimal relationship chains.

5.3.2 Is it the LLMs at work?

To assess the generalizability of our extraction method and ensure that performance improve-



(a) (b) (c)

Figure 4: A performance comparison between **Llama2-7b** and **Llama3-8b** models on WebQSP.The figure presents three varying experimental setups: (a) with the optimal chain, (b) w/o SR , (c) with two chains.

ments aren't solely due to the LLMs emergent abilities, we compared the performance of models with different model sizes. As shown in Figure 4, the performance difference between Llama 2-7B and Llama 3-8B is minimal across all experiments, with both models improving when using our extracted information. This suggests that simply increasing model size offers limited gains for complex tasks like semantic parsing. Incorporating relevant external information, however, significantly boosts contextual understanding and parsing capabilities, proving to be more effective than merely enlarging the model.

5.3.3 Effectiveness of Final Retrieval

Our framework employs a retrieve-generate-retrieve architecture. Prior experiments have demonstrated that eliminating the initial retrieval phase adversely impacts the model's performance. To further investigate this, we conducted an ablation study by removing the final retrieval stage, a common practice in large language model-based systems. As depicted in Table 4, the removal of the final retrieval stage led to a decline in all evaluation metrics. This finding can be attributed to the fact that while the logical forms generated by large language models might contain errors, they typically capture the correct underlying skeleton with high precision (over 92%). The final retrieval step enables the retrieval of diverse entities and relations based on similarity scores, guaranteeing that a viable SPARQL query can be constructed, thereby substantially enhancing the likelihood of generating accurate answers.

5.4 Case Study

By conducting a comparative analysis of generation results with and without the incorporation of optimal relation chains, we observed that LLMs, when deprived of optimal relation chains, tend to produce detailed yet inaccurate or even erroneous outputs. Conversely, by introducing extracted optimal relation chains, we can effectively guide the generation process of LLMs, resulting in significantly improved accuracy. For a detailed examination of these case studies, please refer to Appendix B.

Model	WebQSP			CWQ		
	F1	Hits@1	Acc	F1	Hits@1	Acc
RGR-KBQA	**80.7**	**84.5**	**72.1**	**76.6**	**82.0**	**72.2**
w/o final retrieval	79.9	83.6	71.4	74.7	79.1	70.8

Table 4: An ablation study to evaluate the role of the final retrieval step in enhancing the accuracy of RGRK-BQA.

6 Conclusion

In this paper, we introduced the RGR-KBQA framework, a novel approach that enhances knowledge base question answering by leveraging LLMs and knowledge graphs. Our method integrates a retrieve-generate-retrieve pipeline, which significantly improves the accuracy of logical form generation and question answering. The experimental results on WebQSP and CWQ datasets demonstrate that RGR-KBQA outperforms existing methods, highlighting the effectiveness of incorporating optimal relation chains and fine-tuning LLMs. This approach not only reduces hallucinations but also enhances the interpretability and precision of the generated answers. Future work will explore the potential of further integrating external knowledge sources to refine semantic parsing capabilities.

7 Limitations

Our approach is heavily reliant on the quality of the underlying knowledge graph. Inaccuracies or incompleteness in the knowledge graph can directly impact the accuracy and reliability of our system's output. Moreover, the computational demands of fine-tuning large language models present significant scalability challenges.

Acknowledgments

This work was supported by the National Science and Technology Major Project (2022ZD0115801) and the Key R&D Program in Xinjiang Uygur Autonomous Region (2022B03019-6).

References

AI@Meta. 2024. Llama 3 model card.

Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. 2007. Dbpedia: A nucleus for a web of open data. In international semantic web conference, pages 722–735. Springer.

Nikita Bhutani, Xinyi Zheng, and H V Jagadish. 2019. Learning to answer complex questions over knowledge bases with query composition. In Proceedings of the 28th ACM International Conference on Information and Knowledge Management, CIKM '19, page 739748, New York, NY, USA. Association for Computing Machinery.

Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data, SIGMOD '08, page 12471250, New York, NY, USA. Association for Computing Machinery.

Shulin Cao, Jiaxin Shi, Zijun Yao, Xin Lv, Jifan Yu, Lei Hou, Juanzi Li, Zhiyuan Liu, and Jinghui Xiao. 2022. Program transfer for answering complex questions over knowledge bases. In Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 8128–8140, Dublin, Ireland. Association for Computational Linguistics.

Zi-Yuan Chen, Chih-Hung Chang, Yi-Pei Chen, Jijnasa Nayak, and Lun-Wei Ku. 2019. UHop: An unrestricted-hop relation extraction framework for knowledge-based question answering. In Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 345–356, Minneapolis, Minnesota. Association for Computational Linguistics.

Rajarshi Das, Manzil Zaheer, Dung Thai, Ameya Godbole, Ethan Perez, Jay Yoon Lee, Lizhen Tan, Lazaros Polymenakos, and Andrew McCallum. 2021. Case-based reasoning for natural language queries over knowledge bases. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 9594–9611, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs. arXiv e-prints, arXiv:2305.14314.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yu Gu, Xiang Deng, and Yu Su. 2023. Don't generate, discriminate: A proposal for grounding language models to real-world environments. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 4928–4949, Toronto, Canada. Association for Computational Linguistics.

Yu Gu, Sue Kase, Michelle Vanni, Brian Sadler, Percy Liang, Xifeng Yan, and Yu Su. 2021. Beyond i.i.d.: Three levels of generalization for question answering on knowledge bases. In Proceedings of the Web Conference 2021, WWW '21, page 34773488, New York, NY, USA. Association for Computing Machinery.

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. In Proceedings of the 14th ACM International Conference on Web Search and Data Mining, WSDM '21, page 553561, New York, NY, USA. Association for Computing Machinery.

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

Sen Hu, Lei Zou, Jeffrey Xu Yu, Haixun Wang, and Dongyan Zhao. 2018. Answering natural language questions by subgraph matching over knowledge graphs. IEEE Transactions on Knowledge and Data Engineering, 30(5):824–837.

Xixin Hu, Xuan Wu, Yiheng Shu, and Yuzhong Qu. 2022b. Logical form generation via multi-task learning for complex question answering over knowledge bases. In Proceedings of the 29th International Conference on Computational Linguistics, pages 1687–1696, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. Transactions on Machine Learning Research.

Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Xin Zhao, and Ji-Rong Wen. 2023a. 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, pages 9237–9251, Singapore. Association for Computational Linguistics.

Jinhao Jiang, Kun Zhou, Xin Zhao, and Ji-Rong Wen. 2023b. UniKGQA: Unified retrieval and reasoning for solving multi-hop question answering over knowledge graph. In The Eleventh International Conference on Learning Representations.

Yunshi Lan and Jing Jiang. 2020. Query graph generation for answering multi-hop complex questions from knowledge bases. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 969–974, Online. Association for Computational Linguistics.

Yunshi Lan, Shuohang Wang, and Jing Jiang. 2019. Knowledge base question answering with topic units. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19, pages 5046–5052. International Joint Conferences on Artificial Intelligence Organization.

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Lam Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2021. P-Tuning v2: Prompt Tuning Can Be Comparable to Fine-tuning Universally Across Scales and Tasks. arXiv e-prints, arXiv:2110.07602.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach.

Haoran Luo, Haihong E, Zichen Tang, Shiyao Peng, Yikai Guo, Wentai Zhang, Chenghao Ma, Guanting Dong, Meina Song, Wei Lin, Yifan Zhu, and Anh Tuan Luu. 2024. ChatKBQA: A generate-then-retrieve framework for knowledge base question answering with fine-tuned large language models. In Findings of the Association for Computational Linguistics ACL 2024, pages 2039–2056, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. https://github.com/huggingface/peft.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. In Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pages 1400–1409, Austin, Texas. Association for Computational Linguistics.

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 2022, pages 1535–1546, Seattle, United States. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, and et al. 2023. GPT-4 Technical Report. arXiv e-prints, arXiv:2303.08774.

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

Jorge Pérez, Marcelo Arenas, and Claudio Gutierrez. 2009. Semantics and complexity of sparql. ACM Trans. Database Syst., 34(3).

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. J. Mach. Learn. Res., 21(1).

Stephen Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. Foundations and Trendső in Information Retrieval, 3(4):333–389.

Apoorv Saxena, Aditay Tripathi, and Partha Talukdar. 2020. Improving multi-hop question answering over knowledge graphs using knowledge base embeddings. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 4498–4507, Online. Association for Computational Linguistics.

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, pages 4149–4158, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yiheng Shu, Zhiwei Yu, Yuhan Li, Börje Karlsson, Tingting Ma, Yuzhong Qu, and Chin-Yew Lin. 2022. TIARA: Multi-grained retrieval for robust question answering over large knowledge base. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 8108–8121, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), pages 2380–2390, Hong Kong, China. Association for Computational Linguistics.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel Ni, Heung-Yeung Shum, and Jian Guo. 2024. Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph. In The Twelfth International Conference on Learning Representations.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. In Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 641–651, New Orleans, Louisiana. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open Foundation and Fine-Tuned Chat Models. arXiv e-prints, arXiv:2307.09288.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. Commun. ACM, 57(10):7885.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I. Wang, Victor Zhong, Bailin Wang, Chengzu Li, Connor Boyle, Ansong Ni, Ziyu Yao, Dragomir Radev, Caiming Xiong, Lingpeng Kong, Rui Zhang, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2022. UnifiedSKG: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. In Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 602–631, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

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 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 6032–6043, Dublin, Ireland. Association for Computational Linguistics.

Wen-tau Yih, Ming-Wei Chang, Xiaodong He, and Jianfeng Gao. 2015. Semantic parsing via staged query graph generation: Question answering with knowledge base. In Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 1321–1331, Beijing, China. Association for Computational Linguistics.

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 (Volume 2: Short Papers), pages 201–206, Berlin, Germany. Association for Computational Linguistics.

Donghan Yu, Sheng Zhang, Patrick Ng, Henghui Zhu, Alexander Hanbo Li, Jun Wang, Yiqun Hu, William Yang Wang, Zhiguo Wang, and Bing Xiang. 2023. DecAF: Joint decoding of answers and logical forms for question answering over knowledge bases. In The Eleventh International Conference on Learning Representations.

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 (Volume 1: Long Papers), pages 5773–5784, Dublin, Ireland. Association for Computational Linguistics.

Lingxi Zhang, Jing Zhang, Yanling Wang, Shulin Cao, Xinmei Huang, Cuiping Li, Hong Chen, and Juanzi Li. 2023. FC-KBQA: A fine-to-coarse composition framework for knowledge base question answering. In Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1002–1017, Toronto, Canada. Association for Computational Linguistics.

Appendix

A Baseline Models

- **IR-based KBQA Methods:**

  KV-Men (Miller et al., 2016) use a key-value structured memory model to enhance document comprehension and question-answering by encoding facts and reasoning over them for accurate predictions.

  PullNet (Sun et al., 2019) presents a method that iteratively constructs a question-specific subgraph from knowledge bases and text for effective multi-hop reasoning in open-domain question answering.

  EmbedKGQA (Saxena et al., 2020) introduces a method that uses knowledge graph embeddings to improve multi-hop question answering, addressing knowledge graph sparsity.

  TransferNet (Shi et al., 2021) introduces a model that combines a transparent, attention-based approach with the ability to handle both label and text relations in a unified framework.

  Subgraph Retrieval (Zhang et al., 2022) introduces a method devising a trainable subgraph retriever(**SR**) decoupled from the reasoning process, which efficiently retrieves relevant subgraphs for question answering, enhancing performance by focusing on more relevant and smaller subgraphs and combining with subgraph-oriented reasoners.

- **SP-based KBQA Methods:**

  STAGG (Yih et al., 2016) presents a KBQA method using semantic parse labeling, showing improvements in query accuracy compared to relying solely on question-answer pairs.

  UHop (Chen et al., 2019) introduces a framework for unrestricted-hop relation extraction to handle queries requiring any number of relational hops in a knowledge graph, improving the capability to answer complex and indirect questions.

  Topic Units (Lan et al., 2019) utilizes a wide range of knowledge base units for question answering, employing a generation-and-scoring approach and reinforcement learning to enhance the identification and ranking of relevant topic units.

  QGG (Lan and Jiang, 2020) introduces a method that enhances complex question answering by generating flexible query graphs for multi-hop questions and integrating constraints early.

  UniK-QA (Oguz et al., 2022) proposes a framework that integrates structured, unstructured, and semi-structured knowledge sources, such as text, tables, lists, and knowledge bases, which flattens all data into text and applies a unified retriever-reader model.

  CBR-KBQA (Das et al., 2021) employs a casebased reasoning framework that retrieves similar cases (questions and logical forms) from a nonparametric memory, then reuses and revises these cases to generate logical forms for new questions, demonstrating its capability to handle complex questions and unseen relations without retraining.

  RnG-KBQA (Ye et al., 2022) introduces a framework that combines ranking and generation, using a rank-and-generate approach, where a ranker model identifies candidate logical forms and a generation model refines them.

  Program Transfer (Cao et al., 2022) proposes a novel two-stage parsing framework with an efficient ontology-guided pruning strategy for complex KBQA, which involves a sketch parser that translates questions into high-level program sketches and an argument parser that fills in detailed arguments.

  TIARA (Shu et al., 2022) introduces a novel method that enhances question answering over knowledge bases by using multi-grained retrieval, which improves the performance of pre-trained language models by focusing on the most relevant knowledge base contexts, including entities, logical forms, and schema items, and employs constrained decoding to control the output space, reducing generation errors and enhancing robustness in various generalization settings.

GMT-KBQA (Hu et al., 2022b) proposes a multi-task learning framework with a shared T5 encoder to improve question answering over knowledge bases by simultaneously learning entity disambiguation, relation classification, and logical form generation.

UnifiedSKG (Xie et al., 2022) unifies 21 structured knowledge grounding tasks into a text-to-text format, leveraging T5 models and multi-task learning to improve performance across diverse tasks and facilitate zero-shot and few-shot learning investigations.

DecAF (Yu et al., 2023) combines the generation of logical forms and direct answers, leveraging a sequence-to-sequence framework with retrieval from linearized knowledge bases.

FC-KBQA (Zhang et al., 2023) introduces a Fine-to-Coarse composition framework for question answering over knowledge bases, utilizing finegrained component detection, middle-grained component constraints, and coarse-grained component composition.

- **LLM-based KBQA Methods**

  StructGPT (Jiang et al., 2023a) enhances LLMs reasoning over structured data using an Iterative Reading-then-Reasoning (IRR) approach, which includes specialized interfaces for efficient data access, a novel invoking-linearization-generation procedure, and iterative reasoning to effectively utilize structured data in answering complex questions.

  Pangu (Gu et al., 2023) proposes a grounded language understanding framework that combines a symbolic agent and a neural language model, which allows for the incremental construction of valid plans and utilizes the language model to evaluate the plausibility of these plans.

  ToG (Sun et al., 2024) integrates LLMs with KGs for deep and responsible reasoning, using a beam search algorithm in KG/LLM reasoning, which allows the LLM to dynamically explore multiple reasoning paths in KG and make decisions accordingly, enhancing LLMs deep reasoning capabilities for knowledge-intensive tasks.

  ChatKBQA (Luo et al., 2024) is a novel question answering system that addresses the challenges of traditional knowledge-based question answering (KBQA) by adopting a generate-then-retrieve approach. It first generates a logical form and then retrieves entities and relations to answer questions, thereby overcoming the limitations of inefficient knowledge retrieval, error propagation from retrieval to semantic parsing, and the complexity of previous KBQA methods.

B Case Study

Here are two examples illustrating the benefits of incorporating optimal relation chains. As shown in Table 5, for both Question 1 and Question 2, the generated logical forms without optimal relation chains retained a consistent structure with their corresponding ground truth labels. This consistency suggests that the large language model possesses strong semantic parsing capabilities. For the simpler Question 1, the model generated a relationtypes of places of worshipthat, while close, did not entirely match the ground truth. In the more complex Question 2, the model correctly captured the backbone of the logical form but generated an incorrect relation, potentially leading to further errors downstream. However, after introducing optimal relation chains, there was a noticeable improvement; in both cases, the model generated correct logical forms. This demonstrates that incorporating optimal relation chains significantly enhances the model's semantic understanding.

Question 1	What does the religion who worships in Barcelona Cathedral call their God?
the optimal relation chain	religion. place_of_worship. religion
RGR-KBQA	(JOIN (R [religion , religion , deities]) (JOIN [religion , religion , places of worship] [Barcelona Cathedral]))
w/o SR	(JOIN (R [religion , religion , deities]) (JOIN [religion , religion , types of places of worship] [Barcelona Cathedral]))
Ground Truth Label	(JOIN (R [religion , religion , deities]) (JOIN [religion , religion , places of worship] [Barcelona Cathedral]))
Question 2	Who founded New York University, which held his governmental position from after 1795-03-04?
the optimal relation chain	organization. organization. founders
RGR-KBQA	(AND (GREATER THAN (JOIN [government , politician , government positions held] [government , government position held , from]) 1795-03-04) (JOIN (R [organization , organization , founders]) [New York University]))
w/o SR	(AND (GREATER THAN (JOIN [government , politician , government positions held] [government , government position held , from]) 1795-03-04) (JOIN (R [location , location , containedby]) [New York University]))
Ground Truth Label	(AND (GREATER THAN (JOIN [government , politician , government positions held] [government , government position held , from]) 1795-03-04) (JOIN (R [organization , organization , founders]) [New York University]))

Table 5: A case study exploring the role of optimal relationship chains in generating logical forms based on the CWQ dataset