From Phrases to Subgraphs: Fine-Grained Semantic Parsing for Knowledge Graph Question Answering

Yurun Song*, **Xiangqing Shen***, **Rui Xia**†
School of Computer Science and Engineering,
Nanjing University of Science and Technology, China
{yrsong, xqshen, rxia}@njust.edu.cn

Abstract

The recent emergence of large language models (LLMs) has brought new opportunities to knowledge graph question answering (KGQA), but also introduces challenges such as semantic misalignment and reasoning noise. Semantic parsing (SP), previously a mainstream approach for KGQA, enables precise graph pattern matching by mapping natural language queries to executable logical forms. However, it faces limitations in scalability and generalization, especially when dealing with complex, multi-hop reasoning tasks. In this work, we propose a **F**ine-**G**rained **S**emantic **P**arsing (**FGSP**) framework for KGQA. Our framework constructs a fine-grained mapping library via phrase-level segmentation of historical question-logical form pairs, and performs online retrieval and fusion of relevant subgraph fragments to answer complex queries. This fine-grained, compositional approach ensures tighter semantic alignment between questions and knowledge graph structures, enhancing both interpretability and adaptability to diverse query types. Experimental results on two KGQA benchmarks demonstrate the effectiveness of **FGSP**, with a notable 18.5% relative F1 performance improvement over the SOTA on the complex multi-hop CWQ dataset. Our code is available at https://github.com/NUSTM/From-Phrases-to-Subgraphs.

1 Introduction

Knowledge Graph Question Answering (KGQA) aims to leverage structured information stored in knowledge graphs such as Freebase (Bollacker et al., 2008) and Wikidata (Vrandecic and Krötzsch, 2014) to answer natural language questions. It has been widely applied in domains including Web search (Jang et al., 2017), medical consultation (Wu et al., 2024), and legal analysis (Cui et al., 2024). Knowledge graphs (KGs), which represent information via triple-based structures of entities and relations, provide a structured and semantically rich knowledge source. This inherent structure enables precise reasoning and facilitates multi-hop inference.

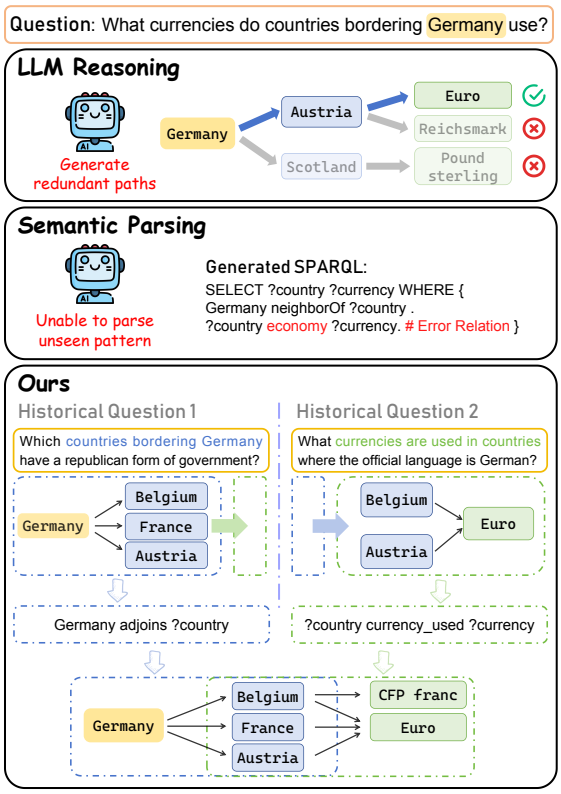

Figure 1: Comparison of three types of KGQA approaches.

In recent years, the emergence of large language models (LLMs) has significantly influenced the research paradigm of KGQA. Existing approaches that combine LLMs with KGs can be broadly categorized into two types: agent exploration frameworks (Sun et al., 2024; Zhu et al., 2024; Jiang et al., 2024) and path generation frameworks (Luo et al., 2024a; Sui et al., 2024). The former generates reasoning paths through continuous interaction be-

*Equal Contribution.
†Corresponding Author.

Findings of the Association for Computational Linguistics: ACL 2025, pages 5232–5246
July 27 - August 1, 2025

tween the LLMs and KGs to solve problems, while the latter has the LLMs generate reasoning paths first, which are then mapped to KGs instances to retrieve relevant information. Both approaches face the dual challenges of semantic gaps and paths: a) the unstructured nature of LLMs training data contrasts with the structured text of KGs, leading to hallucination; b) the LLMs' sequential reasoning paradigm generates redundant paths when handling multi-hop problems, causing noise accumulation in reasoning subgraphs. As illustrated in Figure 1, the model may generate irrelevant paths such as "Germany $\rightarrow$ Scotland $\rightarrow$ Pound sterling", thereby introducing spurious connections that hinder accurate reasoning. The root cause of these issues lies in the difference between natural language and graph-structured data, which motivates us to explore solutions that are more suited to the characteristics of graph structures.

Before the rise of LLM-based methods, semantic parsing (SP) was widely regarded as a mainstream approach for KGQA, known for its precision and efficiency (Berant and Liang, 2014; Dong and Lapata, 2016; Shaw et al., 2019). Semantic parsing transforms natural language questions into executable logical forms (LFs), which are then used to query knowledge graphs. This approach offers several distinct advantages over LLM-based methods. First, LFs align naturally with the entity-relationship structure of KGs, ensuring strict semantic alignment. This mitigates the hallucination problem caused by the unstructured training data of LLMs. Second, the syntax of LFs supports direct graph pattern matching, enabling precise and efficient reasoning. This avoids the redundant and noisy paths often produced by LLMs' sequential reasoning in multi-hop tasks. However, mapping between natural language and graph structures via LFs still poses challenges in scalability and generalization. As illustrated in Figure 1, when the model encounters an unseen semantic pattern—such as a question involving "currency"—it may incorrectly map it to a related but incorrect relation like "economy," leading to an erroneous LF and ultimately failing to retrieve the correct answer.

To address this issue, we propose a **F**ine-**G**rained **S**emantic **P**arsing (**FGSP**) framework for KGQA. FGSP enables the decomposition of complex questions into atomic semantic units, allowing for more precise alignment between natural language and KG structures. The framework operates in two stages. In the offline stage, a fine-grained mapping library is constructed by extracting and curating historical question-LF pairs, where questions are decomposed into sub-questions representing reasoning steps, and the LF is parsed according to the syntax structure. In the online reasoning stage, hierarchical reasoning is performed, where the question decomposition module breaks down complex queries into sub-questions; the phrase retrieval module searches for relevant phrases and instantiates LF fragments; the subgraph fusion module implements novel sequential fusion and combination fusion rules to construct a complete informational subgraph. Through a fine-grained semantic alignment mechanism, the framework achieves hierarchical analysis and compositional resolution of complex questions, enhancing the flexibility of problem-solving and ultimately improving the accuracy of complex semantic parsing by precisely matching atomic semantic units. As shown in Figure 1, fragments such as "countries bordering Germany" and "currencies used in countries" can be extracted from historical questions. Merging their corresponding subgraphs yields a complete structure for answering the current question.

Experiments on the WebQSP (Yih et al., 2016) and CWQ (Talmor and Berant, 2018) benchmark datasets demonstrate that our approach achieves strong performance, with a notable 18.5% relative F1 score improvement on the complex multi-hop CWQ dataset. This demonstrates the architectural advantages of our framework in handling multi-hop reasoning tasks and effectively reducing the retrieval of irrelevant information. Ablation studies further validate that the framework maintains robust performance across various base models and retrieval mechanisms.

2 Preliminary

In this section, we first introduce the definition of KG, KGQA and SP.

Knowledge Graph (KG) is a directed graph $\mathcal{G} = (\mathcal{E}, \mathcal{R}, \mathcal{T})$, where $\mathcal{E}$ is the set of entities, $\mathcal{R}$ is the set of relations, and $\mathcal{T} \subseteq \mathcal{E} \times \mathcal{R} \times \mathcal{E}$ is the set of triples. Each triple $(h, r, t) \in \mathcal{T}$ encodes a factual assertion, where $h, t \in \mathcal{E}$ are the heads and tail entities, and $r \in \mathcal{R}$ is the relation linking them.

Knowledge Graph Question Answering (KGQA) is a multi-hop reasoning task based on a knowledge graph (KG). Its goal is to translate a natural language question $q \in \mathcal{Q}$ to an answer $a \in \mathcal{A}$, where $\mathcal{A} \subseteq \mathcal{E}$. Given a question q , the model

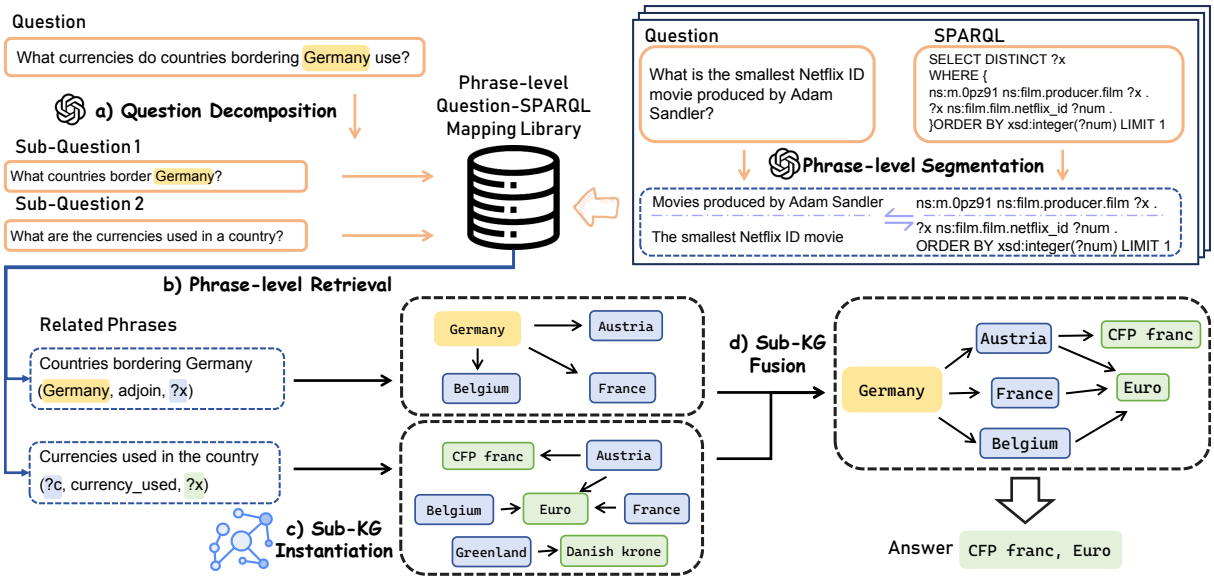

Figure 2: The overall framework of **FGSP**. Our framework is primarily divided into two stages. The offline stage (top-right) involves constructing a phrase-level question-SPARQL mapping library. In the online phase, the question is decomposed into sub-questions. Then the relevant phrases are retrieved and instantiated into sub-graphs, and finally the complete information sub-graph is constructed through sub-graph fusion.

extracts a subgraph from the KG $\mathcal{G}$ that contains the path-relevant evidence to infer the answer a .

Semantic Parsing (SP) transforms natural language queries $q \in \mathcal{Q}$ into logical forms $z \in \mathcal{Z}$, where $\mathcal{Z}$ denotes a space of formal representations such as lambda-DCS, SPARQL, or s-expressions. In the context of KGQA, the logical form z is a structured query executable on a knowledge graph $\mathcal{G}$ to retrieve the answer a.

3 Methodology

This section outlines the framework, which adopts SPARQL as the specific form of LF, and comprises three components: (1) the phrase-level question-SPARQL mapping library construction module, which analyzes questions and SPARQL to build a mapping library; (2) the phrase-level retrieval module, which retrieves relevant phrase pairs from the library based on sub-questions generated by LLMs; (3) the subgraph fusion module, which integrates retrieved SPARQL phrases into global subgraphs to derive accurate answers. Figure 2 presents an overview of the framework.

3.1 Phrase-level Question-SPARQL Mapping Library Construction

In the offline phase, we will construct a mapping library, with the phrases contained within it constituting the basis for phrase-level retrieval in the online phase. This stage consists of two key steps: (1) rule-based preprocessing of SPARQL queries, and (2) generation of context-aware phrase-level segmentation using LLMs.

The direct implementation of LLM-driven phrase-level segmentation introduces two key challenges: (1) over-aggregation, where a single phrase contains multiple reasoning steps, and (2) over-fragmentation, which leads to phrase disintegration and compromises semantic integrity. To address this, we design a rule-based SPARQL parser for preprocessing. Initially, we identify the Basic Graph Patterns (BGPs) representing the core semantics of SPARQL as primary candidates for reasoning paths. Next, BGP components with endpoints that are non-entity nodes are merged to form long phrases with complete semantics. These phrases maintain structural consistency, enabling flexible assembly for semantic extension during multi-hop reasoning, while also ensuring that each phrase serves as the minimal reasoning unit, preserving the robustness of atomic reasoning.

Building on preprocessed SPARQL fragments, we construct prompts that include the preprocessed phrases, the original SPARQL query, and the question. These prompts guide the LLM to establish semantic alignment through bidirectional phrase alignment. This method differs fundamentally from traditional approaches that convert triples into natural language. For instance, given the triple *(Zazaki, iso_639_3_code, zza)*, traditional methods might

generate descriptions such as "Zazaki's ISO 639-3 code is zza" whereas our framework generates a sentence more suited to the question, such as "The abbreviation for the Zazaki language is zza". It is important to note that, while the preprocessing step extracts the primary reasoning paths, the LLM still performs semantic mapping for other SPARQL syntactic structures to ensure the full utilization of SPARQL. In this approach, the LLM does not need to fully comprehend the entire KG; instead, it precisely maps the reasoning process of problem-solving to the closed phrase set, alleviating semantic mapping biases caused by incomplete KG understanding.

Phrase-level Segmentation Prompt

Given a question, SPARQL, mid and its entity name, and the main reasoning path, split the question and SPARQL into pairs of clauses based on the given reasoning path. The parts that do not belong to the reasoning path also need to be split.

3.2 Question Decomposition and Phrase-level Retrieval

For classic multi-hop reasoning tasks such as KGQA, inspired by previous studies (Khot et al., 2023; Wang et al., 2023), we use concise prompt templates to utilize LLMs to decompose a multi-hop question into sub-questions. As shown in Figure 2 a, the question "What currencies do countries bordering Germany use?" is decomposed into two sub-questions: "What countries border Germany?" and "What are the currencies used in a country?". During this process, the LLM is not exposed to KG data.

For each sub-question, we retrieve relevant NL phrases from the phrase-level question-SPARQL mapping library by computing embedding cosine similarity, selecting the top-k candidates, and converting them into corresponding SPARQL phrases. As shown in Figure 2 b, "Countries bordering Germany" and "Currencies used in the country" are retrieved as related clauses.

Question Decomposition Prompt

Decompose the given question into sub-question. There should be no nesting between sub-questions.

3.3 Sub-Knowledge Graph Instantiation and Fusion

To establish a connection between the retrieved SPARQL phrases and the final question answer, this module focuses on converting the retrieved

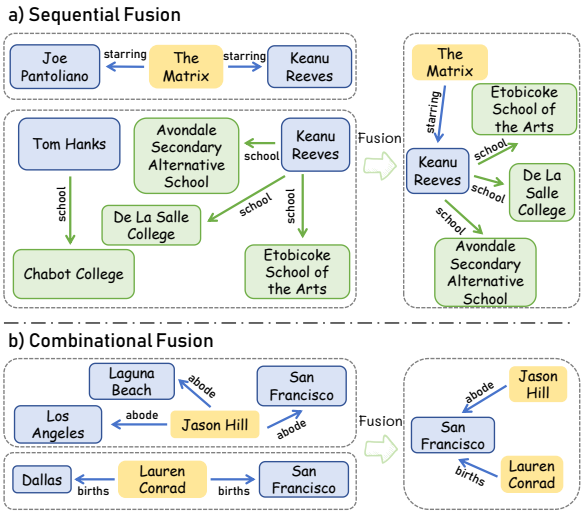

Figure 3: Two fusion paradigms, sequential (top) and combinatorial (bottom)

phrases into a coherent subgraph that accurately represents the original question.

Before subgraph fusion, the first step is to instantiate the retrieved SPARQL phrases into question-relevant subgraphs. SPARQL fragments capture local subgraph information within a KG. For example, the SPARQL phrase "*?country location.country.currency_used Euro*" corresponds to a subgraph containing all countries that use Euro as their currency. The instantiation of the subgraph is achieved by performing entity replacement on the entities within the retrieved SPARQL phrases.

A rule-based approach is then employed to fuse the subgraphs, transforming them into an integrated subgraph that represents the query's semantic content. This process builds on two classic reasoning paradigms in KGQA: (a) sequential reasoning $(A \rightarrow B \rightarrow C)$ and (b) compositional reasoning $(A + B \rightarrow C)$. Specifically, sequential reasoning extends the semantics of a preceding subgraph with the information from subsequent subgraphs, forming a reasoning path that eventually leads to the answer. In contrast, compositional reasoning synthesizes multi-subgraph constraints to generate a subgraph that satisfies multiple constraints simultaneously, as shown in Figure 3. The rule-based scheme designed based on these two reasoning modes systematically integrates local subgraphs into a unified subgraph that reflects the semantics of the original query, from which the final answer is derived, enabling trustworthy inference on the knowledge graph.

Type	Methods	WebQSP		CWQ	
		F1	Hit	F1	Hit
LLM-only	Llama3.1-8b (Meta, 2024)	34.8	55.5	22.4	28.1
	Qwen2-7B (Qwen et al., 2024)	35.5	50.8	21.6	25.3
	ChatGPT (OpenAI, 2022)	59.3	67.4	43.2	47.5
	GPT-4 (OpenAI, 2023)	62.3	73.2	49.9	55.6
Inference-based	StructGPT (Jiang et al., 2023a)	63.7	72.6	49.6	54.3
	Readi w/GPT-4 (Cheng et al., 2024)	-	78.7	-	67.0
	ToG w/GPT-4 (Sun et al., 2024)	-	82.6	-	67.6
	KG-CoT (Zhao et al., 2024)	-	84.9	-	62.3
Training-based	NSM (He et al., 2021)	62.8	68.7	42.4	47.6
	TransferNet (Shi et al., 2021)	-	71.4	-	48.6
	SR+NSM w E2E (Zhang et al., 2022)	64.1	69.5	46.3	49.3
	UniKGQA (Jiang et al., 2023b)	70.2	75.1	48.0	50.7
	DECAF (Yu et al., 2023)	**78.8**	82.1	-	70.4
	RoG (Luo et al., 2024a)	70.8	85.7	56.2	62.6
	Ours	73.3	**88.4**	**66.6**	**91.6**

Table 1: QA performance (F1 and Hit) of **FGSP** on WebQSP and CWQ datasets. Bold fonts indicate the best performance. The results of lama3.1-8b and Qwen2-7B are from Luo et al. (2024b).The results of ChatGPT and GPT-4 are from Jiang et al. (2024). others are from the origin paper

4 Experiment

4.1 Datasets

We use two classic benchmarks, WebQuestionSP (WebQSP) (Yih et al., 2016) and Complex WebQuestions (CWQ) (Talmor and Berant, 2018), to evaluate the effectiveness of our proposed framework. Freebase (Bollacker et al., 2008) is the underlying knowledge graph of these two datasets, containing about 88 million entities, 20,000 relation types, and 126 million RDF triples. The details of datasets are provided in the appendix A

4.2 Baselines

We compared our framework with 14 baseline models, categorized into three groups: 1) LLM-only, where the model relies solely on the LLM to answer questions without the use of KGs; 2) inference-based, where the language model and KGs interact to enable reasoning; and 3) training-based, where the language model is fine-tuned using a training dataset. The details of these approaches are provided in the appendix B.

4.3 Evaluation Metrics

Consistent with previous studies (Tan et al., 2023), we use Hit and F1 as our evaluation metrics. Hit verifies whether at least one correct answer appears in the generated predictions, while F1 measures the overall coverage of all correct answers by balancing precision and recall.

4.4 Implementations

For constructing the phrase-level question-SPARQL mapping library and question decomposition, we used GPT4o (OpenAI, 2024) as the base model, and the temperature was set to 0.4. BGE-m3 (Chen et al., 2024) was used as the embedding model in phrase retrieval. For the retrieved similar phrases, we used Top-5 as the subsequent subgraph fusion. For the baseline results in the table, we directly used the results reported in the corresponding paper for comparison. The LLM-only method used zero-shot prompts.

4.5 Main results

Table 1 presents the comparative experimental results of the proposed framework on two standard KGQA datasets. Compared to existing approaches that integrate LLMs with KGs, our framework demonstrates significant advantages in overall performance, which validates the method of deep integration of natural language and graph structures through semantic parsing. This approach effectively improves the efficiency and accuracy of information extraction from knowledge graphs. Specifically, in comparison to the LLM-only baseline models, our method achieves an F1 score improvement of 20.8% on the CWQ dataset, respectively, demonstrating that the structured knowledge retrieval mechanism can effectively address the

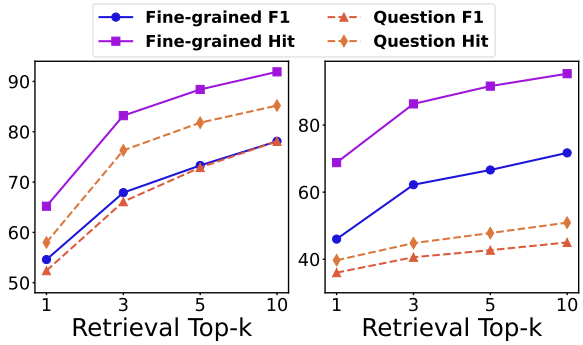

Figure 4: Comparison of fine-grained retrieval and question retrieval on WebQSP (left) and CWQ (right). Performance is shown for varying retrieval Top-k phrases.

knowledge gaps of LLMs. Notably, on the more complex CWQ dataset, our method further surpasses the current state-of-the-art approach, RoG, by 18.5% in F1 score. This improvement is primarily attributed to the technical advantages of the standardized SPARQL query language, unlike traditional methods that rely on autonomous reasoning path generation by the model, the structured query mechanism ensures higher accuracy through a standardized information extraction process. It is particularly worth noting that this substantial improvement empirically demonstrates the flexibility and scalability advantages of our framework, which are realized through fine-grained semantic decomposition. Additionally, our approach adopts a decoupled architectural design, enabling the LLM to operate independently of the KG during question reasoning and sub-question decomposition.

5 Analysis

In this section, we further analyze our framework by addressing the following six questions:

- **Q1**: How does the top-k selection in the retrieval stage affect the performance?
- **Q2**: How does fine-grained retrieval mapping affect the performance relative to the complete question retrieval?
- **Q3**: How does the scale of phrases affect the performance?
- **Q4**: How do different models for phrase-level segmentation affect the performance?
- **Q5**: How do different models for question decomposition affect the performance?
- **Q6**: How do different retrieval methods affect the performance?

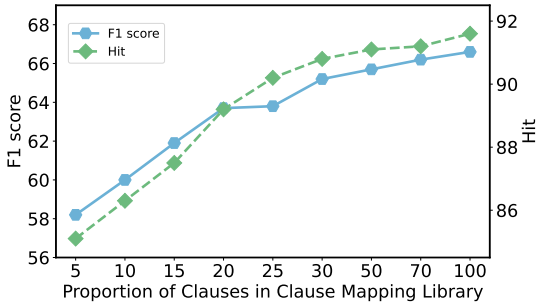

Figure 5: The F1 scores (Left) and Hit (Right) of **FGSP** on the test set of CWQ with a various amount of phrases.

5.1 Impact of Top-k and Fine-grained (Q1, Q2)

This section validates the necessity and effectiveness of the fine-grained framework through comparative experiments, focusing on the performance differences between the proposed method and question retrieval-based approaches under various Top-k configurations. The baseline method based on question retrieval uses all questions from the training set as retrieval sources. Its implementation consists of two key steps: (1) replacing the entities in SPARQL with mentioned entities; (2) executing the knowledge graph query. As shown in Figure 4, the experimental results demonstrate that our method outperforms the baseline across all Top-k values, confirming the effectiveness of the fine-grained framework. Notably, on multi-hop complex questions in the CWQ dataset, our method shows a more significant performance gain, with an average F1 improvement of 49.1%. This is attributed to the modular nature of fine-grained phrases compared to the original queries. Through the flexible phrase composition, the system can flexibly retrieve the knowledge fragments required for multi-hop reasoning. Further analysis reveals that fine-grained retrieval is more sensitive to the Top-k parameter. Specifically, the growth rate of F1 in the range of k=1 to k=10 in the CWQ dataset is 55.9%, while the baseline method experiences a relatively limited performance improvement, with only a 25% increase in the same range. This phenomenon suggests that the fine-grained phrases can better leverage an expanded search space.

5.2 Impact of Phrase Count (Q3)

This section systematically explores the impact mechanism of phrase size on the model's robustness. To simulate the data availability differences in real-world scenarios, The experimental setup

Models	Parameter	CWQ	
		F1	Hit
GPT4o	-	66.6	91.6
gpt4o-mini	-	66.3	91.9
Qwen2.5	14B	66.4	91.8
	7B	66.3	91.0
	3B	63.9	86.9
	1.5B	54.1	77.7
LLAMA 3.1	8B	63.7	89.5
LLAMA 3.2	3B	62.0	87.8
	1B	52.0	77.7

Table 2: Performance of different LLMs on phrase-level segmentation.

Models	Parameter	CWQ	
		F1	Hit
GPT4o	-	66.6	91.6
GPT4o-mini	-	66.2	91.7
Qwen2.5	14B	65.2	89.8
	7B	64.6	89.0
	3B	63.4	88.4
	1.5B	60.7	87.2
	0.5B	58.1	84.3
LLAMA 3.1	8B	64.7	90.1
LLAMA 3.2	3B	62.0	87.8
	1B	52.0	77.7
no deco	-	57.5	80.9

Table 3: Performance of different models in the question decomposition. The no-deco means don't decompose the question.

constructs a progressive data size scenario ranging from sparse phrases (5%) to complete phrases (100%). Figure 5 presents a performance comparison curve across different phrase sizes, revealing two important patterns: first, model performance shows a significant positive correlation with the number of phrases, with accuracy increasing linearly in the 5%-30% range; second, once the number of phrases exceeds 30%, the improvement rate slows down, and as the phrase count approaches 100%, marginal gains are observed, indicating a performance saturation threshold. It is worth noting that under extreme data-limited conditions (e.g., 5% phrases), the framework still exhibits excellent robustness: key metrics, F1 and Hit, remain in the range of 58.2%-66.6% and 85.1%-91.6%. This phenomenon validates the dual advantages of the framework design: scalability via performance improvements driven by incremental data, and strong adaptability under low-resource environments. The experimental results empirically demonstrate that the framework can maintain stable performance in resource-constrained scenarios, which is of significant practical value for real-world applications.

5.3 Impact of Phrase-level Segmentation (Q4)

We systematically evaluated the impact of various open-source and closed-source LLMs on the framework's performance in the phrase-level segmentation. In the experimental design, SPARQL statements preprocessed by rules were consistently input into different LLMs for fine-grained phrase decomposition. As shown in Table 2, GPT-4o demonstrated the best performance in terms of F1 score, while GPT-4o-mini outperformed others in the Hit metric. Notably, open-source models also exhibited performance comparable to that of closed-source models: Qwen-14B's decomposition results were closest to GPT-4o, and LLaMA-8B reached 95.6% of GPT-4o's performance, which validates the compatibility of this framework with both closed-source (GPT-4o) and open-source (Qwen, LLaMA) models. Further analysis revealed a sharp performance decline when the model's parameter size dropped below 1.5B. Diagnosing the outputs of smaller models, we found that, even with the aid of rule preprocessing, low-parameter models still struggled to accurately capture fine-grained semantic boundaries, frequently resulting in errors of over-aggregation and over-fragmentation. This issue is the key factor limiting their performance.

5.4 Impact of Question Decomposition (Q5)

We investigated the impact of different question decomposition models on the phrase-level retrieval stage. Four approaches were compared: GPT-4o, Qwen, LLaMA, and a baseline without decomposition strategy. All experiments were conducted under identical conditions. The same semantic embedding model and Top-5 retrieval parameters were maintained throughout the comparison. As shown in Table 3, when using LLMs for question decomposition, our framework exhibited F1 scores consistently in the 52.0-66.6 range. Remarkably, even 3B-parameter models achieved 93-95% relative performance compared to GPT-4o, which provides strong evidence for the robustness of this framework in the question decomposition stage. Ablation experiments showed that when the question decomposition module was removed, Hit and F1 scores decreased by 9.2% and 9.1%, respectively, compared to GPT-4o's decomposition approach, demonstrating the critical role of the ques-

Top-K	Methods	WebQSP		CWQ	
		F1	Hit	F1	Hit
Top-10	BM25	24.6	35.9	25.3	47.2
	Sentence-BERT	73.9	90.2	70.3	94.8
	conan	66.6	83.0	67.4	92.7
	BGE-m3	78.1	91.9	71.7	95.3
Top-5	BM25	16.5	24.4	17.0	33.2
	Sentence-BERT	63.9	82.4	64.9	90.7
	conan	57.9	74.6	62.0	88.4
	BGE-m3	73.3	88.4	66.6	91.6

Table 4: Performance of different retrieval methods.

tion decomposition module within this framework.

5.5 Impact of Retrieval Methods (Q6)

In phrase-level retrieval method comparison experiments, we systematically evaluated the performance differences between BM25 (Robertson and Walker, 1994) and other dense retrieval models such as BGE-M3 (Chen et al., 2024) , Sentence-BERT (Reimers and Gurevych, 2019), and Conan (Li et al., 2024). As shown in Table 4, vector-based retrieval methods exhibited slight performance variations between different embedding models: on the CWQ dataset, the best-performing BGE-M3 demonstrated a modest improvement of 6.3%-7.4% in F1 and a 2.8%-3.6% improvement in Hit over Conan, indicating that the choice of embedding model has a limited impact on this framework. The BM25 method significantly degraded system performance. This can be attributed to two interrelated factors. First, entities in historical questions exhibit heterogeneous characteristics, where different questions involve substantially distinct entities. Second, the phrase structure introduces additional complexity - specifically, the sparsity of word distributions in short phrases creates modeling challenges. Term-frequency-based methods particularly struggle to establish effective probability distribution models under these conditions of lexical sparsity.

6 Related Work

Currently, research on KGQA can be broadly categorized into two main approaches: one leverages LLMs for KGQA tasks, while the other is based on semantic parsing methods.

6.1 LLMs for KGQA

KGs, as structured knowledge bases, can effectively supplement the factual knowledge of LLMs and reduce hallucinations (Pan et al., 2024). Existing KG-enhanced approaches are primarily divided into two categories: inference-based and training-based methods. Inference-based models, such as StructGPT (Jiang et al., 2023a), ToG (Sun et al., 2024), and KnowAgent (Zhu et al., 2024), facilitate multi-round interactions between LLMs and KGs, treating LLMs as agents. Although this approach effectively leverages LLMs' reasoning abilities, the lack of sufficient understanding of relationships or entities within the KG by the LLM further amplifies hallucinations, leading to incorrect reasoning results. Training-based methods aim to integrate the knowledge of KG into LLMs. RoG (Luo et al., 2024a) samples the reasoning paths from the graph, injects relational information from the KG into the LLM, and generates reasoning paths to retrieve useful information from the KG for inference. GNN-RAG (Mavromatis and Karypis, 2024) uses a lightweight GNN to effectively extract information from the KG to assist LLM reasoning. Training-based methods, in order to ensure coverage of correct reasoning paths, tend to generate additional noise paths, which may affect subsequent LLM reasoning.

6.2 Semantic Parsing

In the KGQA task, semantic parsing (SP) aims to convert natural language questions into structured logical forms, such as lambda-DCS (Liang, 2013), SPARQL queries (Das et al., 2021), graph queries (Yih et al., 2015; Lan and Jiang, 2020), and s-expressions (Gu et al., 2021). Early methods employed grammar-based parsers (Mitra et al., 2022; Sun et al., 2020; Liang et al., 2017), whereas recent research has focused on leveraging pre-trained language models to enhance semantic parsing (Scholak et al., 2021; Zhang et al., 2019). SPARQA (Sun et al., 2020) introduces skeleton grammar to represent the high-level structure of complex questions and incorporates BERT to improve the accuracy of semantic dependency relations. SR (Zhang et al., 2022) employs a trainable subgraph retriever that utilizes a dual-encoder model to expand reasoning paths and dynamically determine termination conditions for expansion. RNG-KBQA (Ye et al., 2022) combines a contrastive ranker with a generative model, leveraging high-confidence candidates to optimize the final composition of logical forms. These methods optimize latent space representations to improve question generalization. In contrast, our approach tackles scalability challenges in semantic parsing through an explicit phrase alignment and retrieval mechanism.

7 Conclusion

In this paper, we propose a two-stage fine-grained phrase alignment framework to address the issues of semantic gaps and redundant paths that arise in the integration of LLMs with KGs. By leveraging SPARQL, we establish a connection between natural language and graph structures, facilitating the precise and efficient retrieval of relevant information from the KG. Experimental results on two standard KGQA benchmarks validate the effectiveness of our approach, demonstrating that the application of semantic parsing effectively bridges the semantic gap and reasoning discrepancies between LLMs and KGs, leading to faithful and reliable inference.

Limitations

Although we have demonstrated the effectiveness and stability of our framework across various experimental settings, several limitations remain. First, our approach depends on Freebase, which may limit its applicability to knowledge bases with different structures and entity distributions, such as Wikidata and DBpedia. Future work should explore its generalization capability across different knowledge bases. Second, we employ rule-based substitution for retrieved SPARQL phrases, a process that could be improved in terms of accuracy and adaptability through generative models. Lastly, we use SPARQL as the LF for fine-grained alignment; extending LF to S-expression and lambda-DCS could further validate the framework's generality and compatibility.

Ethics Statement

This work does not pose any ethical issues. All the data and models used in this paper are publicly available and are used under following licenses: Creative Commons BY 4.0 License, MIT License, Apache license 2.0, Llama 3.1 Community License Agreement, Llama 3.2 Community License Agreement.

Acknowledgments

This work was supported by the Natural Science Foundation of China (No. 62476134).

References

Jonathan Berant and Percy Liang. 2014. Semantic parsing via paraphrasing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 1: Long Papers*, pages 1415–1425. The Association for Computer Linguistics.

Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250. ACM.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 2318–2335. Association for Computational Linguistics.

Sitao Cheng, Ziyuan Zhuang, Yong Xu, Fangkai Yang, Chaoyun Zhang, Xiaoting Qin, Xiang Huang, Ling Chen, Qingwei Lin, Dongmei Zhang, Saravan Rajmohan, and Qi Zhang. 2024. Call me when necessary: Llms can efficiently and faithfully reason over structured environments. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 4275–4295. Association for Computational Linguistics.

Jiaxi Cui, Munan Ning, Zongjian Li, Bohua Chen, Yang Yan, Hao Li, Bin Ling, Yonghong Tian, and Li Yuan. 2024. Chatlaw: A multi-agent collaborative legal assistant with knowledge graph enhanced mixture-of-experts large language model. *Preprint*, arXiv:2306.16092.

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, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 9594–9611. Association for Computational Linguistics.

Li Dong and Mirella Lapata. 2016. Language to logical form with neural attention. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics.

Yu Gu, Sue Kase, Michelle Vanni, Brian M. Sadler, Percy Liang, Xifeng Yan, and Yu Su. 2021. Beyond I.I.D.: three levels of generalization for question answering on knowledge bases. In *WWW '21: The Web Conference 2021, Virtual Event / Ljubljana, Slovenia, April 19-23, 2021*, pages 3477–3488. ACM / IW3C2.

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 *WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021*, pages 553–561. ACM.

Heewon Jang, Yeongtaek Oh, Seunghee Jin, Haemin Jung, Hyesoo Kong, Dokyung Lee, Dongkyu Jeon, and Wooju Kim. 2017. Kbqa: Constructing structured query graph from keyword query for semantic search. In *Proceedings of the International Conference on Electronic Commerce*, pages 1–8.

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, EMNLP 2023, Singapore, December 6-10, 2023*, pages 9237–9251. Association for Computational Linguistics.

Jinhao Jiang, Kun Zhou, Wayne Xin Zhao, Yang Song, Chen Zhu, Hengshu Zhu, and Ji-Rong Wen. 2024. Kg-agent: An efficient autonomous agent framework for complex reasoning over knowledge graph. *CoRR*, abs/2402.11163.

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, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2023. Decomposed prompting: A modular approach for solving complex tasks. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

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, ACL 2020, Online, July 5-10, 2020*, pages 969–974. Association for Computational Linguistics.

Shiyu Li, Yang Tang, Shizhe Chen, and Xi Chen. 2024. Conan-embedding: General text embedding with more and better negative samples. *Preprint*, arXiv:2408.15710.

Chen Liang, Jonathan Berant, Quoc V. Le, Kenneth D. Forbus, and Ni Lao. 2017. Neural symbolic machines: Learning semantic parsers on freebase with weak supervision. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 23–33. Association for Computational Linguistics.

Percy Liang. 2013. Lambda dependency-based compositional semantics. *CoRR*, abs/1309.4408.

Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. 2024a. Reasoning on graphs: Faithful and interpretable large language model reasoning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Linhao Luo, Zicheng Zhao, Chen Gong, Gholamreza Haffari, and Shirui Pan. 2024b. Graph-constrained reasoning: Faithful reasoning on knowledge graphs with large language models. *CoRR*, abs/2410.13080.

Costas Mavromatis and George Karypis. 2024. GNN-RAG: graph neural retrieval for large language model reasoning. *CoRR*, abs/2405.20139.

Meta. 2024. Build the future of ai with meta llama 3.

Sayantan Mitra, Roshni R. Ramnani, and Shubhashis Sengupta. 2022. Constraint-based multi-hop question answering with knowledge graph. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies: Industry Track, NAACL 2022, Hybrid: Seattle, Washington, USA + Online, July 10-15, 2022*, pages 280–288. Association for Computational Linguistics.

OpenAI. 2022. Introducing chatgpt.

OpenAI. 2023. GPT-4 technical report. *CoRR*, abs/2303.08774.

OpenAI. 2024. Hello gpt-4o.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Stephen E. Robertson and Steve Walker. 1994. Some simple effective approximations to the 2-poisson

model for probabilistic weighted retrieval. In *Proceedings of the 17th Annual International ACM-SIGIR Conference on Research and Development in Information Retrieval. Dublin, Ireland, 3-6 July 1994 (Special Issue of the SIGIR Forum)*, pages 232–241. ACM/Springer.

Torsten Scholak, Nathan Schucher, and Dzmitry Bahdanau. 2021. PICARD: parsing incrementally for constrained auto-regressive decoding from language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 9895–9901. Association for Computational Linguistics.

Peter Shaw, Philip Massey, Angelica Chen, Francesco Piccinno, and Yasemin Altun. 2019. Generating logical forms from graph representations of text and entities. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 95–106. 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, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 4149–4158. Association for Computational Linguistics.

Yuan Sui, Yufei He, Nian Liu, Xiaoxin He, Kun Wang, and Bryan Hooi. 2024. Fidelis: Faithful reasoning in large language model for knowledge graph question answering. *CoRR*, abs/2405.13873.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel M. 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, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yawei Sun, Lingling Zhang, Gong Cheng, and Yuzhong Qu. 2020. SPARQA: skeleton-based semantic parsing for complex questions over knowledge bases. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8952–8959. AAAI Press.

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.

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. In *The Semantic Web - ISWC 2023 - 22nd International Semantic Web Conference, Athens, Greece, November 6-10, 2023, Proceedings, Part I*, volume 14265 of *Lecture Notes in Computer Science*, pages 348–367. Springer.

Denny Vrandecic and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85.

Lei Wang, Wanyu Xu, Yihuai Lan, Zhiqiang Hu, Yunshi Lan, Roy Ka-Wei Lee, and Ee-Peng Lim. 2023. Plan-and-solve prompting: Improving zero-shot chain-of-thought reasoning by large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 2609–2634. Association for Computational Linguistics.

Junde Wu, Jiayuan Zhu, and Yunli Qi. 2024. Medical graph RAG: towards safe medical large language model via graph retrieval-augmented generation. *CoRR*, abs/2408.04187.

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), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 6032–6043. 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 of the Asian Federation of Natural Language Processing, ACL 2015, July 26-31, 2015, Beijing, China, Volume 1: Long Papers*, pages 1321–1331. The Association for Computer Linguistics.

Wen-tau Yih, Matthew Richardson, Christopher 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, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer 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, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Haoyu Zhang, Jingjing Cai, Jianjun Xu, and Ji Wang. 2019. Complex question decomposition for semantic parsing. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 4477–4486. Association for Computational Linguistics.

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), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 5773–5784. Association for Computational Linguistics.

Ruilin Zhao, Feng Zhao, Long Wang, Xianzhi Wang, and Guandong Xu. 2024. Kg-cot: Chain-of-thought prompting of large language models over knowledge graphs for knowledge-aware question answering. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI 2024, Jeju, South Korea, August 3-9, 2024*, pages 6642–6650. ijcai.org.

Yuqi Zhu, Shuofei Qiao, Yixin Ou, Shumin Deng, Ningyu Zhang, Shiwei Lyu, Yue Shen, Lei Liang, Jinjie Gu, and Huajun Chen. 2024. Knowagent: Knowledge-augmented planning for llm-based agents. *CoRR*, abs/2403.03101.

Datasets	#Train	#Test	Max #hop
WebQSP	2,826	1,628	2
CWQ	27,639	3,531	4

Table 5: Statistics of datasets.

Dataset	#Ans = 1	$2 \geq$ #Ans ≤ 4	$5 \geq$ #Ans ≤ 9	#Ans ≥ 10
WebQSP	51.2%	27.4%	8.3%	12.1%
CWQ	70.6%	19.4%	6%	4%

Table 6: Statistics of the number of answers for questions in WebQSP and CWQ.

Dataset	1 hop	2 hop	≥ 3 hop
WebQSP	65.49 %	34.51%	0.00%
CWQ	40.91 %	38.34%	20.75%

Table 7: Statistics of the question hops in WebQSP and CWQ.

A Datasets

We utilize two standard datasets, WebQSP and CWQ. To ensure fairness, we adopt the same training and test splits as in previous studies. The detailed statistics of the datasets are presented in Table 5. Both WebQSP and CWQ are based on Freebase. In our approach, we employ the complete Freebase as the knowledge graph for information retrieval.

B Baselines

We compared **FGSP** against 14 baselines, which can be categorized into three groups: (1) LLM-only methods, (2) inference-based methods, and (3) training-based methods. The specific details of these approaches are as follows.

B.1 LLM-only method

The results of lama3.1-8b (Meta, 2024) and Qwen2-7B (Qwen et al., 2024) are from Luo et al. (2024b).The results of ChatGPT (OpenAI, 2022) and GPT-4 (OpenAI, 2023) are from Jiang et al. (2024).

B.2 Inference-based methods

- StructGPT (Jiang et al., 2023a) enhances large language models' reasoning ability on structured data by integrating the iterative reading-reasoning (IRR) method with structured data interfaces.
- Readi (Cheng et al., 2024) enables efficient and faithful reasoning in structured environments by allowing large language models (LLMs) to generate and edit reasoning paths.
- ToG (Sun et al., 2024) tightly integrates large language models with knowledge graphs, leveraging graphs for deep reasoning and employing iterative beam search execution to improve reasoning capability and interpretability.
- KG-CoT (Zhao et al., 2024) enhances the knowledge reasoning ability of LLMs by incorporating small-scale stepwise graph reasoning models and leveraging knowledge graphs (KGs), thereby improving LLMs' performance on knowledge-intensive question-answering tasks without requiring fine-tuning.

B.3 Training-based methods

- NSM (He et al., 2021) proposes a multi-hop knowledge base question-answering method based on a teacher-student framework, improving reasoning ability by learning intermediate supervision signals and generating more reliable intermediate entity distributions through bidirectional reasoning mechanisms.
- GraftNet (Shi et al., 2021) is an efficient and transparent framework for multi-hop question answering that infers answers by propagating entity scores across entities, supporting the processing of both labeled and textual entity relations within a unified framework.
- SR+NSM (Zhang et al., 2022) decouples the retrieval and reasoning processes to enhance the performance of embedding-based KBQA models.
- UniKGQA (Jiang et al., 2023b) addresses multi-hop knowledge graph question answering by integrating semantic matching and information propagation modules while unifying retrieval and reasoning in model architecture and parameter learning.
- DECAF (Yu et al., 2023) improves the accuracy of knowledge base question-answering tasks by jointly generating logical forms and

Type	Split results	Remark
over-aggregation	["The character in Cars played by the actor who played Porco Rosso.", "?c ns:film.actor.dubbing_performances ?k . "] ?k ns:film.dubbing_performance.character ns:m.0nfpg4s . ?c ns:film.actor.film ?y ?y ns:film.performance.character ?x"]	No separation of questions and SPARQL at all
over-fragmentation	["What character", "character ?x"]	The split result is not a triple
	["Porco Rosso", "m.0nfpg4s"]	Split to only a single entity
	["in Cars", "?y ns:film.performance.film ns:m.03q0r1 ."]	The split result cannot support complete semantics

Table 8: Two kinds of mistakes in the phrase-level segmentation: a)over-encapsulation b)excessive fragmentation

Model	Parameter	Number	Average length	
			NL clauses	Question
GPT4o	-	43066	6.12	13.21
gpt4o-mini	-	42373	6.01	
Qwen2.5	14B	40310	6.30	
	7B	41330	5.96	
	3B	41395	6.24	
	1.5B	46943	6.72	
	0.5B	23393	7.13	
LLAMA 3.1	8B	44135	6.38	
LLAMA 3.2	3B	58886	8.63	
	1B	61623	6.53	

Table 9: Scale details of phrase-level question-SPARQL mapping libraries built by different models

direct answers, leveraging the strengths of both approaches while simplifying model adaptation across different datasets.

- ROG (Luo et al., 2024a) combines large language models (LLMs) and knowledge graphs (KGs), employing a plan-retrieve-reason framework to generate faithful and interpretable reasoning results.

C Error in Phrase-level Segmentation

In phrase-level segmentation, we identify two issues that arise when directly relying on LLMs for phrase-level segmentation: over-encapsulation and excessive fragmentation. These two issues are presented in detail in the table 8.

D Details of the Phrase-level Question-SPARQL Mapping Libraries

The details of the phrase-level question-SPARQL mapping libraries are presented in the table 9. The constructed library comprises approximately 40,000 phrase pairs, with an average natural language phrase length of around six words and an average original question length of 13.21 words.

E Prompt

We present the prompts used throughout our framework.

The prompt employed for the phrase pair decomposition process is shown in table 10. We adopt a few-shot approach to ensure that the phrase pairs generated by the LLM adhere to our predefined format. The prompt includes preprocessed Basic Graph Patterns (BGPs), the original SPARQL query, and the complete question. Notably, the natural language phrases derived from the question are not sub-questions but rather declarative statements.

For the online stage, the prompt used for LLM-based question decomposition is presented in table 10. To minimize constraints on the LLM's reasoning process, we refrain from designing overly complex prompts for question decomposition.

Prompt for Phrase-level Segmentation Given a question, a SPARQL query, mid and its entity name in the SPARQL statement, and the main reasoning path implied by SPARQL, split the question and SPARQL into pairs of clauses based on the given reasoning path. The parts that do not belong to the reasoning path also need to be split.
Examples
Few-shot Question
<*Question*>
SPARQL:
<*SPARQL*>
mention entity:
<*mention entity*>
inference chain:
<*inference chain*>
pairs:

Demonstration Example
Question: What is the national currency of the country where Bajan is spoken?
SPARQL:
PREFIX ns: <http://rdf.freebase.com/ns/>
SELECT DISTINCT ?x
WHERE {
FILTER (?x != ?c)
FILTER (!isLiteral(?x) OR lang(?x) = '' OR langMatches(lang(?x), 'en'))
?c ns:location.country.languages_spoken ns:m.03xx69 .
?c ns:location.country.currency_used ?x .
}
mention entity:
m.03xx69: Bajan Language
inference chain:
?c ns:location.country.languages_spoken ns:m.03xx69 .
?c ns:location.country.currency_used ?x .
pairs:
[["Countries that speak Barbadian", "?c ns:location.country.languages_spoken ns:m.03xx69 ."],
["The country's currency", "?c ns:location.country.currency_used ?x ."]]

Prompt for Question Decomposition
Decompose the given problem into subproblems. There should be no nesting between subproblems.
Examples:
<*Few-shot*>
Question
<*Question*>

Demonstration Example
Question: Which city of residence for Tom Hanks was the birthplace of Elon Musk?
Sub-questions: ["Which city did Tom Hanks live in?", "Where was Elon Musk born?"]

Table 10: Detailed prompts for modules of **FGSP**