Graph Memory-based Editing for Large Language Models

Anonymous ACL submission

Abstract

The information within Large Language Models (LLMs) quickly becomes outdated, prompting the development of various techniques to perform knowledge editing with new facts. However, existing knowledge editing methods often overlook the interconnected nature of facts, failing to account for the ripple effects caused by changing one piece of information. In our study, we present GMeLLo (**G**raph **Me**mory-based Editing for **L**arge **L**anguage **Mo**dels), a simple yet effective memory-based method that transitions the Multi-hop Question Answering for Knowledge Editing (MQuAKE) task into a Knowledge-based Question Answering (KBQA) framework. GMeLLo stores all relevant facts externally in a Knowledge Graph (KG) and directs the language model to engage in semantic parsing. This involves translating natural language questions into formal queries to extract information from the KG. Notably, our method eliminates the need to fine-tune LLMs, ensuring that edited facts do not corrupt other information. In our experimental findings, we noted a noteworthy enhancement of GMeLLo in comparison to state-of-the-art model editors on the MQuAKE benchmark—a dataset tailored for multi-hop question answering, particularly evident when editing multiple facts simultaneously.

1 Introduction

As the widespread deployment of Large Language Models (LLMs) continues, the imperative to maintain their knowledge accuracy and currency, without incurring extensive retraining costs, becomes increasingly evident (Sinitsin et al., 2020). Several approaches have been proposed in prior works to address this challenge, with some focusing on the incremental injection of new facts into language models (Rawat et al., 2020; De Cao et al., 2021; Meng et al., 2022; Mitchell et al., 2022a). Interestingly, certain methodologies in the literature diverge from the conventional path of updating model weights, opting instead for an innovative strategy involving the use of external memory to store the edits (Mitchell et al., 2022b; Zhong et al., 2023). As LLMs operate as black boxes, modifying one fact might inadvertently alter another, making it challenging to guarantee accurate revisions. In light of this challenge, opting for an external memory system, rather than directly editing the LLMs, emerges as a prudent choice. On a different note, even though information undergoes rapid evolution, the patterns of sentences—various ways to convey meaning—tend to change at a comparatively slower rate. LLMs, trained on extensive sentence corpora (Brown et al., 2020; Rae et al., 2022; Chowdhery et al., 2023), are anticipated to encapsulate a broad spectrum of commonly used sentence structures. Consequently, they serve as invaluable tools for analyzing complex relation chains within sentences.

This paper introduces GMeLLo, an innovative approach designed to synergize the strengths of LLMs and KG in addressing the multi-hop question answering task after knowledge editing (Zhong et al., 2023). An illustrative example is presented in Figure 1. Following an update regarding the information of the British Prime Minister, it becomes evident that the corresponding spouse information should also be modified.

Specifically, we utilize LLMs to analyze question sentences, extracting the underlying relation chain. Simultaneously, we employ the KG as an external memory to maintain up-to-date information, encompassing both the modified and unaltered facts. Ultimately, we translate the relation chain into a formal query using heuristic rules and search for information within the KG. Using LLMs for question analysis ensures coverage of diverse patterns, thanks to their extensive training on large datasets, enabling them to understand various representations of the same meaning. Once the correct relation chain is returned, using a formal query to interrogate the KG ensures precision. Through ex-

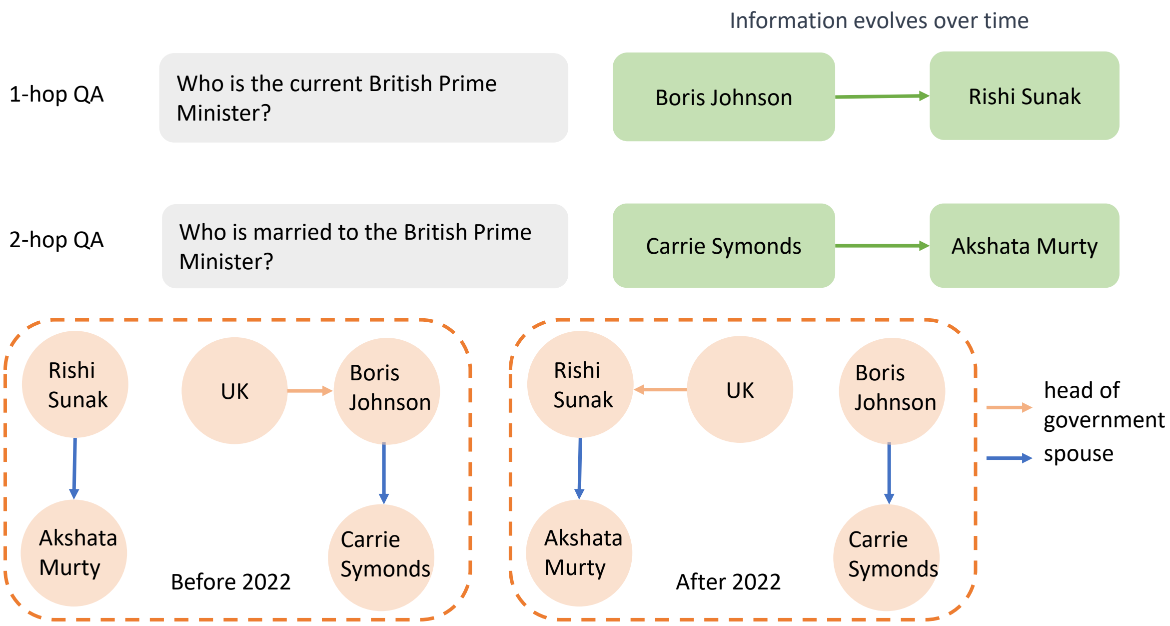

Figure 1: Dynamic nature of information: Changes over time may trigger subsequent modifications. For instance, a transition in the British Prime Minister, such as from Boris Johnson to Rishi Sunak, necessitates corresponding adjustments, like the change in the British Prime Minister's spouse.

perimentation, GMeLLo demonstrates significantly enhanced performance compared to current baseline models on the MQuAKE benchmark-multi-hop question answering dataset for knowledge editing, affirming its effectiveness.

2 Related Work

The primary focus of this paper is on knowledge editing for multi-hop question answering, with our predominant methodology being semantic parsing. Consequently, we delve into the related work within both research domains.

2.1 Knowledge Editing

As highlighted in Yao et al. (2023), two paradigms exist for editing LLMs: preserving model parameters and modifying model parameters. In the case of preserving model parameters, the introduction of additional parameters or external memory becomes necessary. The paradigm of additional parameters, as presented in (Dong et al., 2022; Hartvigsen et al., 2022; Huang et al., 2022), incorporates extra trainable parameters into the language model. These parameters are trained on a modified knowledge dataset, while the original model parameters remain static. On the other hand, memory-based models (Mitchell et al., 2022b; Zhong et al., 2023) explicitly store all edited examples in memory and employ a retriever to extract the most relevant edit facts for each new input, guiding the model in generating the edited output.

In the case of modifying model parameters, this can be further categorized into meta-learning or locate-and-edit approaches. Meta-learning methods, as discussed in (De Cao et al., 2021; Mitchell et al., 2022a), utilize a hyper network to learn the necessary adjustments for editing LLMs. The locate-then-edit paradigm, as demonstrated in (Dai et al., 2022; Meng et al., 2022, 2023; Li et al., 2023; Gupta et al., 2023), involves initially identifying parameters corresponding to specific knowledge and subsequently modifying them through direct updates to the target parameters.

While previous evaluation paradigms have primarily focused on validating the recall of edited facts, Zhong et al. (2023) proposed MQuAKE, a benchmark dataset comprising multi-hop questions. This dataset assesses whether edited models correctly answer questions where the response should change as a consequence of edited facts.

2.2 Knowledge-based Question Answering

Knowledge-based Question Answering (KBQA) (Cao et al., 2023) seeks to provide answers to natural language questions using a knowledge base as its primary information source. Recently, the advent of LLMs has spurred the development of LLM-based KBQA systems. For instance, KB-Coder (Nie et al., 2024) proposes a code-style in-context learning approach for KBQA, which transforms the unfamiliar logical form generation process into a more familiar code generation process for LLMs.

The disparity between the MQuAKE task and the KBQA task lies in: 1) MQuAKE does not provide a predefined knowledge base, necessitating the creation of one from scratch or the identification of a suitable existing knowledge base; and 2) Complex questions in KBQA entail multi-hop reasoning over the KB, constrained relations, and numerical operations, whereas MQuAKE questions primarily revolve around multi-hop reasoning (up to 4-hop). Consequently, in our study, we exploit LLMs to generate a relation chain instead of tasking them with generating a more intricate logical form. This approach obviates the need for extensive expertise, enabling even smaller LLMs like GPT-J-6B to effectively analyze linguistic patterns and extract relation chains.

3 GMeLLo: Graph Memory-based Editing for Large Language Models

In this section, we explore the intricacies of our innovative knowledge editing method, GMeLLo, leveraging the combined strengths of LLMs and KGs. Drawing inspiration from memory-based knowledge-editing approaches (Mitchell et al., 2022b; Zhong et al., 2023), GMeLLo preserves the foundational language model in a frozen state while storing all edits in an explicit memory. Figure 2 provides a visual representation of the GMeLLo framework.

3.1 Extracting the Relation Chain of a Question Sentence Using LLMs

Given the rapid pace of change in the world, LLMs' training data may become quickly outdated. Therefore, we recommend employing LLMs for sentence analysis rather than relying on them for direct answers. This approach is justified by the relatively slower evolution of patterns compared to the intricate details. In this paper, we employ LLMs to extract the relation chain from a sentence, encompassing the mentioned entity and relations with other unidentified entities. To mitigate varied representations of the same relation, we task LLMs with selecting a relation from a predefined list. Take a question sentence from the MQuAKE dataset as an example,

- Question: What is the capital of the country of citizenship of the child of the creator of Eeyore?
- Relation Chain: Eeyore->creator->?x->child->?y->country of citizenship->?z->capital->?m

The presented question necessitates a 4-hop reasoning process. With "Eeyore" as the known entity in focus, the journey to the final answer involves identifying its creator, moving on to the creator's child, obtaining the child's country of citizenship, and culminating with the retrieval of the country's capital. The relation chain encapsulates all essential information for arriving at the conclusive answer.

To ensure that LLMs comprehend the task of extracting the relation chain and generate output in a structured template, we employ in-context learning (Dong et al., 2023). This technique involves providing LLMs with a set of examples in the prompt, guiding them through the desired output format.

3.2 Utilizing KGs for Storing Correlated Facts to Enhance Multi-hop Reasoning

KGs play a pivotal role in enhancing the capabilities of LLMs by offering external knowledge for improved inference and interpretability, as demonstrated by recent studies (Pan et al., 2023; Rawte et al., 2023). Unlike conventional approaches that rely on question templates for each relation type (Petroni et al., 2019; Meng et al., 2022), and then store the updated information in an external memory as a list of separated sentence statements (Zhong et al., 2023), we represent information as a graph to preserve inherent connections.

In our approach, we consolidate all relevant information within a KG. Rather than constructing a new external memory specifically for updated data, we opt for a more efficient strategy—directly updating the existing KG. This not only simplifies the information storage process but also leverages the inherent connectivity within the graph, providing a more cohesive and context-rich representation of correlated facts.

Our mechanism offers an additional advantage by storing both updated and unchanged facts. This approach facilitates the identification of conflicts between facts. In contrast, if only updated facts are explicitly stored, detecting inconsistencies between updated facts and unchanged ones becomes challenging, as the latter are not explicitly recorded. We provide further details on this matter in Section 4.5.2.

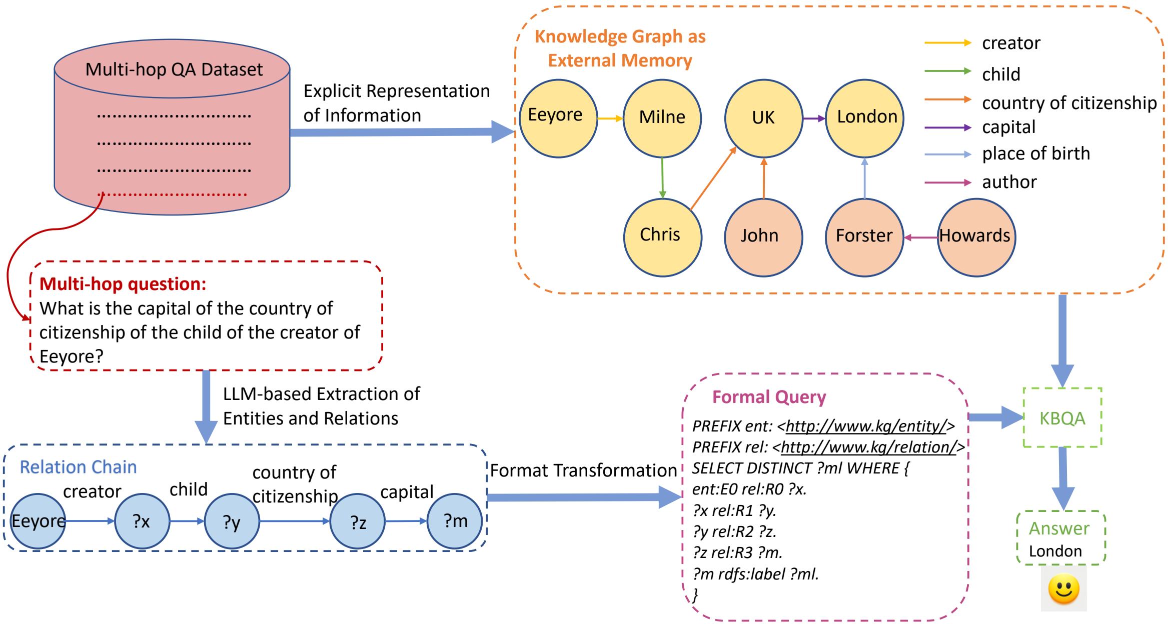

Figure 2: The illustration delineates our proposed method, GMeLLo. Commencing the process, we establish a KG either by extracting information from the QA dataset or by utilizing an existing KG as the foundational external memory. If there are updates to the information, we directly modify the KG. Simultaneously, we leverage LLMs to extract the primary relation chain from a given multi-hop question, capturing the known entity and its relationships with other unidentified entities. Following the acquisition of the relation chain, we transform it into a formal query format, such as SPARQL. Armed with the KG and the formal query, we employ Knowledge-based Question Answering (KBQA) (Lan et al., 2022) to deduce the final answer.

3.3 Converting the Relation Chain into a Formal Query for Retrieving Updated Information from KGs

Once the relation chain is obtained, the next step involves extracting the known entity and the relations from the relation chain, integrating them into a formal query template. To optimize the retrieval process from a KG, we enhance efficiency by initially mapping entity and relation strings to their corresponding identifiers within the KG. This mapping information is conveniently stored in a separate file.

For instance, consider a KG represented in RDF[1] format and a corresponding SPARQL[2] query. The relation chain elucidated in Section 3.1 should be represented as follows, underscoring the seamless integration of the obtained information into a structured query framework.

```
PREFIX ent: <http://www.kg/entity/>
PREFIX rel: <http://www.kg/relation/>
SELECT DISTINCT ?ml WHERE {
    ent:E0 rel:R0 ?x.
    ?x rel:R1 ?y.
    ?y rel:R2 ?z.
    ?z rel:R3 ?m.
    ?m rdfs:label ?ml.
}
```

In this context, "ent" and "rel" serve as prefixes for entity and relation, respectively. The identifier "E0" uniquely represents "Eeyore" within the KG, while the identifiers for "creator," "child," "country of citizenship," and "capital" are denoted as "R0", "R1", "R2", and "R3" respectively. After identifying the entity "?m", we retrieve its string label "ml" as the final answer.

In conclusion, we harness the powerful capabilities of LLMs to analyze the question sentence and extract the relation chain—the foundation of a formal query. We systematically store all pertinent information, encompassing both updated and unchanged facts, within a KG. Armed with the formal query and the KG, our approach empowers us to conduct multi-hop question answering in a Knowledge-based Question Answering (KBQA) (Lan et al., 2022) fashion. Beyond efficiency, our GMeLLo approach stands out by offering explainability, a facet that will be elaborated upon in the next section.

[1] https://www.w3.org/RDF/
[2] https://www.w3.org/TR/sparql11-query/

#Edited instances		MQuAKE-CF 1	100	1000	3000	MQuAKE-T 1	100	500	1868
Base Model	Method								
GPT-J	MEMIT	12.3	9.8	8.1	1.8	4.8	1.0	0.2	0.0
GPT-J	MEND	11.5	9.1	4.3	3.5	38.2	17.4	12.7	4.6
GPT-J	MeLLo	20.3	12.5	10.4	9.8	**85.9**	45.7	33.8	30.7
GPT-J	**GMeLLo**	**30.0**	**30.0**	**30.0**	**30.0**	74.3	**74.3**	**74.3**	**74.3**
Vicuna-7B	MeLLo	20.3	11.9	11.0	10.2	**84.4**	56.3	52.6	51.3
Vicuna-7B	**GMeLLo**	**30.4**	**30.4**	**30.4**	**30.4**	65.6	**65.6**	**65.6**	**65.6**
GPT-3	MeLLo	**68.7**	50.5	43.6	41.2	**91.1**	**87.4**	**86.2**	85.5
GPT-3	**GMeLLo**	67.6	**67.6**	**67.6**	**67.6**	85.7	85.7	85.7	**85.7**

Table 1: Performance results of GMeLLo (ours) on MQuaKE-CF and MQuaKE-T using GPT-J, Vicuna-7B, or GPT-3 (text-davinci-003) as the base language model. Following the approach of Zhong et al. (2023), we group instances in batches of size k, where k takes values from 1, 100, 1000, 3000 for MQuAKE-CF and 1, 100, 500, 1868 for MQuaKE-T. The metric is multi-hop accuracy.

4 Experiment

Within our GMeLLo framework, we harness the analytical capabilities of LLMs to interpret sentences rather than tasking them with direct question-answering. In the upcoming section, we will conduct experiments to demonstrate the effectiveness and superiority of employing our GMeLLo methodology.

	#Edits	2-hop	3-hop	4-hop	Total
MQuaKE-CF	1	513	356	224	1,093
	2	487	334	246	1,067
	3	-	310	262	572
	4	-	-	268	268
	All	1,000	1,000	1,000	3,000
MQuaKE-T	1 (All)	1,421	445	2	1,868

Table 2: Data statistics of MQuAKE

4.1 Experiment Setup

4.1.1 Dataset

Our experiment centers on the multi-hop question-answering dataset, MQuAKE (Zhong et al., 2023). This dataset comprises MQuAKE-CF[3], designed for counterfactual edits, and MQuAKE-T, tailored for temporal knowledge updates. These datasets enable the evaluation of model editors under scenarios involving counterfactual changes and real-world temporal updates.

Table 2 provides a summary of the statistics for the MQuAKE-CF and MQuAKE-T datasets. The MQuAKE-CF dataset comprises 3,000 N-hop questions ($N \in \{2, 3, 4\}$), each linked to one or more edits. This dataset functions as a diagnostic tool for examining the effectiveness of knowledge editing methods in handling counterfactual edits. The MQuAKE-T dataset consists of 1,868 instances, each associated with a real-world fact change. Its purpose is to evaluate the efficacy of knowledge editing methods in updating obsolete information with contemporary, factual data.

[3]Due to constrained computational resources, our experiments on MQuAKE-CF are carried out on a randomly sampled subset of the complete dataset, comprising 3000 instances (1000 instances for each of 2, 3, 4-hop questions), aligning with the experiments outlined in Zhong et al. (2023).

4.1.2 Language Models

Similar to MeLLo, we broaden our investigation by integrating three robust language models into our framework. This expansion allows for a comprehensive comparison with baseline models, providing a more nuanced evaluation of our approach. Specifically, we leverage GPT-J (6B) (Wang and Komatsuzaki, 2021), vicuna-7B (Chiang et al., 2023), and text-davinci-003 (Ouyang et al., 2022).

4.1.3 Baseline Models

To demonstrate the effectiveness of our approach, we conduct comparisons with the following state-of-the-art knowledge editing methodologies.

- MEND (Mitchell et al., 2022a). It trains a hypernetwork to generate weight updates by transforming raw fine-tuning gradients based on an edited fact.

- MEMIT (Meng et al., 2023). It updates feed-forward networks across various layers to incorporate all relevant facts.
- MeLLo (Zhong et al., 2023). It employs a memory-based approach for multi-hop question answering, storing all updated facts in an external memory. In contrast to our GMeLLo, their approach retains only the updated facts, with each fact stored as a separate sentence.

4.1.4 Evaluation Metric

Building upon the framework proposed by Zhong et al. (2023), our evaluation employs the following metrics to assess the effectiveness of edits:

- Edit-wise success rate: gauging the successful recall of facts.
- Instance-wise accuracy: assessing the model's ability to recall all individual single-hop facts within multi-hop instances.
- Multi-hop accuracy: evaluating the model's accuracy in answering multi-hop questions.

Given our paper's primary focus on multi-hop question answering, we employ "multi-hop accuracy" as the main metric to assess the accuracy of both the original and edited language models in handling multi-hop questions.

4.2 Implementation Details and Key Findings

Given that the MQuAKE datasets provide both triples information and rewrite information, we construct a knowledge graph by connecting all the triples information. Subsequently, we modify the triple information based on the provided rewrite information to generate an updated knowledge graph.

Due to constrained computational resources, we opted to evaluate only the first multi-hop question in the MQuAKE dataset for our GMeLLo, rather than testing all three. To improve the understanding of this task by LLMs and ensure outputs conform to a specified format, we default to employing a 3-shot learning approach. This involves presenting the model with one 2-hop question sample, one 3-hop question sample, and one 4-hop question sample. To achieve comparable performance, we supplied Vicuna-7B with an additional set of 4-hop question sample. The reason will be discussed in Section 4.5.1. Due to GPT-J and Vicuna-7B's limitation in adhering to the desired output format, we establish a heuristic rule to extract essential information, outlined as follows:

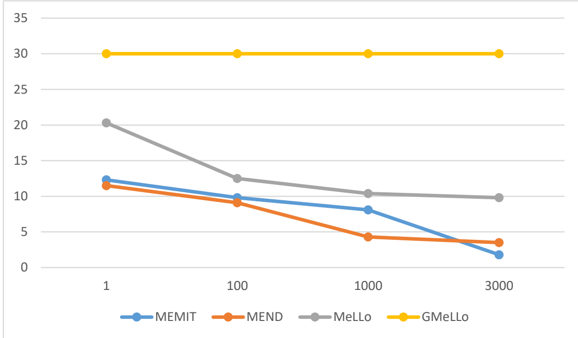

Figure 3: Multi-hop performance comparison of GPT-J before and after editing on MQuAKE-CF, utilizing different knowledge editing methods. The evaluation is conducted with varying numbers of edited instances (k) selected for editing, where k ranges from 1 to 3000.

- Narrow the attention to the output sentence containing the "->" indicator.
- Divide the sentence based on the "->" delimiter.
- Consider the initial segment as the predicted entity, and subsequently, process the following segments sequentially if they correspond to relations in the predefined relation list.

As illustrated in Table 1, our GMeLLo demonstrates significantly superior performance compared to state-of-the-art models on the MQuAKE-CF dataset, exhibiting an approximately 20% improvement when editing 3000 instances simultaneously. The sole source of error stems from the extraction of relation chains using LLMs. The recording of all fact edits in the KG eliminates the possibility of errors during fact retrieval. It is important to note that the relation chain remains consistent regardless of information updates. This confers a distinct advantage to our GMeLLo. As depicted in Figure 3, the integration of the latest information into our KG allows GMeLLo to sustain a consistent performance, even with an increasing number of edits. Nevertheless, in MeLLo, the expansion of external memory alongside a growing number of edited facts may result in slower and less accurate comparisons with the retriever (Izacard et al., 2022).

4.3 Breakdown Results on MQuAKE-CF

Tables 3 and 4 display the detailed results for MQuAKE-CF when employing GPT-J as the foundational model. Our analysis reveals that

	2-hop	3-hop	4-hop	All
MEND	13.9	11.3	**9.5**	11.5
MEMIT	22.5	6.0	8.4	12.3
GMeLLo	**54.8**	**27.0**	8.2	**30.0**

Table 3: Multi-hop performance breakdown on MQuAKE-CF for 2,3,4-hop questions using GPT-J as the base model.

# Edits=	1	2	3	4	All
MEND	16	11	7.3	4.4	11.5
MEMIT	20.5	9.8	5.5	2.6	12.3
GMeLLo	**34.5**	**34.4**	**24.8**	**5.2**	**30.0**

Table 4: Breakdown of multi-hop performance on MQuAKE-CF for questions with 1, 2, 3, 4 edits, utilizing GPT-J as the base model in this experiment.

- In 2-hop and 3-hop question answering, our method, GMeLLo, demonstrates twice the performance of the next best baseline. Furthermore, in 4-hop question answering, our method achieves comparable performance with the other two baseline models.

- In question answering with various edits, our model, GMeLLo, significantly outperforms the other two baseline models.

4.4 Performance in Addressing Single-Hop Questions

Although GMeLLo is primarily tailored for multi-hop question answering, it is adept at handling single-hop questions as well. As evidenced in Table 5, GMeLLo attains performance levels comparable to those of other approaches, even under the rigorous evaluation criteria of an exact match. In future iterations, we plan to implement semantic matching instead of relying on exact matches to extract more correct responses from LLMs. This involves identifying semantic equivalences, such as recognizing that "founder" which conveys the same meaning as "founded by" as correct output.

4.5 Futher Analyais

This subsection presents additional analyses conducted to identify errors in our experiments, showcase the advantages of employing GMeLLo, and explore potential applications.

Base Model	Method	Edit-wise	Instance-wise
GPT-J	MEND	72.8	59.6
	MEMIT	**97.4**	**94.0**
	GMeLLo	87.7	69.6
Vicuna-7B	MEND	65.2	47.6
	MEMIT	**96.6**	84.0
	GMeLLo	95.4	**84.9**

Table 5: Performance results for both edit-wise and instance-wise evaluations on MQuAKE-CF (with a maximum of 4 edits) are presented for baseline knowledge editing methods and our GMeLLo, utilizing two base models: GPT-J and Vicuna-7B. Each instance's associated edits are considered independently.

4.5.1 Error Analysis

Through our comprehensive comparative analysis, it became evident that GMeLLo consistently outperforms existing models in this specific task, especially when editing multiple instances. Among the three base models, Vicuna-7B demonstrates inferior performance compared to the other two, despite being provided with an additional 4-hop question answering sample in the prompt.

Following an in-depth error analysis, we identified that Vicuna exhibits more unconventional behavior. Instead of selecting a relation from the predefined list, it tends to create its own defined relations. For instance, it prefers using "citizen" to convey meaning rather than simply outputting "country of citizenship." This highlights the importance of prioritizing the consideration of meaning over strict exact matches in the mapping process—an aspect we plan to address in our future work. Another concern arises from the fact that, while Vicuna consistently identifies relations accurately—examples include "head of state" and "country of citizenship"—it frequently makes errors in their sequencing.

Moreover, our analysis uncovered some inconsistencies in the MQuAKE dataset. For instance,

- Question_1: Who founded The Christian Science Monitor?

- Multi-hop Relation in MQuAKE-CF: The Christian Science Monitor->headquaters location->?x->founded by->?y

- Prediction of Multi-hop Relations by Vicuna-7B: The Christian Science Monitor->founded by->?x

- Question_2: Who is the head of state of the country where the child of Kyle Reese has citizenship?

- Multi-hop Relation in MQuAKE-T: Kyle Reese->Spouse->?x->child->?y->country of citizenship->?z->head of state->?m

- Prediction of Multi-hop Relations by Vicuna-7B: Kyle Reese->child->?x->country of citizenship->?y->head of state->?z

While LLMs may accidentally provide correct answers, discerning the "headquarters location" from the first question and the "spouse" relation from the second question based solely on the question sentences is challenging.

4.5.2 Detection of Factual Inconsistencies

Throughout our experiments, we observed that simultaneous editing of numerous instances could lead to factual inconsistencies. For instance, the capital relationship might be exist in multiple questions. In a scenario from the counterfactual dataset, an edit changes the capital of one country to another city. However, to accurately answer the subsequent question, knowledge of the correct capital for that country is essential.The utilization of explicit external memory for storing all pertinent information, encompassing both updated and unchanged facts, clearly underscores these issues. Moreover, establishing rules, such as defining that a country should only have one capital, proves effective in preventing and addressing these types of inconsistencies.

4.5.3 Explainability

Illustrated by the yellow node path in Figure 2, our GMeLLo not only delivers answers but also offers traceability. This implies that we can retrieve the path leading to the obtained answer. Utilizing the clarity inherent in KG, GMeLLo is interpretable to a certain degree, providing a transparent understanding of the basis behind its responses.

4.5.4 Domain-specific Application

In the MQuAKE dataset, we establish direct connections among all triples to construct the KG. In cases where no triples are available, we can leverage the capabilities of LLMs to map diverse sentence representations into relation triples, as illustrated in Table 6. This process aligns with our endeavors in extracting relation chains.

Questions	Relation
Where did x graduate from? In which university did x study? What is x's alma mater?	educated_at(x,y)
What did x do for a living? What is x's job? What is the profession of x?	occupation(x, y)
Who is x's spouse? Who did x marry? Who is x married to?	spouse(x, y)

Table 6: Mapping natural language sentences to knowledge-base relations, illustrating the inverse process discussed by Levy et al. (2017) and Zhong et al. (2023), which can be implemented similarly to the relation chain extraction in our GMeLLo.

Although LLMs contain a wealth of information, they may not be privy to certain domain-specific confidential details. Moreover, the available domain-specific data might fall short for training an LLM from the ground up, adding to the substantial resources required. Nevertheless, domain-specific databases should be able to support knowledge graph construction. In such cases, our GMeLLo approach serves as a crucial bridge, allowing the harnessing of LLMs' formidable capabilities without the necessity of revealing sensitive information.

5 Conclusion

In this paper, we present a memory-based knowledge editing approach tailored for multi-hop question answering. This method leverages the capabilities of LLMs to analyze question sentences and generate a relation chain, rather than providing direct answers to the questions. The rationale behind this lies in the observation that linguistic patterns change more slowly than specific information. We construct the KG directly from the dataset and transform the relation chain, extracted by LLMs, into a formal query to retrieve information from the KG. This approach capitalizes on the strengths of both LLMs and KGs—leveraging the high coverage of LLMs and the precision of using KGs. By utilizing LLMs to comprehend most sentences and KBQA to provide accurate and explainable results, we achieve a synergy between the two methodologies.

Limitations

Nevertheless, it's important to note that this investigation is still in its early stages. Although our performance surpasses that of baseline approaches, especially the multi-hop question answering when editing multiple facts simultaneously, we recognize the potential for further improvement. Looking ahead, our future plans involve enhancing GMeLLo in the following key areas:

- Experiment with more sophisticated prompts, such as Chain of Thought (CoT) (Wei et al., 2022), to elevate performance.
- Emphasize the identification of semantically similar relations, aiming to mitigate potential confusion between them and thereby enhance overall performance.
- Contrast the output of LLMs with the golden relations in terms of semantics, prioritizing meaningful matches over exact verbatim matches, to yield more correct responses.
- Pioneering the integration of the strengths inherent in both LLMs and KGs, we aim to extend their application to diverse research endeavors.

References

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Yong Cao, Xianzhi Li, Huiwen Liu, Wen Dai, Shuai Chen, Bin Wang, Min Chen, and Daniel Hershcovich. 2023. Pay more attention to relation exploration for knowledge base question answering. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 2119–2136, Toronto, Canada. Association for Computational Linguistics.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2023. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240):1–113.

Damai Dai, Li Dong, Yaru Hao, Zhifang Sui, Baobao Chang, and Furu Wei. 2022. Knowledge neurons in pretrained transformers. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8493–8502, Dublin, Ireland. Association for Computational Linguistics.

Nicola De Cao, Wilker Aziz, and Ivan Titov. 2021. Editing factual knowledge in language models. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6491–6506, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Qingxiu Dong, Damai Dai, Yifan Song, Jingjing Xu, Zhifang Sui, and Lei Li. 2022. Calibrating factual knowledge in pretrained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5937–5947, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey on in-context learning.

Anshita Gupta, Debanjan Mondal, Akshay Sheshadri, Wenlong Zhao, Xiang Li, Sarah Wiegreffe, and Niket Tandon. 2023. Editing common sense in transformers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8214–8232.

Thomas Hartvigsen, Swami Sankaranarayanan, Hamid Palangi, Yoon Kim, and Marzyeh Ghassemi. 2022. Aging with grace: Lifelong model editing with discrete key-value adaptors. In *NeurIPS 2022 Workshop on Robustness in Sequence Modeling*.

Zeyu Huang, Yikang Shen, Xiaofeng Zhang, Jie Zhou, Wenge Rong, and Zhang Xiong. 2022. Transformer-patcher: One mistake worth one neuron. In *The Eleventh International Conference on Learning Representations*.

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*.

Yunshi Lan, Gaole He, Jinhao Jiang, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. 2022. Complex knowledge base question answering: A survey. *IEEE Transactions on Knowledge and Data Engineering*.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342.

Xiaopeng Li, Shasha Li, Shezheng Song, Jing Yang, Jun Ma, and Jie Yu. 2023. Pmet: Precise model editing in a transformer.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. 2022. Locating and editing factual associations in gpt. In *Advances in Neural Information Processing Systems*, volume 35, pages 17359–17372. Curran Associates, Inc.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. 2023. Mass editing memory in a transformer. *The Eleventh International Conference on Learning Representations (ICLR)*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D Manning. 2022a. Fast model editing at scale. In *International Conference on Learning Representations*.

Eric Mitchell, Charles Lin, Antoine Bosselut, Chelsea Finn, and Christopher D. Manning. 2022b. Memory-based model editing at scale. In *International Conference on Machine Learning*.

Zhijie Nie, Richong Zhang, Zhongyuan Wang, and Xudong Liu. 2024. Code-style in-context learning for knowledge-based question answering.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul F Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pages 27730–27744. Curran Associates, Inc.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2023. Unifying large language models and knowledge graphs: A roadmap. *arXiv preprint arXiv:2306.08302*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Jack W. Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, Eliza Rutherford, Tom Hennigan, Jacob Menick, Albin Cassirer, Richard Powell, George van den Driessche, Lisa Anne Hendricks, Maribeth Rauh, Po-Sen Huang, Amelia Glaese, Johannes Welbl, Sumanth Dathathri, Saffron Huang, Jonathan Uesato, John Mellor, Irina Higgins, Antonia Creswell, Nat McAleese, Amy Wu, Erich Elsen, Siddhant Jayakumar, Elena Buchatskaya, David Budden, Esme Sutherland, Karen Simonyan, Michela Paganini, Laurent Sifre, Lena Martens, Xiang Lorraine Li, Adhiguna Kuncoro, Aida Nematzadeh, Elena Gribovskaya, Domenic Donato, Angeliki Lazaridou, Arthur Mensch, Jean-Baptiste Lespiau, Maria Tsimpoukelli, Nikolai Grigorev, Doug Fritz, Thibault Sottiaux, Mantas Pajarskas, Toby Pohlen, Zhitao Gong, Daniel Toyama, Cyprien de Masson d'Autume, Yujia Li, Tayfun Terzi, Vladimir Mikulik, Igor Babuschkin, Aidan Clark, Diego de Las Casas, Aurelia Guy, Chris Jones, James Bradbury, Matthew Johnson, Blake Hechtman, Laura Weidinger, Iason Gabriel, William Isaac, Ed Lockhart, Simon Osindero, Laura Rimell, Chris Dyer, Oriol Vinyals, Kareem Ayoub, Jeff Stanway, Lorrayne Bennett, Demis Hassabis, Koray Kavukcuoglu, and Geoffrey Irving. 2022. Scaling language models: Methods, analysis & insights from training gopher.

Ankit Singh Rawat, Chen Zhu, Daliang Li, Felix Yu, Manzil Zaheer, Sanjiv Kumar, and Srinadh Bhojanapalli. 2020. Modifying memories in transformer models. In *International Conference on Machine Learning (ICML) 2021*.

Vipula Rawte, Amit Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *arXiv preprint arXiv:2309.05922*.

Anton Sinitsin, Vsevolod Plokhotnyuk, Dmitry Pyrkin, Sergei Popov, and Artem Babenko. 2020. Editable neural networks. In *International Conference on Learning Representations*.

Ben Wang and Aran Komatsuzaki. 2021. GPT-J-6B: A 6 Billion Parameter Autoregressive Language Model. `https://github.com/kingoflolz/mesh-transformer-jax`.

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

Yunzhi Yao, Peng Wang, Bozhong Tian, Siyuan Cheng, Zhoubo Li, Shumin Deng, Huajun Chen, and Ningyu Zhang. 2023. Editing large language models: Problems, methods, and opportunities. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10222–10240, Singapore. Association for Computational Linguistics.

Zexuan Zhong, Zhengxuan Wu, Christopher Manning, Christopher Potts, and Danqi Chen. 2023. MQuAKE: Assessing knowledge editing in language models via multi-hop questions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 15686–15702, Singapore. Association for Computational Linguistics.

A Appendix

A.1 Comparison between existing KBQA methods and our GMeLLo

We evaluate the performance of existing KBQA approaches, such as KB-Coder (Nie et al., 2024). Our findings indicate that, when provided with similar prompts, our approach yields more accurate results. For example, when presented with a 4-hop sample in the prompt and parsing the question "What is the capital of the country of citizenship of the child of the creator of Eeyore?" KB-Coder yields the following results:

expression = START('Eeyore')
expression = JOIN('child of', expression)
expression = JOIN('creator', expression)
expression = JOIN('country of citizenship', expression)
expression = JOIN('child', expression)
expression = STOP(expression)

However, the resulting relation chain given by our GMeLLo is notably more accurate:

"Eeyore -> creator -> ?x -> child of -> ?y -> country of citizenship -> ?z -> capital -> ?m"