Enhancing Textbook Question Answering with Knowledge Graph-Augmented Large Language Models

Mengliang He 51255901020@stu.ecnu.edu.cn
Aimin Zhou amzhou@cs.ecnu.edu.cn
Xiaoming Shi[*] xmshi@ir.hit.edu.cn
School of Computer Science and Technology, East China Normal University, Shanghai 200062, China

Editors: Vu Nguyen and Hsuan-Tien Lin

Abstract

Previous works on **T**extbook **Q**uestion **A**nswering suffer from limited performance due to the small-scale neural network based backbone. To alleviate the issue, we propose to utilize LLMs as the backbone of TQA tasks. To this end, we utilize two methods, the raw-context based prompting method and the knowledge graph based prompting method. Specifically, we introduce the Textbook Question Answering-Knowledge Graph (TQA-KG) method, which first converts textbook content into structural knowledge graphs and then combining knowledge graph into LLM prompting, thereby enhancing the model's reasoning capabilities and answer accuracy. Extensive experiments conducted on the CK12-QA dataset illustrate the effectiveness of the method, achieving an improvement of 5.67% in accuracy compared to current state-of-the-art methods on average.

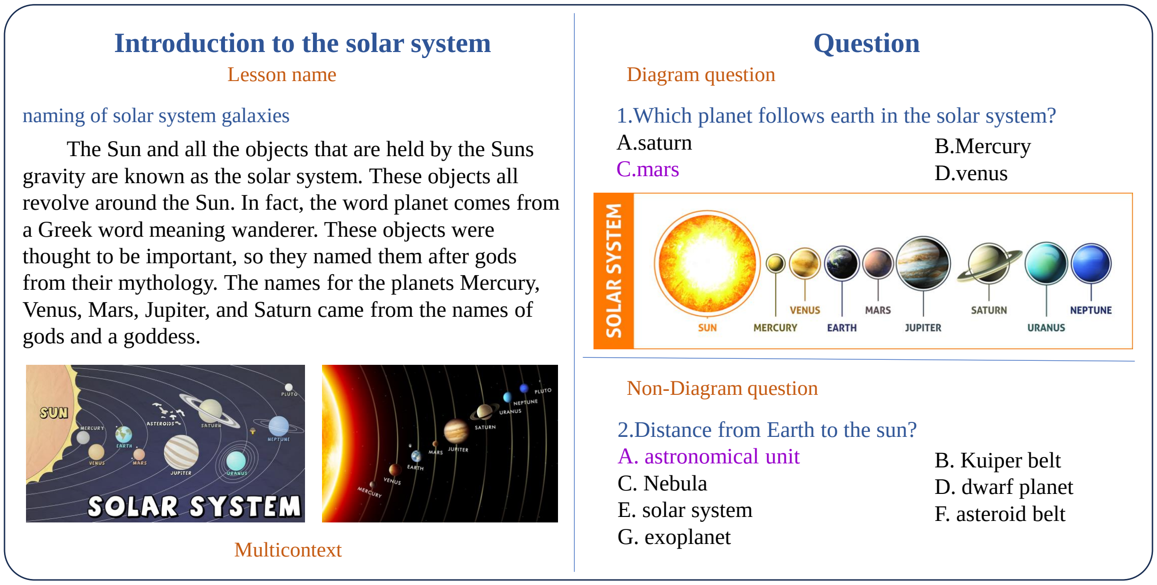
Keywords: Textbook Question Answering, Large language model, knowledge graphs, Retrieve-and-Generate

1. Introduction

Question **A**nswering (QA) tasks, such as VQA (Yu et al., 2019) and MRC (Nie et al., 2019), is a research hotspot in **N**atural **L**anguage **P**rocessing (NLP), thanks to the alluring technological and application value. Among various QA tasks, **T**extbook **Q**uestion **A**nswering (TQA) (Kembhavi et al., 2017) is an emerging task, which is designed to provide answers for questions based on the text or diagrams in given textbooks. As shown in Figure 1, it is required to answer non-diagram and diagram questions on the right based on learning and understanding the textbook fragment provided on the left.

Current works can be mainly divided into two categories. First, some works (Gomez-Perez and Ortega, 2020; Xu et al., 2023) leverage advanced neural network architectures, particularly focusing on attention mechanisms to parse and integrate information from various sources. Besides, (Kim et al., 2019; Wang et al., 2022; Ma et al., 2023b,a) employ graph-based and semantic understanding methods and focus on integrating complex knowledge structures and multimodal data to improve understanding and reasoning in TQA. However, these works suffer from limited performance due to the small-scale neural network based backbone.

[*] Corresponding Author



Introduction to the solar system

Lesson name

naming of solar system galaxies

The Sun and all the objects that are held by the Suns gravity are known as the solar system. These objects all revolve around the Sun. In fact, the word planet comes from a Greek word meaning wanderer. These objects were thought to be important, so they named them after gods from their mythology. The names for the planets Mercury, Venus, Mars, Jupiter, and Saturn came from the names of gods and a goddess.

Multicontext

Question

Diagram question

1.Which planet follows earth in the solar system?

A.saturn B.Mercury

C.mars D.venus

Non-Diagram question

2.Distance from Earth to the sun?

A. astronomical unit B. Kuiper belt

C. Nebula D. dwarf planet

E. solar system F. asteroid belt

G. exoplanet

Figure 1: An example of the CK12-QA dataset. The left side illustrates the textbook content, while the right side illustrates non-diagram and diagram questions for the given textbook content.

Recently, the revolutionary progress in **L**arge **L**anguage **M**odels (LLM) (Achiam et al., 2023; Team et al., 2023; Anthropic, 2024) has catalyzed substantial technological transformations in NLP and reshape NLP's foundation. Inspired by LLMs, we propose to utilize LLMs as the backbone of TQA tasks. To this end, we utilize two manners, the raw-context based prompting method and **k**nowledge **g**raph (KG) based prompting method.

Specifically, the method of prompting based on the raw context entails the direct application of contexts in natural language as the input for LLMs. However, the raw context based prompting suffers from long context, which restricts performance and inference speed. To alleviate the issue, we propose a new approach to utilize KG based prompting method for TQA, termed TQA-KG, for the non-diagram problem in the TQA task. KGs are structured, semantic networks that capture entities, their attributes, and relationships, which are widely used. In TQA-KG, we adopt an iterative step-by-step generation strategy to extract KG from textbook texts using LLMs, and then we combine the textbook KG with instructions as the input of LLMs. In addition, the techniques of **R**etrieval **A**ugmented **G**eneration (RAG) (Lewis et al., 2020) is employed to better utilize extracted KG, which enhance the credibility and accuracy of the responses of LLMs, and improve the performance of the TQA task. The pipeline of the TQA-KG method is shown in Figure 2. It is noteworthy that the comparative performance of raw-context and the KG derived from raw-context for LLM prompting has been insufficiently investigated. This study represents the inaugural effort to systematically analyze and juxtapose these two approaches.

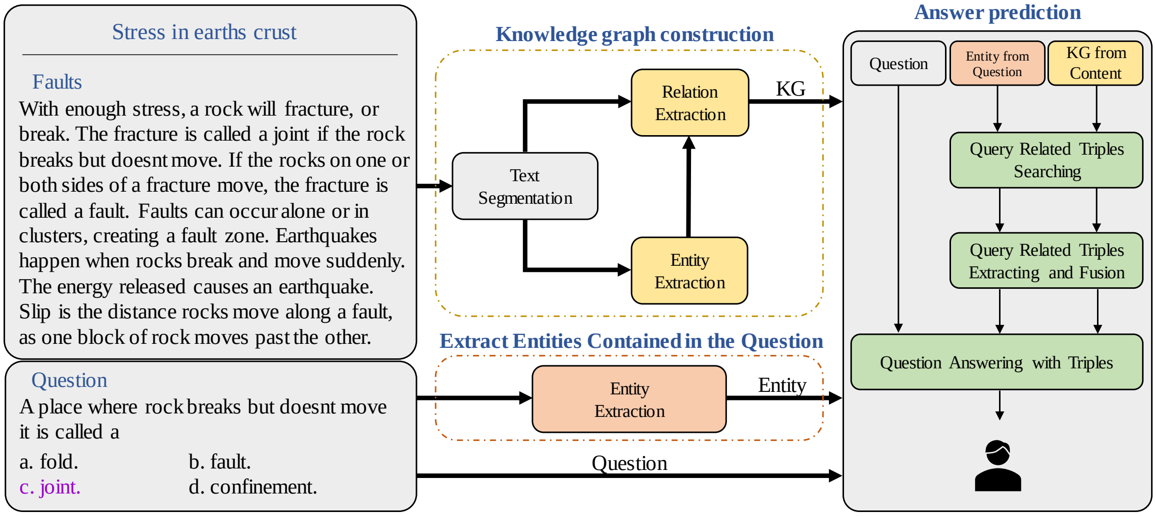

Stress in earths crust
Faults
With enough stress, a rock will fracture, or break. The fracture is called a joint if the rock breaks but doesnt move. If the rocks on one or both sides of a fracture move, the fracture is called a fault. Faults can occur alone or in clusters, creating a fault zone. Earthquakes happen when rocks break and move suddenly. The energy released causes an earthquake. Slip is the distance rocks move along a fault, as one block of rock moves past the other.
Question
A place where rock breaks but doesnt move it is called a
a. fold.
b. fault.
c. joint.
d. confinement.
Knowledge graph construction
Text Segmentation
Relation Extraction
Entity Extraction
KG
Extract Entities Contained in the Question
Entity Extraction
Entity
Question
Answer prediction
Question
Entity from Question
KG from Content
Query Related Triples Searching
Query Related Triples Extracting and Fusion
Question Answering with Triples

Figure 2: Framework of the TQA-KG approach. The left side shows course content and a question from the CK12-QA dataset, the middle side shows constructing the KG through the course content and extracting entities from the question, and the far right side shows combining the KG with the LLMs to enhance the accuracy of the prediction results ultimately.

Extensive experiments are conducted on the CK12-QA dataset (Kembhavi et al., 2016). The results of both experiments show the effectiveness and high inference speed of TQA-KG by outperforming the current state-of-the-art methods.

To summarize, the contributions of this work are as follows:

- We propose the TQA-KG approach that can be applied to LLMs as an additional component. Combining LLMs with KG improves the performance and inference speed of LLMs on the TQA task.
- To evaluate the effectiveness of TQA-KG, we conducted extensive experiments on the CK12-QA dataset and compared our results with other methods. Experimental results demonstrate the superiority of our method in improving answer accuracy.
- We are the first to explore and analyze the effect of incorporating LLM in TQA tasks.

2. Related Work

2.1. TQA Task

The earliest research on the TQA task was started by (Kembhavi et al., 2017), which presented the CK12-QA dataset, consisted of 1,076 lessons and 26,260 non-diagram and diagram problems taken from a secondary school science curriculum. They used the BiDAF model (Seo et al., 2016) and VQA model, which had better performance at the time, to test the non-diagram and diagram problems respectively, and completed the baseline of the TQA dataset. Since then there has been a lot of work around TQA as well, e.g., XTQA focuses on improving the interpretability of the answers of the TQA system, and DDGNet (Wang et al., 2023) performs question-guided multi-step inference on the dynamic directed **D**iagram **G**raph **N**etwork (DGN) and **T**extual **G**raph **N**etworks (TGN) to improve the performance of the TQA task. MoCA (Xu et al., 2023) combines multi-stage domain pre-training and cross-directed multi-modal attention for the TQA task.

2.2. LLM-augmented KG Construction

LLMs have great potential in the field of KG generation. In recent years, (Yao et al., 2023) introduced an innovative framework called Knowledge Graph LLM (KG-LLM), through which triples are modeled. KG-LLM employs entity-relationship triples as prompts, and predictions are made through responses, which ultimately lead to KG completion. On the other hand, (Meyer et al., 2023), proposed the first LLM **K**nowledge **G**raph **E**ngineering (KGE) benchmarking framework, LLMKG-Bench, which provides an automated and continuous evaluation platform for different tasks in KGE scenarios. (Carta et al., 2023) use generative LLMs (such as GPT-4) to construct KG, and complete the construction of text to KG under zero-shot conditions through an iterative method. (Ning and Liu, 2024) introduced UrbanKGent, a unified framework using a large language model agent to automate the construction of urban knowledge graphs, and demonstrates its practical applicability by providing richer urban knowledge graphs with reduced data requirements.

2.3. KG-enhanced LLM Inference

Although LLMs have powerful natural language understanding capabilities, the "hallucination" problem has been a great challenge in using LLMs (Ji et al., 2023). In recent years, many approaches have been proposed to solve the hallucination problem in LLMs, among which KG-enhanced LLMs have become a promising solution. Based on the semantic similarity between a question and its associated facts, KAPING (Baek et al., 2023) retrieves facts related to the input question from a knowledge graph and adds them to the input question to generate a more accurate answer. (Wen et al., 2023) built a cueing pipeline to enable LLMs to understand KG inputs and reason by combining the hidden knowledge with the retrieved external knowledge. (Sun et al., 2023) proposed a new LLM-KG integration paradigm "LLM$\otimes$KG". It treats the LLM as an agent that interactively explores relevant entities and relationships with the KG and reasons based on the retrieved knowledge.

3. Methodology

3.1. Overall

We describe the TQA-KG method in detail in the following. In Section 3.2, we delineate the application of LLMs for the extraction and generation of KG from the textual content of textbooks. Section 3.3 will introduce the direct application of textbook texts as raw-context to enhance LLM responses. And in Section 3.4, we present the TQA-KG method, which combines LLMs with the generated KG to achieve high-quality Textbook Question Answering.

3.2. KG Extraction and Construction

Inspired by previous work (Trajanoska et al., 2023), we decided to split the big task of KG extraction and construction into three small tasks, namely, segmenting long text, extracting entities, and extracting entity relationships, to better utilize the performance of LLMs and improve the quality of KG construction.

3.2.1. Recursive Segmentation

Long text processing has been a major challenge in the field of large models, so in the first step of extracting KG, the textbook text is divided by length. Each segmented text is set as a chunk and takes the chunk as the smallest processing unit of LLMs. At the same time, to prevent the loss of semantic information of the text due to the segmentation operation, and also to prevent two related entities from being separated due to the text segmentation, the chunk is repeatedly sampled by recursive segmentation. Recursive segmentation effectively diminishes the likelihood of fragmenting two correlated entities by defining text chunks as partially overlapping sliding windows of the input. This approach illustrates that larger windows and greater overlaps substantially mitigate the probability of entity separation.

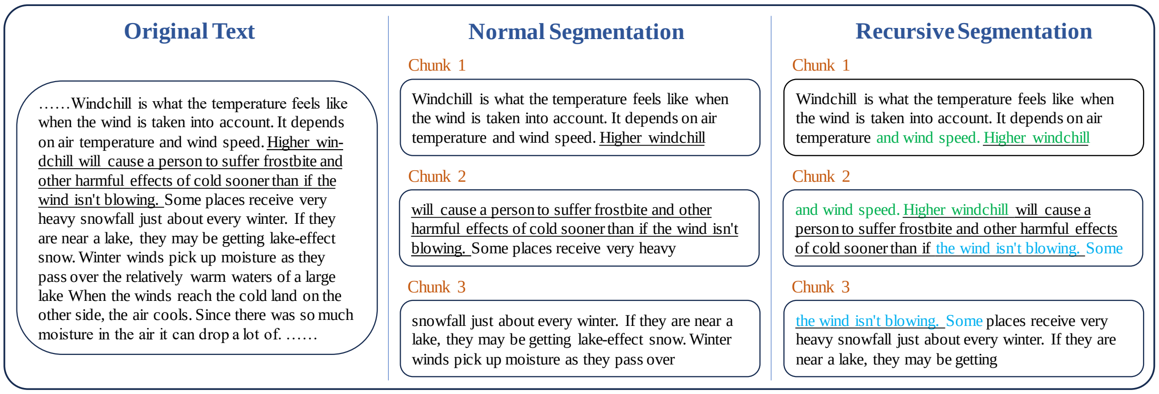
Fig. 3 demonstrates the principle of the recursive segmentation method. On the left of the image is the original text, highlighted with an underlined example sentence. In the normal segmentation approach shown in the middle, the underlined example sentence is split across different chunks, resulting in a loss of semantic integrity. On the right, the recursive segmentation method repeatedly samples the last n characters in the sliding window of the chunk (indicated by text in the same color), ensuring that the example sentence is wholly contained within one chunk, thus preserving its semantic integrity. In this way, the excessively long textbook text can be transformed into a series of chunks without losing semantic information. For our study, we adopted a resampling ratio commonly used in most RAG work, which is 20% of the original chunk length. For instance, if the text length is 500, the resampled text length before and after the chunk would be 100.

3.2.2. Extract Entities and Relationships

Some previous work, such as BEAR (Yu et al., 2023) attempted to directly extract KG triples in text end-to-end using LLMs, allowing LLMs to output the extracted pairs of entities and the relationships between the entities at once. In contrast, the traditional manual method of constructing KG involves extracting the entities first and only then looking for the relationships between the entities (Lehmann et al., 2015). Inspired by this, we decided to separate the extraction of entities and the extraction of inter-entity relationships and designed a set of prompts for extracting entities, and entity relationships. A chunk is the smallest unit of knowledge graph construction so the same chunk is operated twice in an iterative manner.

First, the chunk is fed into the LLMs, and based on the prompt engineering, the LLMs return a list of entities extracted from the chunk, and the length of each chunk is limited due to the text segmentation operation that has been performed.

Then the chunk and the extracted list of entities are re-fed together to LLMs, which is prompted to find the semantic relationship between each two entities in the text of the



……Windchill is what the temperature feels like when the wind is taken into account. It depends on air temperature and wind speed. Higher windchill will cause a person to suffer frostbite and other harmful effects of cold sooner than if the wind isn't blowing. Some places receive very heavy snowfall just about every winter. If they are near a lake, they may be getting lake-effect snow. Winter winds pick up moisture as they pass over the relatively warm waters of a large lake When the winds reach the cold land on the other side, the air cools. Since there was so much moisture in the air it can drop a lot of. ……

Chunk 1

Windchill is what the temperature feels like when the wind is taken into account. It depends on air temperature and wind speed. Higher windchill

Chunk 2

will cause a person to suffer frostbite and other harmful effects of cold sooner than if the wind isn't blowing. Some places receive very heavy

Chunk 3

snowfall just about every winter. If they are near a lake, they may be getting lake-effect snow. Winter winds pick up moisture as they pass over

Chunk 1

Windchill is what the temperature feels like when the wind is taken into account. It depends on air temperature and wind speed. Higher windchill

Chunk 2

and wind speed. Higher windchill will cause a person to suffer frostbite and other harmful effects of cold sooner than if the wind isn't blowing. Some

Chunk 3

the wind isn't blowing. Some places receive very heavy snowfall just about every winter. If they are near a lake, they may be getting

Figure 3: The original text is in the left block, with an example sentence marked by an underline. The middle block depicts a conventional segmentation approach, while the right block shows the recursive segmentation method, where text in the same color represents parts that have been resampled.

chunk and output it in the form of a triple. This completes the construction of a KG triple for a chunk. The resultant KG is systematically structured and stored in the form of triples.

3.3. Raw-context-enhanced LLM Inference

The TQA dataset comprises two main components: textbook content enriched with multimodal information and the questions designed based on this content. For the current experiment employing the CK12-QA dataset, each data entry includes an indeterminate number of textbook passages ranging from 1 to 21, alongside 1 to 34 variably numbered questions. The raw-context based prompting method involves concatenating all textual content from the textbook within each data entry to form a raw-context. This raw-context is then utilized as supplementary material in conjunction with the questions when input into the LLMs for the prediction of answers.

3.4. KG-enhanced LLM Inference

Retrieval-augmented generation models (e.g., RAG and RALM (Liu et al., 2019)) provide context-awareness of documents when dealing with knowledge-intensive tasks. By extending the knowledge scope of the model and providing context-aware documents, the KG-augmented retrieval technique helps to mitigate the illusion problem in LLMs. With the help of KG, LLMs are able to consider contextual information and external knowledge more comprehensively, which reduces the possibility of error generation and improves the quality and accuracy of the generated text (Agrawal et al., 2023).

This paper employs the RAG methods to develop the TQA-KG method, as illustrated in Fig. 2. Initially, the textbook question is input into the LLM, which employs the previously delineated "Entity Extraction" method to identify a list of entities associated with the question. Subsequently, each entity on this list is matched with nodes from the constructed KG. Upon successful matching, all triples containing the corresponding node are extracted. The RAG method was initially utilized, whereby all extracted triples, along with the question, were input into the LLM for further processing. However, due to the large number of extracted triples and not all of them are complementary knowledge related to the problem. Consequently, requiring the LLMs to process a substantial volume of triples while simultaneously addressing a question could not only compromise the quality of the LLM's responses but also hinder its problem-solving capabilities. This interference stems from the incorporation of superfluous knowledge from triples that are irrelevant to the posed question, thereby detracting from the efficiency of its computational processes. So we decompose the complex task into:

- Initially, the KG is queried based on entities extracted from the question, resulting in the derivation of a subset referred to as sub-KG1.
- Subsequently, this subset (sub-KG1) is further refined using a LLM. The sub-KG1, along with the original question, is then input into the LLM, which is tasked with generating a more focused subset, sub-KG2, exclusively pertinent to the question.
- The final step involves inputting sub-KG2, along with the question, back into the LLM, which is then responsible for predicting the answer based on the refined knowledge graph information.

Ultimately, this methodology enables the LLM to sequentially complete the task and accurately predict the answer to the question.

4. Experiment

4.1. Experimental Setting

4.1.1. Dataset and Evaluations

Currently, there are two mainstream datasets for the TQA task, namely CK12-QA and AI2D (Kembhavi et al., 2016). Since all the data in the AI2D dataset are diagram questions and this paper only discusses non-diagram questions, the non-diagram questions in the CK12-QA dataset are finally used. The CK12-QA dataset is drawn from middle school science curricula. It consists of 1,076 lessons from Life Science, Earth Science, and Physical Science textbooks. This includes 26,260 questions, of which 12,567 have an accompanying diagram. In addition to the diagram questions, there are 13,693 non-diagram questions, which are divided into two categories: multiple-choice questions with 4-7 choices, and true-false questions with only two choices. To assess the effectiveness of the TQA-KG method, we have prepared four sets of tests using the original LLM, the LLM fine-tuned with the CK12-QA dataset, and the LLM employing the TQA-KG method and raw-context method. The evaluation will be based on the accuracy of matching the predicted results with the ground truth.

4.1.2. Implementation Details

To ensure fairness, the task of constructing the knowledge graph was carried out by a non-test model, ChatGPT-3.5 (Ouyang et al., 2024). All our test models are implemented in the Ollama framework[1] for inference and use llamafactory (Zheng et al., 2024) for instruction fine-tuning. We use the LoRA fine-tuning method (Hu et al., 2021) with the specific parameters Learning rate of 5e-5, training epochs of 3, Maximum gradient norm of 1.0, Max samples of 100000, Batch size of 2, and LR scheduler of cosine. After fine-tuning, the model was reloaded into Ollama for test use. As for the main hardware information of the experimental server, the CPU model is Intel(R) Xeon(R) Silver 4410Y, the RAM is 124G, and the graphics card model is one NVIDIA RTX A6000.

4.1.3. Baseline Model

In order to demonstrate the effectiveness of our TQA-KG method, we used several LLMs for baseline testing, while due to hardware performance limitations, we selected most of the models with 7-8B parameters and a 13B Llama 2 as an additional reference, respectively, as shown below:

Llama 2(7/13B): Llama 2 is released by Meta Platforms, Inc. This model is trained on 2 trillion tokens, and by default supports a context length of 4096. Llama 2 Chat models are fine-tuned on over 1 million human annotations and are made for chat.

Llama 3(8B): Llama 3 instruction-tuned models are fine-tuned and optimized for dialogue/chat use cases and outperform many of the available open-source chat models on common benchmarks.

Mistral(7B): Mistral is a 7.3B parameter model, distributed with the Apache license. It is available in both instruct (instruction following) and text completion.

Zephyr(7B): Zephyr is a series of fine-tuned versions of the Mistral and Mixtral models that are trained to act as helpful assistants.

Gemma(7B): Gemma is a new open model developed by Google and its DeepMind team. It's inspired by Gemini models at Google.

4.2. Constructing Datasets for LLM Instruction Fine-tuning

Instruction tuning LLMs for specific tasks or domains is an effective strategy to enhance the model's capabilities and augment its knowledge repository within that particular vertical (Gupta et al., 2023). Since the ultimate goal is a QA task, we decided to build a QA dialog dataset based on the textbook text of the CK12-QA. This dataset is used to teach the knowledge of the textbook text, as well as the Q&A approach in an instruction fine-tuned manner to the LLMs. First, utilize the strong reading comprehension and role-playing ability of the LLMs by following the "play-as-playwrights" instruction technique (Zhang et al., 2024), which allows the LLMs to act as a scriptwriter, fictionalizing two conversation users, the two users have a knowledge quiz based on the textual content of the textbook, and the quizzed dialog content, i.e., the final QA dialog dataset. Then fine-tune the LLMs with instructions on the alpaca_gpt4_en dataset (Peng et al., 2023) and generated QA dialog dataset from train and validation set to get our TQA task LLMs. Table 1. shows the number of QA dialogue datasets finally obtained.

[1] https://ollama.com/

CK12-QA	**QA dialog**
train set	22992
val set	6847
test set	7330

Table 1: The number of dialogue datasets extracted from CK12-QA.

4.3. Results on KG Construction

As illustrated in Table 2, a total of 30,499 entities and 58,933 relationships were extracted from the three parts of the CK12-QA dataset. On average, each chunk yielded 2.17 entities and 4.18 edges, with the average lengths of entities and relationships being 11.3 and 70 characters, respectively. With the least possible information loss, we transformed the minimal information units from original chunks of 500 characters into knowledge graph triples with an average length of 92.6 characters. This reduction to 18.52% of the original chunk volume decreased irrelevant noise within the minimal information units and focused more sharply on the information pertaining to individual knowledge points, rather than including two to three knowledge points typically found in a chunk.

CK12-QA	**chunks**	**nodes**	**edges**
train set	8757	16774	35925
val set	2509	6163	10812
test set	2816	7562	12196

Table 2: The number of triplets extracted from CK12-QA.

4.4. Results on CK12-QA

We conducted evaluations on six different models using the CK12-QA dataset, From Figure 4, we can observe that both the TQA-KG and the raw-context methods substantially surpass the baseline results, demonstrating that with the aid of additional pertinent information, the LLMs is capable of more accurately responding to relevant questions. Furthermore, the accuracy of the TQA-KG method is on average 0.89% higher than that of the raw-context method. This suggests that the TQA-KG method not only condenses information but also achieves an average performance comparable to the raw-context method. TQA-KG extracts knowledge from textbook texts in the form of triples, eliminating ineffective and redundant noise. This approach maintains the essential knowledge needed to answer questions while streamlining the information. Moreover, compared to the raw-context method, which has an average character length of 4647.9, the average length of triples retrieved for each question is only 1032.7, constituting 22.2% of the former. This reduction in information simplifies the prompt's token length, thereby allowing for stable performance even on large models with smaller contextual windows, without the issues of model inefficiency due to exceeding the contextual window with the raw-context length.

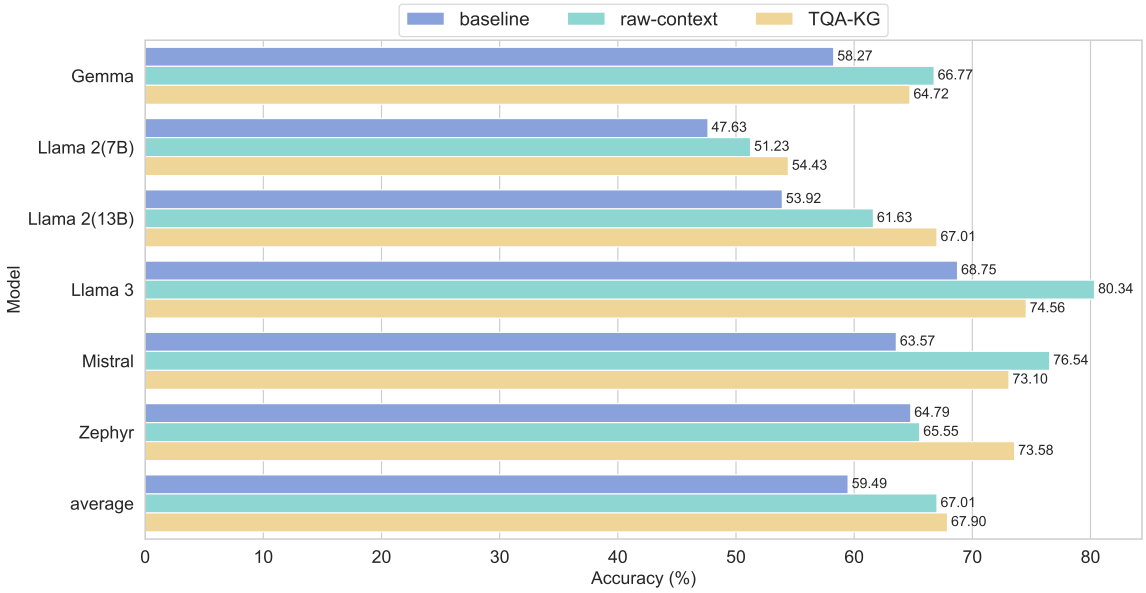


Figure 4: The results of six models on three settings, baseline, raw-context and TQA-KG on the validation set. These three methods are training-free. The results are presented in percentage (%)

Table 3, demonstrates that the LLMs incorporating the TQA-KG method outperform both the baseline and fine-tuned models. Specifically, the TQA-KG method exhibited a minimum increase in accuracy of 1.21% on the Non-Diagram True/False (ND T/F) question type and a similar increase of 5.31% on the Non-Diagram Multiple Choice (ND MC) question type. It is evident that the baseline model, lacking any supplementary information, relies solely on the knowledge retained during its pre-training phase. Consequently, when faced with questions in a new domain, this model often lacks sufficient reference information, leading to the generation of incorrect answers. In contrast, the TQA-KG method enhances the large model's comprehension of the questions by integrating relevant knowledge graph triples, thereby enabling the model to respond to questions more accurately and effectively.

Additionally, we also employed the instruction tuning dataset, extracted as mentioned previously, to fine-tune the baseline model. This approach was intended to evaluate the gap between the TQA-KG method and traditional fine-tuning techniques. As shown in Table 3, the TQA-KG method consistently outperformed the instruction tuning method, showing a minimum increase in accuracy of 2.63%/1.06% for the ND T/F and ND MC question types, respectively. This indicates that the domain knowledge infused into the model through instruction tuning is often limited, and the results suggest that our proposed TQA-KG method utilizes knowledge more effectively than instruction tuning.

Overall, as shown in Fig. 5, the TQA-KG method leverages structured information from KG to enhance LLMs, achieving an average accuracy improvement of 5.67% over the baseline model. The Llama 2(13b) model, in particular, demonstrated an increase in

ND T/F	**Gemma**	**Llama 2(7B)**	**Llama 2(13B)**	**Llama 3**	**Mistral**	**Zephyr**	**average**
baseline	62.61	63.05	65.35	73.36	71.16	73.68	68.20
fine-tine	67.76	55.26	57.89	65.57	69.52	64.04	63.34
TQA-KG(ours)	**70.39**	**70.07**	**74.67**	**74.56**	**79.82**	**78.40**	**74.65**
ND MC	**Gemma**	**Llama 2(7B)**	**Llama 2(13B)**	**Llama 3**	**Mistral**	**Zephyr**	**average**
baseline	53.19	40.00	49.19	65.31	57.88	57.81	53.90
fine-tine	57.50	31.50	42.88	64.56	64.88	57.25	53.09
TQA-KG(ours)	**58.56**	**45.31**	**58.69**	**71.00**	**66.63**	**67.88**	**61.34**

Table 3: Results of three methods on the CK12-QA test dataset. The prediction accuracy (%) of the three methods under the two question types of ND T/F and ND MC are reported.

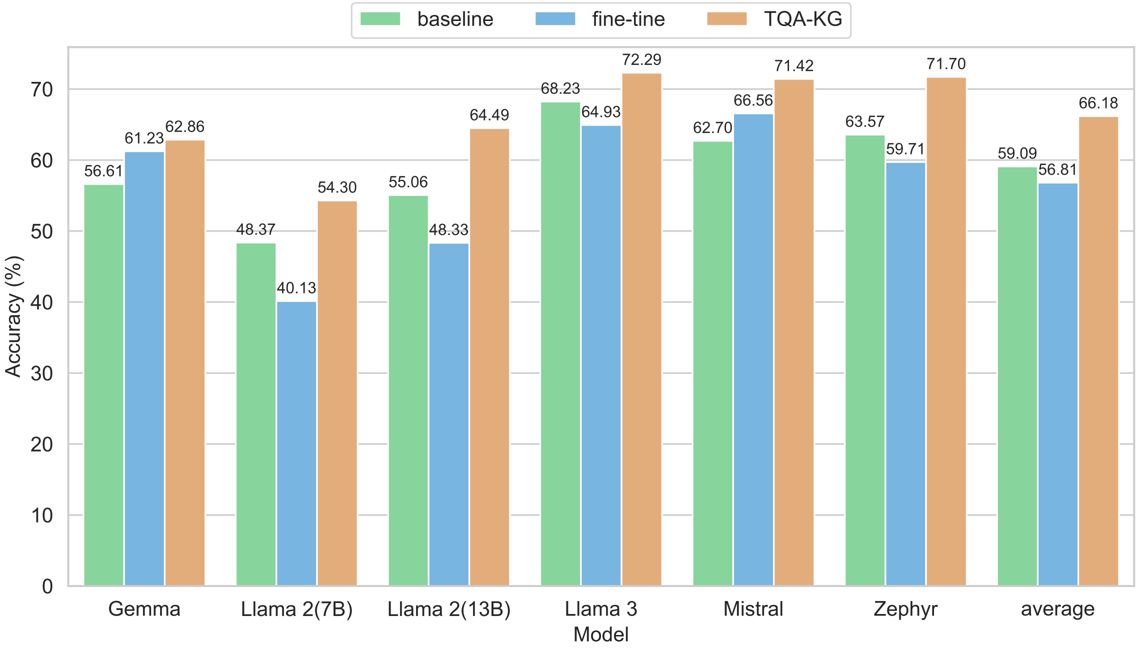


Figure 5: The results on the CK12-QA test data of three methods are reported. The TQA-KG method consistently yields the best results across all six models.

accuracy by 9.43%, significantly enhancing the quality of task completion for the TQA tasks. In Table 4, compared to previous works that utilized small-scale neural networks, the TQA-KG method achieved the best performance, demonstrating both the superior applicability of large models in this task domain and the effectiveness of the TQA-KG approach.

Work	Years	Validation	Test
MemN (Kembhavi et al., 2017)	2017	38.83	-
F-GCN (Kim et al., 2019)	2018	54.75	-
ISAAQ (Gomez-Perez and Ortega, 2020)	2020	71.76	72.13
XTQA (Ma et al., 2023a)	2020	41.32	41.67
RAFR (Ma et al., 2023b)	2021	43.35	41.03
DDGNet (Wang et al., 2023)	2023	41.62	41.96
TQA-KG(ours)	2024	**74.56**	**72.29**

Table 4: The results of the prediction accuracy (%) on the CK12-QA dataset of our method and other works.

5. Conclusion and Future Work

This article presents a novel methodology termed TQA-KG, which effectively integrates Large Language Models with Knowledge Graphs. This synergistic approach leverages the computational power of LLMs to facilitate the construction of KGs and subsequently employs these knowledge graphs to refine and enhance the outputs of the LLMs. This dual-faceted strategy addresses the critical need for heightened accuracy in TQA tasks and broadens the scope of applying integrated frameworks of large models and knowledge graphs in complex problem-solving scenarios. Looking ahead, our future research endeavors will focus on further exploring the potential of knowledge graphs to augment the capabilities of large models, enhancing their utility across a broader spectrum of applications.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Garima Agrawal, Tharindu Kumarage, Zeyad Alghami, and Huan Liu. Can knowledge graphs reduce hallucinations in llms?: A survey. *arXiv preprint arXiv:2311.07914*, 2023.

AI Anthropic. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1, 2024.

Jinheon Baek, Alham Fikri Aji, and Amir Saffari. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering. *arXiv preprint arXiv:2306.04136*, 2023.

Salvatore Carta, Alessandro Giuliani, Leonardo Piano, Alessandro Sebastian Podda, Livio Pompianu, and Sandro Gabriele Tiddia. Iterative zero-shot llm prompting for knowledge graph construction. *arXiv preprint arXiv:2307.01128*, 2023.

Jose Manuel Gomez-Perez and Raul Ortega. Isaaq–mastering textbook questions with pre-trained transformers and bottom-up and top-down attention. *arXiv preprint arXiv:2010.00562*, 2020.

Himanshu Gupta, Saurabh Arjun Sawant, Swaroop Mishra, Mutsumi Nakamura, Arindam Mitra, Santosh Mashetty, and Chitta Baral. Instruction Tuned Models are Quick Learners. *arXiv preprint arXiv:2306.05539*, 2023. doi: 10.48550/ARXIV.2306.05539. URL https://arxiv.org/abs/2306.05539.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models. *arXiv preprint arXiv:2106.09685*, 2021.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of Hallucination in Natural Language Generation. *ACM Computing Surveys*, 55(12):248:1–248:38, 2023. ISSN 0360-0300. doi: 10.1145/3571730. URL https://doi.org/10.1145/3571730.

Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A Diagram is Worth a Dozen Images. *Computer Vision – ECCV 2016*, 9908: 235–251, 2016. doi: 10.1007/978-3-319-46493-0_15. URL http://link.springer.com/10.1007/978-3-319-46493-0_15.

Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. Are You Smarter Than a Sixth Grader? Textbook Question Answering for Multimodal Machine Comprehension. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5376–5384. IEEE, 2017. ISBN 978-1-5386-0457-1. doi: 10.1109/CVPR.2017.571. URL http://ieeexplore.ieee.org/document/8100054/.

Daesik Kim, Seonhoon Kim, and Nojun Kwak. Textbook Question Answering with Multimodal Context Graph Understanding and Self-supervised Open-set Comprehension. In Anna Korhonen, David Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3568–3584. Association for Computational Linguistics, 2019. doi: 10.18653/v1/P19-1347. URL https://aclanthology.org/P19-1347.

Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N. Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, and Christian Bizer. DBpedia – A large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web*, 6(2):167–195, 2015. ISSN 15700844. doi: 10.3233/SW-140134. URL https://www.medra.org/servlet/aliasResolver?alias=iospress&doi=10.3233/SW-140134.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.

Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Yudan Liu, Kaikai Ge, Xu Zhang, and Leyu Lin. Real-time Attention Based Look-alike Model for Recommender System. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, KDD '19, pages 2765–2773. Association for Computing Machinery, 2019. ISBN 978-1-4503-6201-6. doi: 10.1145/3292500.3330707. URL https://doi.org/10.1145/3292500.3330707.

Jie Ma, Qi Chai, Jun Liu, Qingyu Yin, Pinghui Wang, and Qinghua Zheng. XTQA: Span-Level Explanations for Textbook Question Answering. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–11, 2023a. ISSN 2162-2388. doi: 10.1109/TNNLS.2023.3294991. URL https://ieeexplore.ieee.org/document/10191037.

Jie Ma, Jun Liu, Yaxian Wang, Junjun Li, and Tongliang Liu. Relation-Aware Fine-Grained Reasoning Network for Textbook Question Answering. *IEEE Transactions on Neural Networks and Learning Systems*, 34(1):15–27, 2023b. ISSN 2162-2388. doi: 10.1109/TNNLS.2021.3089140. URL https://ieeexplore.ieee.org/abstract/document/9466370.

Lars-Peter Meyer, Johannes Frey, Kurt Junghanns, Felix Brei, Kirill Bulert, Sabine Gründer-Fahrer, and Michael Martin. Developing a scalable benchmark for assessing large language models in knowledge graph engineering. *arXiv preprint arXiv:2308.16622*, 2023.

Yixin Nie, Songhe Wang, and Mohit Bansal. Revealing the Importance of Semantic Retrieval for Machine Reading at Scale. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *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 2553–2566. Association for Computational Linguistics, 2019. doi: 10.18653/v1/D19-1258. URL https://aclanthology.org/D19-1258.

Yansong Ning and Hao Liu. Urbankgent: A unified large language model agent framework for urban knowledge graph construction. *arXiv preprint arXiv:2402.06861*, 2024.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. 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 Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, pages 27730–27744. Curran Associates Inc., 2024. ISBN 978-1-71387-108-8.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*, 2023.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional Attention Flow for Machine Comprehension. *ArXiv*, 2016. URL https://www.semanticscholar.org/

paper/Bidirectional-Attention-Flow-for-Machine-Seo-Kembhavi/3a7b63b50c64f4ec3358477790e84cbd6be2a0b4.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Heung-Yeung Shum, and Jian Guo. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *arXiv preprint arXiv:2307.07697*, 2023.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

Milena Trajanoska, Riste Stojanov, and Dimitar Trajanov. Enhancing knowledge graph construction using large language models. *arXiv preprint arXiv:2305.04676*, 2023.

Yaxian Wang, Bifan Wei, Jun Liu, Qika Lin, Lingling Zhang, and Yaqiang Wu. Spatial-semantic collaborative graph network for textbook question answering. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022.

Yaxian Wang, Jun Liu, Jie Ma, Hongwei Zeng, Lingling Zhang, and Junjun Li. Dynamic dual graph networks for textbook question answering. *Pattern Recognition*, 139:109441, 2023. ISSN 0031-3203. doi: 10.1016/j.patcog.2023.109441. URL https://www.sciencedirect.com/science/article/pii/S0031320323000961.

Yilin Wen, Zifeng Wang, and Jimeng Sun. Mindmap: Knowledge graph prompting sparks graph of thoughts in large language models. *arXiv preprint arXiv:2308.09729*, 2023.

Fangzhi Xu, Qika Lin, Jun Liu, Lingling Zhang, Tianzhe Zhao, Qi Chai, Yudai Pan, Yi Huang, and Qianying Wang. MoCA: Incorporating domain pretraining and cross attention for textbook question answering. *Pattern Recognition*, 140:109588, 2023. ISSN 0031-3203. doi: 10.1016/j.patcog.2023.109588. URL https://www.sciencedirect.com/science/article/pii/S0031320323002893.

Liang Yao, Jiazhen Peng, Chengsheng Mao, and Yuan Luo. Exploring large language models for knowledge graph completion. *arXiv preprint arXiv:2308.13916*, 2023.

Shuang Yu, Tao Huang, Mingyi Liu, and Zhongjie Wang. BEAR: Revolutionizing Service Domain Knowledge Graph Construction with LLM. In Flavia Monti, Stefanie Rinderle-Ma, Antonio Ruiz Cortés, Zibin Zheng, and Massimo Mecella, editors, *Service-Oriented Computing*, Lecture Notes in Computer Science, pages 339–346. Springer Nature Switzerland, 2023. ISBN 978-3-031-48421-6. doi: 10.1007/978-3-031-48421-6_23.

Zhou Yu, Jun Yu, Yuhao Cui, Dacheng Tao, and Qi Tian. Deep Modular Co-Attention Networks for Visual Question Answering. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6274–6283. IEEE, 2019. ISBN 978-1-72813-293-8. doi: 10.1109/CVPR.2019.00644. URL https://ieeexplore.ieee.org/document/8953581/.

Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Dongzhan Zhou, et al. Chemllm: A chemical large language model. *arXiv preprint arXiv:2402.06852*, 2024.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models. *arXiv preprint arXiv:2403.13372*, 2024. URL http://arxiv.org/abs/2403.13372.