Graph Reasoning Transformers for Knowledge-Aware Question Answering

Ruilin Zhao[1,3], Feng Zhao[1*], Liang Hu[2], Guandong Xu[3]

[1]Natural Language Processing and Knowledge Graph Lab, School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, China
[2]College of Electronic and Information Engineering, Tongji University, Shanghai, China
[3]Data Science and Machine Intelligence Lab, University of Technology Sydney, Sydney, Australia
{ruilinzhao,zhaof}@hust.edu.cn, lianghu@tongji.edu.cn, guandong.xu@uts.edu.au

Abstract

Augmenting Language Models (LMs) with structured knowledge graphs (KGs) aims to leverage structured world knowledge to enhance the capability of LMs to complete knowledge-intensive tasks. However, existing methods are unable to effectively utilize the structured knowledge in a KG due to their inability to capture the rich relational semantics of knowledge triplets. Moreover, the modality gap between natural language text and KGs has become a challenging obstacle when aligning and fusing cross-modal information. To address these challenges, we propose a novel knowledge-augmented question answering (QA) model, namely, Graph Reasoning Transformers (GRT). Different from conventional node-level methods, the GRT serves knowledge triplets as atomic knowledge and utilize a triplet-level graph encoder to capture triplet-level graph features. Furthermore, to alleviate the negative effect of the modality gap on joint reasoning, we propose a representation alignment pretraining to align the cross-modal representations and introduce a cross-modal information fusion module with attention bias to enable cross-modal information fusion. Extensive experiments conducted on three knowledge-intensive QA benchmarks show that the GRT outperforms the state-of-the-art KG-augmented QA systems, demonstrating the effectiveness and adaptation of our proposed model.

Introduction

Pretrained language models (LMs) have been widely applied to various downstream natural language processing (NLP) tasks due to their impressive natural language understanding and generation performance (Radford et al. 2018a,b). However, the reliance on parametric knowledge leads LMs to generate answers with factual errors when addressing questions that demand current or domain-specific knowledge, as LMs are based on occasionally obsolete or inaccurate parametric knowledge for reasoning. This leads to the fact that large LMs (LLMs) (Chowdhery et al. 2022; Brown et al. 2020) with few-shot prompting still lag behind small fine-tuned state-of-the-art models on knowledge-aware question answering (QA) tasks such as CommonsenseQA and OpenbookQA, even when using elaborate prompting strategies

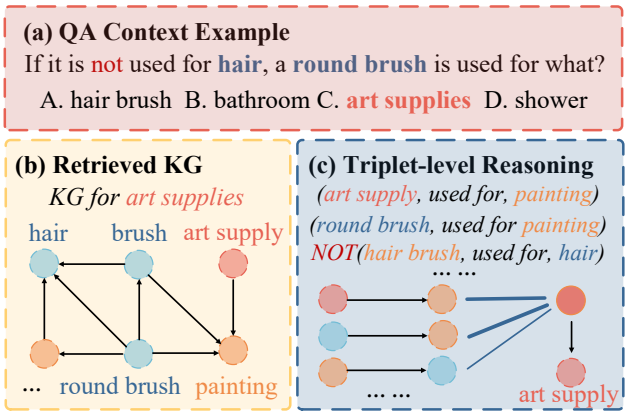

Figure 1: (a) An example of QA context from a knowledge-aware QA dataset. (b) We follow the procedure of Yasunaga et al. (2021) to retrieve the relevant KG. (c) Knowledge-aware QA places emphasis on aggregating knowledge triplets that support the answer.

(Wei et al. 2022). Moreover, LLM hallucination greatly affects their performance in knowledge-intensive QA tasks that require high accuracy and interpretability, as they usually generate answers that sound reasonable but lack interpretability (Ji et al. 2023). In this context, the augmentation of LMs with knowledge graphs (KGs) has gained widespread attention.

Since vanilla LMs are "blind" to other modalities in natural language text, previous works (Yasunaga et al. 2021; Zhang et al. 2022) tended to use graph neural networks (GNNs) to encode KGs. These methods use the GNN message passing method to aggregate information from the neighbours of each node and combine node and language features for answer prediction. While GNNs have been widely used to handle graph structures in downstream tasks such as node classification, node clustering, and link prediction, recent works have shown that GNNs are unable to fully utilize external knowledge graphs in knowledge-aware QA tasks. Jiang et al. (2022) surprisingly found that even when reducing the dimensionality of the GNN node embeddings from 1024 to 1, the performance of KG-augmented QA systems was still unchanged. Wang et al. (2022) leveraged

*Corresponding author

SpareVD to detect which ingredients of a GNN are crucial for knowledge augmentation and found that even by ablating the node embedding and message passing methods, the performance of KG-augmented QA systems could surprisingly be improved. Although these works have investigated these phenomena in knowledge-aware QA tasks, they did not provide reasonable explanations or solutions to address these limitations. As shown in Figure 1, the knowledge-aware QA task is not a node-level classification task. Before modeling KGs using GNNs, traditional rule-based KG-augmented QA systems (Galárraga et al. 2013; Ortona, Meduri, and Papotti 2018) regard knowledge triplets as atomic formulae and view the graph reasoning process as a combination of multiple atomic formulae. They are more concerned with designing rules to aggregate the knowledge triplets that support the answer, while the node information is replaced by serial numbers to distinguish between different entities. In other words, knowledge-aware QA tasks focus on retrieving triplet-level relational facts as evidence to support the reasoning process rather than node-level features.

In addition, some challenges related to the modality gap between text and KGs still exist. Due to the differences between various data structures and encoding methods, a distribution gap can be observed for the same entity between natural language text and a KG, which greatly limits the model's ability to align and utilize both sources of data (Ye et al. 2023). Furthermore, recently proposed KG-augmented QA systems (Yasunaga et al. 2021; Wang et al. 2022) simply concatenate the language and pooled node representations and use a classifier to directly perform answer prediction, without deep interaction and fusion between the two modalities. This results in the graph reasoning module being insensitive to linguistic features such as 'negation' that play important roles in QA.

To address the above challenges, we propose a novel KG-augmented QA system called Graph Reasoning Transformers (GRT), which can be applied to knowledge-aware QA tasks. The structure of the GRT consists of two separate uni-modal encoders for natural language text and KGs, followed by a cross-modal information fusion module. **To address the challenge of node-level graph modeling**, we serve knowledge triplets as atomic knowledge and utilize a triplet-level graph encoder to capture relational knowledge features for joint reasoning. **To address the challenge of the modality gap**, we propose a representation alignment pretraining, which consists of text-triplet matching (TTM) and masked language modeling (MLM), to pretrain uni-modal encoders for aligning the representations across modalities. Furthermore, we propose a information fusion module with elaborate attention bias to establish connections across different modalities, enabling cross-modal information fusion between the QA contexts and KGs.

In summary, the contributions of our work are as follows.

- We propose a novel KG-augmented QA model GRT, which provide a new perspective that utilizes triplets as atomic knowledge for augmenting LMs with structured world knowledge for knowledge-aware QA.
- We propose a triplet-level graph encoder to capture "relational" knowledge features for joint reasoning, thus allowing the given relational knowledge to be fully utilized for knowledge-aware QA tasks.
- We propose a representation alignment pretraining to align the cross-modal representations during pretraining and introduce a information fusion layer with elaborate attention bias to enable cross-modal information fusion between the language and KG during fine-tuning.
- We achieve state-of-the-art results on two commonsense QA benchmarks and a biomedical QA benchmark that heavily rely on knowledge-aware reasoning, demonstrating the ability of the GRT to capture uni-modal features and utilize cross-modal information for joint reasoning.

Related Works

LM-Based QA Systems. Pretrained LMs have been shown to learn a substantial amount of in-depth knowledge from a wide variety of sources during pretraining (Petroni et al. 2019). Roberts, Raffel, and Shazeer (2020) directly fine-tuned an LM on downstream tasks and demonstrated that LMs can answer questions with high accuracy in closed-book settings. In addition, some works aim to enhance the parametric knowledge of LMs. Sun et al. (2020a) and Sun et al. (2020b) proposed a knowledge-enhanced pretraining method to inject world knowledge into the parameters of LMs during pretraining. Other works aim to activate parametric knowledge for downstream tasks. Gardner et al. (2021); Hwang et al. (2021) fine-tuned LMs on link prediction tasks to activate the world knowledge stored in the LMs. Huang et al. (2022) utilized QA contexts to generate topic-related clues, which served as prompts for activating parametric knowledge. However, the reliance on parametric knowledge limits their performance on knowledge-intensive tasks that require up-to-date or domain-specific knowledge.

KG-Augmented QA Systems. GNNs are commonly used to model external KGs for conducting joint reasoning with LMs. Some works (Feng et al. 2020; Yasunaga et al. 2021; Jiang et al. 2022) use the information of one modality to augment another modality. The most representative work, the QA-GNN (Yasunaga et al. 2021), adds language representation as a new node to the retrieved KG and employs an elaborate GNN to jointly update the LM and GNN representations via message passing. Since these methods ignore the modality gap between text and KGs, the distribution gap results in inconsistencies when combining cross-modal representations and finally leads to suboptimal interactions between cross-modal information. Other works tend to use two-tower models for jointly modeling text and graphs (Zhang et al. 2022; Wang et al. 2022; Ye et al. 2023). They encode language and KGs via LMs and GNNs, respectively, and fuse the language and knowledge representations in the final layer for answer prediction. However, the information exchanges between LMs and GNNs are limited, and the information fusion is shallow. Therefore, how to align and fuse cross-modal information remains an important open question.

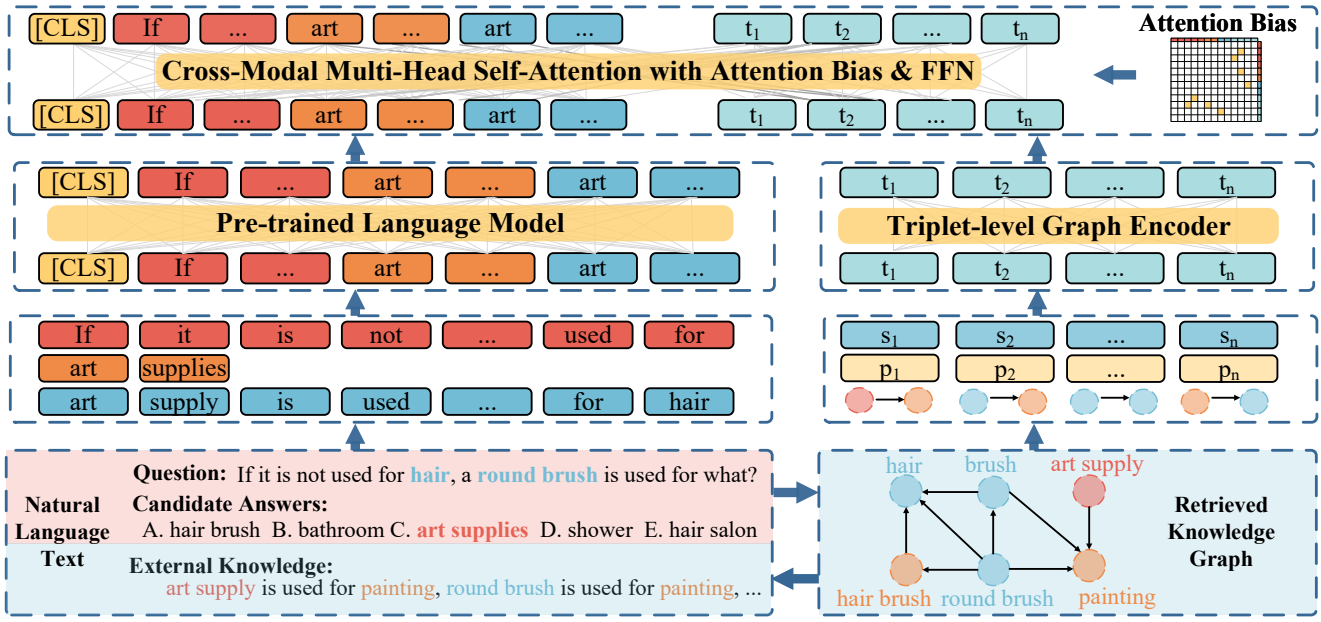

Figure 2: An overview of the graph reasoning transformers. The GRT consists of two separate uni-modal encoders for text and KGs, followed by a cross-modal information fusion module. To better utilize structured relational knowledge, we first serve knowledge triplets as atomic knowledge and propose a triplet-level graph encoder for knowledge encoding (Section 3.1). To address the challenge of the modality gap, we then propose a representation alignment pretraining to pretrain uni-modal encoders to align the representations across modalities (Section 3.2). Finally, we establish cross-modal connections and propose a cross-modal information fusion with attention bias, enabling information fusion across modalities (Section 3.3)).

Graph Reasoning Transformers

In this section, we propose a novel knowledge-augmented QA model named GRT for knowledge-aware QA. We explain our model from the following aspects: the triplet-level graph encoder, the representation alignment pretraining, and the cross-modal information fusion process.

Relational Knowledge Feature via Triplet-Level Graph Encoder

Vanilla LMs are blind to the modality of natural language text, which is not conducive to the augmentation of structured world knowledge. Therefore, the previous methods (Yasunaga et al. 2021; Zhang et al. 2022) are devoted to using GNNs to obtain graph features, where each node aggregates information from its neighbours. However, knowledge-aware QA is more concerned with identifying which knowledge triplets can support the answer, while the node information can be replaced by serial numbers to distinguish between different entities (Galárraga et al. 2013). As a result, the initial node embeddings are demonstrated to be dispensable, and some GNN layers are shown to be over-parameterized (Jiang et al. 2022; Wang et al. 2022). Different from the existing node-based models, we use knowledge triplets as atomic knowledge for graph encoding and propose a triplet encoder for graph encoding.

Semantic Triplet Embedding. As mentioned above, we serve triplets as the atomic knowledge for graph encoding and thus convert the given KG to a set of knowledge triplets. Here, we use the types of entities and relations to initialize the semantic triplet embeddings. The type of an entity represents the location of the entity in the QA context. If an entity is mentioned in the question/answer context, it is denoted as a question/answer entity. Otherwise, the entity is denoted as an other entity. Therefore, the semantic triplet embeddings can be expressed as follows:

$$h_{hrt} = f_s(\tau_h, \tau_r, \tau_o), \tag{1}$$

where τ_h and τ_o are one-hot representations of the head and tail types, and τ_r is a one-hot representation of the relation.

Spatial Position Embedding. In a KG, the connections between entities and relations are represented by the graph structure. However, when linearizing the KG, the inherent graph connectivity will inevitably be disrupted. To maintain the structural information of the KG, we transform the physical connectivity of the knowledge triplets in a graph structure into virtual connectivity in the embedding space. Here, we utilize a pretrained knowledge graph embedding approach called TransE (Bordes et al. 2013) to uniquely locate the spatial positions of the knowledge triplets. TransE is a knowledge graph embedding model that represents entities and relations as continuous vectors in a low-dimensional embedding space. In other words, TransE maps entities into an embedding space and models the relations between entities as translations in the embedding space. Therefore, we can use TransE to uniquely locate the positions of entities and relations in the embedding space, which can be ex-

pressed as follows:

$$p_h = \text{TransE}(h), p_t = \text{TransE}(t), \tag{2}$$

$$p_r = p_h - p_t, \tag{3}$$

$$p_{hrt} = f_p(p_h, p_r, p_t), \tag{4}$$

By utilizing the entity and relation embeddings in the embedding space, a knowledge triplet can be uniquely located.

Triplet Encoder. After obtaining the semantic triplet and spatial position embeddings, we concatenate the two types of embeddings into the input representation t and use a triplet encoder to encode the knowledge triplets:

$$\{t^l_{\text{GLS}}, t^l_1, ...t^l_J\} = \text{Tri-Enc}(\{t^{l-1}_{\text{GLS}}, t^{l-1}_1, ...t^{l-1}_J\}). \tag{5}$$

Specifically, we leverage the triplet-level graph encoder to fuse the representations and utilize the final hidden states for cross-modal information fusion and joint reasoning.

Representation Alignment Pretraining based on Cross-Modal Correspondence

Due to the differences among the utilized data sources and encoding methods, a distribution gap is observed between text and KGs, which greatly limits the model's ability to utilize both sources of data (Ye et al. 2023). To alleviate the distribution gap between text and graphs, we propose a representation alignment pretraining based on cross-modal correspondence to align cross-modal representations.

Text-Triplet Matching. First, we use a contextual LM (Liu et al. 2019) as the backbone to encode the QA context and texutalized world knowledge:

$$\{h^l_{CLS}, h^l_1, ...h^l_I\} = \text{PLM}(\{h^{l-1}_{CLS}, h^{l-1}_1, ...h^{l-1}_I\}). \tag{6}$$

For the text entity mentioned in the QA context, we use the last hidden states corresponding to the entity as the text entity representation. If the entity contains multiple tokens, we apply mean pooling over all tokens of the entity to obtain the entity representation. For each knowledge triplet mentioned in the knowledge graph, we use the corresponding output representation in the final layer of the triplet encoder as the knowledge representation.

As the KG is obtained by entity linking (Yasunaga et al. 2021), the entities mentioned in the QA context are associated with the knowledge triplets in the KG. Therefore, we can easily utilize this **cross-modal correspondence** for pretraining. For a text entity mentioned in the QA context, we select the knowledge triplet containing this entity as a positive text-triplet pair; otherwise, the triplet is described as a negative pair. Then, we concatenate the representations of the text entity and knowledge triplet of a text-triplet pair and use an MLP to calculate the probability y that indicates whether the text entity is matched to the triplet.

For each QA context, we select k_1 positive text-triplet pairs and k_2 negative pairs for pretraining. If these text-triplet pairs are positive pairs, their labels $\hat{y}$ are assigned to 1; otherwise, they are assigned to 0. Finally, the text-triplet matching loss can be expressed as follows:

$$\mathcal{L}_{TTM} = -\frac{1}{k_1 + k_2} \sum_{i=1}^{k_1+k_2} y_i \log \hat{y_i} + (1 - y_i) \log(1 - \hat{y_i}). \tag{7}$$

Masked Language Modeling. Structured world knowledge cannot be directly used by vanilla LMs. When encoding the input text sequence, LMs rely solely on their own parametric knowledge for text encoding, which can be heavily limited by factors such as knowledge availability, timeliness, and biases. However, the relational facts in the KG can provide rich contextual information to the input sequence and background knowledge to the text entities to enhance the overall performance of text encoding.

To incorporate structured knowledge into the text encoding process, we first use predefined templates to convert knowledge triplets to natural language text. However, KGs contain vast numbers of knowledge triplets. Including all of these relational facts during text encoding would introduce knowledge noise and potentially hinder the text encoding performance. To mitigate this issue, we use a pretrained sentence transformer (Reimers and Gurevych 2019) to calculate the similarity scores of all knowledge triplets and the QA context and finally select the top-20 sentences as clues.

Finally, we concatenate the QA context and the clues as the input sequence. We follow the procedure of the official BERT (Devlin et al. 2019) model to randomly select 15% of the tokens for masked language modeling and calculate the masked language modeling loss $\mathcal{L}$. The overall pretraining loss can be expressed as follows:

$$\mathcal{L} = \mathcal{L}_{TTM} + \mathcal{L}_{MLM}. \tag{8}$$

Masked language modeling with external knowledge enhances the ability of LMs to utilize external knowledge during text encoding and can effectively prevent the negative model distribution effect during contrastive learning (e.g., catastrophic forgetting).

Cross-Modal Information Fusion with Attention Bias

Previous works tend to fuse cross-modal information in a shallow way, such as incorporating the text representation as a new node into the KG (Yasunaga et al. 2021) or using an MLP between uni-modal encoders for cross-modal information fusion (Zhang et al. 2022). However, these lead to inconsistency when combining the cross-modal representations and finally lead to suboptimal cross-modal feature aggregation results (Park et al. 2023). Therefore, our goal is to enable the cross-modal information fusion module to better utilize the complementary advantages of different modalities to improve the effectiveness of joint reasoning.

Cross-Modal Connection. During information fusion, we aim to enhance the cross-modal interactions between text entities and their corresponding knowledge triplets. A text entity may undergo grammatical changes or contain multiple words. For instance, after entity linking, the text entity "car shows" is linked to the KG entities "car" and "show" simultaneously. In this case, we aim to establish a cross-modal connection between the text entity token "shows" and the corresponding KG entity "show".

To this end, we use a pretrained word embedding to calculate the similarity between the text words and the KG entities to establish cross-modal connections. For each knowledge triplet, we first obtain the representations of the head

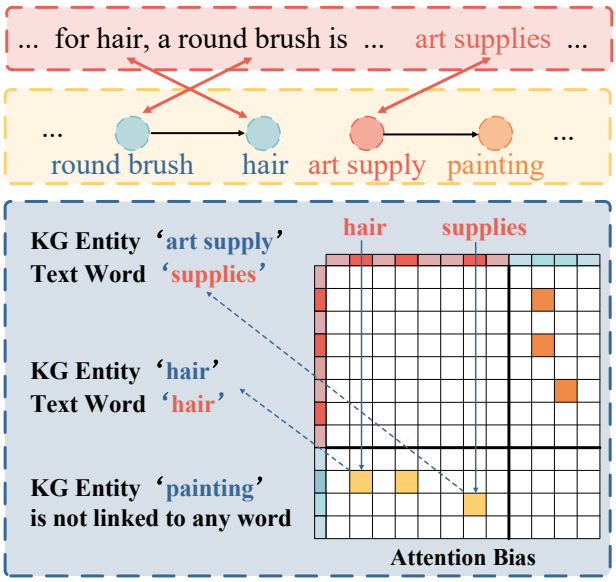

Figure 3: Attention bias for information fusion. We establish connections between the head/tail entity and the words with the highest similarity and utilize this connection matrix to construct attention bias.

and tail entities. Then, we obtain the word representations of all text entities and calculate the representation similarities between the words and the head/tail entity. As shown in Figure 3, for each knowledge triplet, we establish connections between the head/tail entity and the words with the highest similarity. Note that some entities in the KG are obtained by retrieving the 2-hop neighbourhoods of the linked entities (i.e., other entities). Therefore, we only retain the connections where the KG entity is mentioned in the input sequence to avoid introducing attention noise between the tokens and unrelated knowledge triplets. Finally, we can obtain the connection matrix $\hat{X}$.

Attention Bias. After obtaining the connection matrix, we construct an attention bias matrix that enables the connected cross-modal tokens to obtain higher attention weights when calculating attention in the cross-modal fusion layer. The attention bias matrix can be expressed as follows:

$$\Omega = \begin{bmatrix} 0 & \omega_1 \hat{X} \\ \omega_2 \hat{X}^T & 0 \end{bmatrix} \tag{9}$$

Since a knowledge triplet contains a head entity and a tail entity, we use different biases ω_1 and ω_2 to initialize the attention bias matrix for distinguishing the direction of cross-modal attention.

Cross-Modal Information Fusion. In the cross-modal information fusion module, we stack M transformer blocks as the backbone and modify the multi-head attention block to adopt the original transformer for fusing cross-modal information. Our cross-modal multi-head attention mechanism can be expressed as follows:

$$\mathbf{Q} = XW_Q, \mathbf{K} = XW_K, \mathbf{V} = XW_V, \tag{10}$$

$$\text{Attention}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \text{softmax}(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d}} + \Omega)\mathbf{V}. \tag{11}$$

After the cross-modal information module, we use an MLP to calculate the confidence score of the candidate answer. Formally, given a question q and a candidate answer c, the confidence score of candidate answer c can be expressed as follows:

$$\text{p}(c|q) = \exp(\text{MLP}([h_{\text{CLS}}; h_1; ...; h_I])), \tag{12}$$

where $[h_{\text{CLS}}; h_1; ...; h_I]$ is the output hidden states of cross-modal information fusion layers. We leverage the final hidden states for answer prediction. We use the cross-entropy loss during training and predict the candidate answer with the maximum probability as the answer prediction.

Experiments

Experimental Setup

In this section, we empirically evaluate the performance and adaptability of our proposed GRT model.

Datasets. We employ 3 challenging knowledge-intensive QA benchmarks that heavily rely on knowledge-aware reasoning, including 2 commonsense QA benchmarks *CommonsenseQA* and *OpenbookQA* and a biomedical QA benchmark *MedQA-USMLE*.

- *CommonsenseQA*: This is a commonsense QA dataset with 12247 questions that require knowledge-aware reasoning based on commonsense knowledge. Each question provides 5 candidate answers. Due to the unavailability of official data splits, we adopt the in-house (IH) split (Lin et al. 2019) used in prior studies for evaluations.
- *OpenbookQA*: This is a commonsense QA dataset with 5957 questions that require commonsense knowledge and elementary-level science knowledge for reasoning. Each question provides 4 candidate answers. Note that OpenbookQA additionally provides 5,167 crowd-sourced commonsense knowledge facts, which can be used as external knowledge for joint reasoning. In this work, we utilize the official data splits (Mihaylov and Frank 2018) for our evaluations.
- *MedQA-USMLE*: This is a medical-domain QA dataset with 12723 questions that require biomedical and clinical knowledge for reasoning. Each question provides 4 candidate answers. We use MedQA-USMLE mainly to verify the adaptability of our proposed GRT. In this work, we utilize the official data splits (Jin et al. 2020) for evaluation purposes.

Baseline Methods. We compare our proposed GRT with existing KG-augmented QA systems.

- *RGCN*, *GconAttn*, *RN*: These classic works use relation-aware GNNs to encode KGs for joint reasoning.
- *MHGRN*, *SAFE*, *GSC*, *QAT*: These methods further investigate the effect of edge modeling on graph reasoning.
- *QA-GNN*, *GreaseLM*, *FIT*: These works mainly focus on establishing cross-modal interactions and understanding.

Models	IHdev-Acc	IHtest-Acc
RoBERTa-large (w/o KG)	73.0 (±0.4)	68.6 (±0.5)
GconAttn (Wang et al. 2019)	72.6 (±0.3)	68.5 (±0.9)
RN (Santoro et al. 2017)	74.5 (±0.9)	69.0 (±0.2)
MHGRN (Feng et al. 2020)	74.4 (±0.1)	71.1 (±0.8)
QA-GNN (Yasunaga et al. 2021)	76.5 (±0.2)	73.4 (±0.9)
SAFE (Jiang et al. 2022)	-	74.0
GreaseLM (Zhang et al. 2022)	78.5 (±0.5)	74.2 (±0.4)
GSC (Wang et al. 2022)	79.1 (±0.2)	74.4 (±0.4)
QAT (Park et al. 2023)	79.5 (±0.4)	75.4 (±0.3)
FIT (Ye et al. 2023)	78.5 (±0.5)	75.6 (±0.3)
GRT (ours)	**79.6**(±0.3)	**76.1**(±0.4)

Table 1: Performance comparison with fine-tuned LMs and KG-augmented QA systems on CommonsenseQA.

Models	w/o Facts	w Facts
Fine-tuned LM (w/o KG)	64.8 (±2.3)	78.4 (±1.6)
GconAttn (Wang et al. 2019)	64.7 (±1.4)	71.8 (±1.2)
RN (Santoro et al. 2017)	65.2 (±1.1)	75.3 (±1.3)
MHGRN (Feng et al. 2020)	66.8(±1.1)	80.6 (±1.4)
QAGNN (Yasunaga et al. 2021)	67.8 (±2.7)	82.7 (±1.5)
SAFE (Jiang et al. 2022)	69.2	87.1
GreaseLM (Zhang et al. 2022)	-	84.8
GSC (Wang et al. 2022)	70.3 (±0.8)	86.6 (±0.4)
QAT (Park et al. 2023)	71.2 (±0.8)	86.9 (±0.2)
FIT (Ye et al. 2023)	70.1 (±1.0)	86.0 (±0.4)
GRT (ours)	**72.6** (±1.0)	**87.3** (±0.8)

Table 2: Performance comparison with fine-tuned LMs and KG-augmented QA systems on OpenbookQA.

Implementation Details. For CommonsenseQA and OpenbookQA, we adopt the vanilla contextual LM, *RoBERTa-large* (Liu et al. 2019), as our LM backbone to validate the performance of KG-augmented QA systems. Specifically, we follow previous works and use *AristoRoBERTa* (Clark et al. 2020) to combine the QA context with its corresponding commonsense knowledge facts for evaluation. To evaluate the domain generality of the method, we adopt a biomedical-domain LM, *SapBERT-Base*, as our LM backbone for MedQA-USMLE and compare it with fine-tuned biomedical LMs, *ClinicalBERT* and *BioBERT*, for further analysis. Our code is available at https://github.com/HUSTNLP-codes/GRT

Main Results

Table 1 summarizes the overall results of our experiments on the CommonsenseQA. Our proposed GRT achieves the best performance of 76.1%, which is an improvement of 4.1% over the fine-tuned LM. In addition, the GRT outperforms the existing state-of-the-art GNN-oriented QA systems such as GSC (Wang et al. 2022) and GreaseLM (Zhang et al. 2022), indicating the effectiveness of the triplet-level graph encoder for joint reasoning.

On the OpenbookQA benchmark, as shown in Figure 2, our proposed GRT also achieves a state-of-the-art performance level of 72.6%, which is an improvement of 11.4% over the vanilla RoBERTa-large baseline (i.e., **w/o Facts**). Moreover, we conduct experiments using AristoRoBERTa (i.e., **w Facts**) to combine the crowd-sourced commonsense knowledge facts, which are provided by OpenbookQA, for further evaluations. Our proposed GRT achieves the best result of 87.3% and outperform other KG-augmented QA systems. These results demonstrate the effectiveness of our representation alignment pretraining.

Models	Test-Acc. (%)
ClinicalBERT (Huang et al. 2019)	32.4
BioRoBERTa-base (Lee et al. 2020)	36.1
BioBERT-basse (Lee et al. 2020)	34.1
BioBERT-large (Lee et al. 2020)	36.7
SapBERT-base (w/o KG) (Liu et al. 2021)	37.2
QAGNN (Yasunaga et al. 2021)	38.0
GreaseLM (Zhang et al. 2022)	38.5
GSC (Wang et al. 2022)	39.3
QAT (Park et al. 2023)	39.3
FIT (Ye et al. 2023)	39.0
GRT (ours)	**39.5**

Table 3: Test accuracy comparison on MedQA-USMLE with biomedical LMs and KG-augmented QA systems.

Graph Feature	CSQA	OBQA	OBQA w/ facts
Node	73.1	68.4	84.2
Node + GNN	74.1	68.8	85.4
Triplet	**76.1**	**72.6**	**87.3**
Graph Feature	**Negations**	**Entities < 7**	**Entities > 7**
Node	74.3	75.1	79.6
Node + GNN	75.6	75.8	79.1
Triplet	**79.7**	**79.1**	**80.0**

Table 4: Performance comparison with node-level graph features and triplet-level graph features.

Domain Generality

Med-USMLE. In addition to commonsense-domain QA benchmarks, we also conduct experiments on a biomedical-domain benchmark to further investigate the adaptation of our model. Table 3 shows the experimental results obtained on the Med-USMLE benchmark. Our proposed GRT achieves a state-of-the-art result of 39.5% and outperform other pretrained LMs from the biomedical domain. Compared with other QA systems possessing shallow information fusion, our proposed information fusion can effectively improve the capability of GRT to fuse cross-modal information. This result demonstrates the effective adaptability and domain generality of our proposed model.

Empirical Analyses

To further analyse each component of our proposed GRT, we mainly focus on 3 research questions as follows.

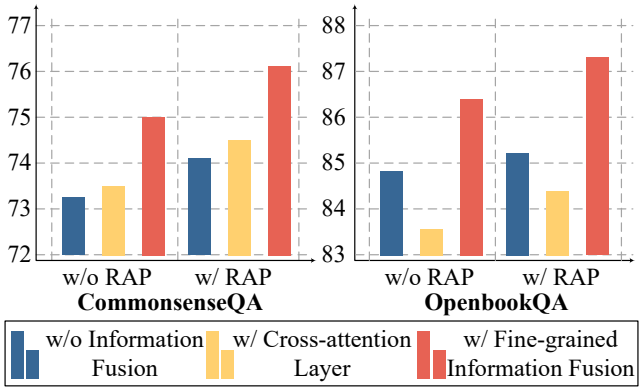

Figure 4: Ablation studies on the representation alignment pretraining (RAP). We additionally compare with 3 different joint reasoning methods for further evaluations.

- **Research Question 1:** Is the triplet-level graph encoder more effective than node-level encoding methods?
- **Research Question 2:** Does representation alignment improve the effectiveness of joint reasoning?
- **Research Question 3:** How does attention bias influence the performance of information fusion?

Effectiveness of the Triplet-Level Encoder. To answer Research Question 1, we compare node-level graph features, including the raw input node embeddings (Node) and the node representations obtained through a GNN (Node+GNN). The results in Table 4 show that the triple-level graph feature significantly outperforms the node-level graph feature, demonstrating its ability to enhance the knowledge-aware QA task. Moreover, we conduct experiments on distinct question types for a further evaluation. The triplet-level graph features achieve the best performances of 79.7%, 79.1%, and 80.0% on all question types. The significant improvements achieved for questions with negations and fewer than 7 entities demonstrate that triple-level graph features effectively handle negations and utilize KGs to supplement the background knowledge for text entities.

Ablation Studies. To answer Research Question 2, we conduct ablation studies on the representation alignment pretraining (RAP) and compare it with different joint reasoning methods, including method without information fusion, using a cross-attention layer without explicit supervision, and information fusion with attention bias. The results in Figure 4 show that RAP yields significantly improved performance across all methods on both benchmarks, demonstrating that representation alignment can effectively alleviate the negative impact of the modality gap and enhance the model's ability to utilize cross-modal information. Furthermore, the performance of the cross-attention layer, which is widely used in multimodal models, is lower than that of the method without information fusion on OpenbookQA. The main reason for this is the insufficient corpora utilized to train the cross-attention layer since large multimodal models require pretraining on massive multimodal corpora. However, cross-modal information fusion with attention bias still exhibits stable performance, indicating that the attention bias can guide the information fusion to utilize cross-modal information.

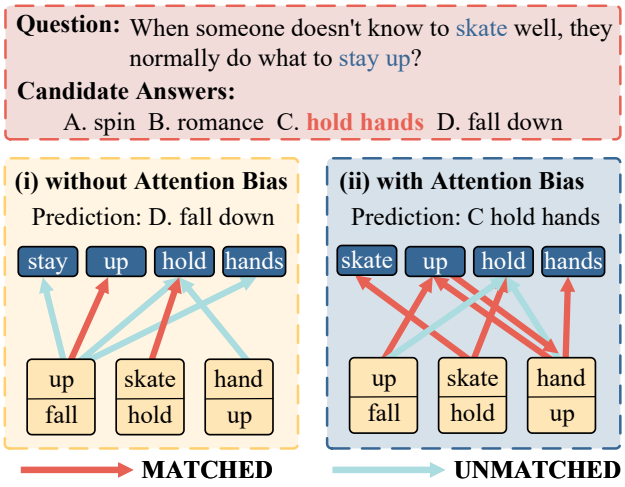

Figure 5: The effectiveness of attention bias for information fusion. The direction of an arrow indicates the direction of cross-modal attention.

Qualitative Analyses. To answer Research Question 3, we visualize the produced attention maps to demonstrate how attention bias improves the cross-modal information fusion process. As shown in Figure 5, the tokens above are language entity tokens, while those below are knowledge triplets. To better observe the attention changes, we only show relatively high attention values between tokens and triplets. After introducing the attention bias, the language tokens tend to receive high attention values from their corresponding knowledge triplets, and vice versa. For instance, the text token 'skate' receives the highest attention value from the 'up-antonym-fall' triplet, but after introducing the attention bias, 'skate' tends to attract more attention from the corresponding 'skate-related-hold' triplet. Additionally, as we only establish connections between the triplets and the entities mentioned in the text sequence, the attention bias never causes the other text tokens to be linked to unrelated triplets. This avoids introducing attention noise and enables effective information fusion across modalities.

Conclusion

In this work, we propose GRT, a novel KG-augmented QA model for knowledge-aware QA. We utilize triplets as atomic knowledge and propose a novel triplet-level graph encoder to better model structured knowledge. Moreover, we propose a representation alignment pre-training to align cross-modal representations and introduce an information fusion method with attention bias to fuse cross-modal information. Experimental results on 3 knowledge-intensive QA benchmarks demonstrate the effectiveness and adaptation of our proposed model to utilize cross-modal information.

Acknowledgments

This work was supported in part by the National Key R&D Program of China under Grant 2023YFF0905503, National Natural Science Foundation of China under Grants No.62072203, No.62072257 and the Australian Research Council Under Grants DP22010371, LE220100078.

References

Bordes, A.; Usunier, N.; García-Durán, A.; Weston, J.; and Yakhnenko, O. 2013. Translating Embeddings for Modeling Multi-relational Data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, 2787–2795.

Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A.; and et al. 2020. Language Models are Few-Shot Learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Chowdhery, A.; Narang, S.; Devlin, J.; Bosma, M.; Mishra, G.; Roberts, A.; Barham, P.; Chung, H. W.; Sutton, C.; Gehrmann, S.; and et al. 2022. PaLM: Scaling Language Modeling with Pathways. *CoRR*, abs/2204.02311.

Clark, P.; Etzioni, O.; Khot, T.; Khashabi, D.; Mishra, B. D.; Richardson, K.; Sabharwal, A.; Schoenick, C.; Tafjord, O.; Tandon, N.; Bhakthavatsalam, S.; Groeneveld, D.; Guerquin, M.; and Schmitz, M. 2020. From 'F' to 'A' on the N.Y. Regents Science Exams: An Overview of the Aristo Project. *AI Magazine*, 41(4): 39–53.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 4171–4186.

Feng, Y.; Chen, X.; Lin, B. Y.; Wang, P.; Yan, J.; and Ren, X. 2020. Scalable Multi-Hop Relational Reasoning for Knowledge-Aware Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 1295–1309.

Galárraga, L. A.; Teflioudi, C.; Hose, K.; and Suchanek, F. M. 2013. AMIE: association rule mining under incomplete evidence in ontological knowledge bases. In Schwabe, D.; Almeida, V. A. F.; Glaser, H.; Baeza-Yates, R.; and Moon, S. B., eds., *22nd International World Wide Web Conference, WWW '13, Rio de Janeiro, Brazil, May 13-17, 2013*, 413–422. International World Wide Web Conferences Steering Committee / ACM.

Gardner, M.; Merrill, W.; Dodge, J.; Peters, M. E.; Ross, A.; Singh, S.; and Smith, N. A. 2021. Competency Problems: On Finding and Removing Artifacts in Language Data. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 1801–1813.

Huang, Z.; Wu, A.; Zhou, J.; Gu, Y.; Zhao, Y.; and Cheng, G. 2022. Clues Before Answers: Generation-Enhanced Multiple-Choice QA. In Carpuat, M.; de Marneffe, M.; and Ruíz, I. V. M., eds., *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics*, 3272–3287.

Hwang, J. D.; Bhagavatula, C.; Bras, R. L.; Da, J.; Sakaguchi, K.; Bosselut, A.; and Choi, Y. 2021. Comet-Atomic 2020: On Symbolic and Neural Commonsense Knowledge Graphs. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, 6384–6392.

Ji, Z.; Lee, N.; Frieske, R.; Yu, T.; Su, D.; Xu, Y.; Ishii, E.; Bang, Y.; Madotto, A.; and Fung, P. 2023. Survey of Hallucination in Natural Language Generation. *ACM Comput. Surv.*, 55(12): 248:1–248:38.

Jiang, J.; Zhou, K.; Wen, J.; and Zhao, X. 2022. Great Truths are Always Simple: A Rather Simple Knowledge Encoder for Enhancing the Commonsense Reasoning Capacity of Pre-Trained Models. In *Findings of the Association for Computational Linguistics*, 1730–1741.

Jin, D.; Pan, E.; Oufattole, N.; Weng, W.; Fang, H.; and Szolovits, P. 2020. What Disease does this Patient Have? A Large-scale Open Domain Question Answering Dataset from Medical Exams. *CoRR*, abs/2009.13081.

Lee, J.; Yoon, W.; Kim, S.; Kim, D.; Kim, S.; So, C. H.; and Kang, J. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinform.*, 36(4): 1234–1240.

Lin, B. Y.; Chen, X.; Chen, J.; and Ren, X. 2019. KagNet: Knowledge-Aware Graph Networks for Commonsense Reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, 2829–2839.

Liu, F.; Shareghi, E.; Meng, Z.; Basaldella, M.; and Collier, N. 2021. Self-Alignment Pretraining for Biomedical Entity Representations. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*, 4228–4238.

Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; and Stoyanov, V. 2019. RoBERTa: A Robustly Optimized BERT Pretraining Approach. *CoRR*, abs/1907.11692.

Mihaylov, T.; and Frank, A. 2018. Knowledgeable Reader: Enhancing Cloze-Style Reading Comprehension with External Commonsense Knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, 821–832.

Ortona, S.; Meduri, V. V.; and Papotti, P. 2018. Robust Discovery of Positive and Negative Rules in Knowledge Bases. In *34th IEEE International Conference on Data Engineering, ICDE 2018, Paris, France, April 16-19, 2018*, 1168–1179. IEEE Computer Society.

Park, J.; Choi, H. K.; Ko, J.; Park, H.; Kim, J.; Jeong, J.; Kim, K.; and Kim, H. J. 2023. Relation-aware Language-Graph Transformer for Question Answering. *CoRR*, abs/2212.00975.

Petroni, F.; Rocktäschel, T.; Riedel, S.; Lewis, P. S. H.; Bakhtin, A.; Wu, Y.; and Miller, A. H. 2019. Language Mod-

els as Knowledge Bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2463–2473.

Radford, A.; Narasimhan, K.; Salimans, T.; and Sutskever, I. 2018a. Improving language understanding by generative pre-training.

Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; and Sutskever, I. 2018b. Language Models are Unsupervised Multitask Learners.

Reimers, N.; and Gurevych, I. 2019. Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks. In Inui, K.; Jiang, J.; Ng, V.; and Wan, X., eds., *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, 3980–3990. Association for Computational Linguistics.

Roberts, A.; Raffel, C.; and Shazeer, N. 2020. How Much Knowledge Can You Pack Into the Parameters of a Language Model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, 5418–5426.

Santoro, A.; Raposo, D.; Barrett, D. G. T.; Malinowski, M.; Pascanu, R.; Battaglia, P. W.; and Lillicrap, T. 2017. A simple neural network module for relational reasoning. In *Advances in Neural Information Processing Systems*, 4967–4976.

Sun, T.; Shao, Y.; Qiu, X.; Guo, Q.; Hu, Y.; Huang, X.; and Zhang, Z. 2020a. CoLAKE: Contextualized Language and Knowledge Embedding. In *Proceedings of the 28th International Conference on Computational Linguistics*, 3660–3670.

Sun, Y.; Wang, S.; Li, Y.; Feng, S.; Tian, H.; Wu, H.; and Wang, H. 2020b. ERNIE 2.0: A Continual Pre-Training Framework for Language Understanding. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, 8968–8975.

Wang, K.; Zhang, Y.; Yang, D.; Song, L.; and Qin, T. 2022. GNN is a Counter? Revisiting GNN for Question Answering. In *Proceedings of the 10th International Conference on Learning Representations*.

Wang, X.; Kapanipathi, P.; Musa, R.; Yu, M.; Talamadupula, K.; Abdelaziz, I.; Chang, M.; Fokoue, A.; Makni, B.; Mattei, N.; and Witbrock, M. 2019. Improving Natural Language Inference Using External Knowledge in the Science Questions Domain. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 7208–7215.

Wei, J.; Wang, X.; Schuurmans, D.; Bosma, M.; Ichter, B.; Xia, F.; Chi, E. H.; Le, Q. V.; and Zhou, D. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *NeurIPS*.

Yasunaga, M.; Ren, H.; Bosselut, A.; Liang, P.; and Leskovec, J. 2021. QA-GNN: Reasoning with Language Models and Knowledge Graphs for Question Answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics*, 535–546.

Ye, Q.; Cao, B.; Chen, N.; Xu, W.; and Zou, Y. 2023. FiTs: Fine-grained Two-stage Training for Knowledge-aware Question Answering. *CoRR*, abs/2302.11799.

Zhang, X.; Bosselut, A.; Yasunaga, M.; Ren, H.; Liang, P.; Manning, C. D.; and Leskovec, J. 2022. GreaseLM: Graph REASoning Enhanced Language Models. In *Proceedings of the 10th International Conference on Learning Representations*.