Dynamic Heterogeneous-Graph Reasoning with Language Models and Knowledge Representation Learning for Commonsense Question Answering

Yujie Wang[1], Hu Zhang[1,2,*], Jiye Liang[1,2,*], Ru Li[1,2]
1.School of Computer and Information Technology, Shanxi University, Taiyuan, China
2.Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, Shanxi University, Taiyuan, China
init_wang@foxmail.com,{zhanghu,ljy,liru}@sxu.edu.cn

Abstract

Recently, knowledge graphs (KGs) have won noteworthy success in commonsense question answering. Existing methods retrieve relevant subgraphs in the KGs through key entities and reason about the answer with language models (LMs) and graph neural networks. However, they ignore (i) optimizing the knowledge representation and structure of subgraphs and (ii) deeply fusing heterogeneous QA context with subgraphs. In this paper, we propose a dynamic heterogeneous-graph reasoning method with LMs and knowledge representation learning (DHLK), which constructs a heterogeneous knowledge graph (HKG) based on multiple knowledge sources and optimizes the structure and knowledge representation of the HKG using a two-stage pruning strategy and knowledge representation learning (KRL). It then performs joint reasoning by LMs and Relation Mask Self-Attention (RMSA). Specifically, DHLK filters key entities based on the dictionary vocabulary to achieve the first-stage pruning while incorporating the paraphrases in the dictionary into the subgraph to construct the HKG. Then, DHLK encodes and fuses the QA context and HKG using LM, and dynamically removes irrelevant KG entities based on the attention weights of LM for the second-stage pruning. Finally, DHLK introduces KRL to optimize the knowledge representation and perform answer reasoning on the HKG by RMSA. We evaluate DHLK at CommonsenseQA and OpenBookQA, and show its improvement on existing LM and LM+KG methods.

1 Introduction

Question answering (QA) is a challenging task that requires machines to understand questions asked by natural language and respond to the questions based on the knowledge acquired. Recently, QA has made remarkable progress with the development of Language Models (LMs) (Devlin et al., 2019; Liu et al., 2019; Lan et al., 2020; Raffel et al., 2020). Fine-tuning based on LMs has now become a major paradigm for QA tasks. LMs are pre-trained on a general large-scale corpus containing rich world knowledge, which the machine can utilize when fine-tuning downstream tasks using LMs. In some simple, fact-based QA tasks, such as SQuAD (Rajpurkar et al., 2016, 2018) and RACE (Lai et al., 2017), machine has surpassed humans in terms of answer accuracy. However, the machine remains less satisfactory in some structured reasoning QA tasks that require commonsense knowledge.

Commonsense knowledge is the general law summarized by human beings through observation, research, and reflection of various phenomena in the objective world, which is verified by the long-term experience of countless people and is the common daily consensus of people. When humans answer questions, they use this knowledge unconsciously. For example, if you ask "John had an urgent matter to attend to at his company, and he drove fast to the company but stopped at an intersection, what could have happened?". We can reason that John may be passing through the intersection when the traffic light turns red. Thus, he has to stop and wait for the light to turn green. This commonsense reasoning is easy for humans. However, considering that commonsense knowledge is a relatively tacit knowledge, LMs do not capture it well.

Knowledge graphs (KGs) store a large amount of commonsense knowledge that can be used by machines to make sound judgments, and this knowledge can provide the machines with displayed and interpretable evidence. Therefore, some methods (Lin et al., 2019; Feng et al., 2020; Yasunaga et al., 2021; Sun et al., 2022; Zheng and Kordjamshidi, 2022; Zhang et al., 2022) have introduced KGs into LMs-based QA methods to model and reason about structured knowledge in KGs through graph

*Corresponding author.

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 14048–14063

QA context

What **furniture** will you normally find near a **side chair**?

A. bedroom **B. table*** C.wheel barrow

D. building E. office

Knowledge paths and paraphrases

side —relatedto→ bank —relatedto→ table

side —relatedto→ top —usedfor→ table

furniture —madeof→ wood —relatedto→ table

furniture —isa→ desk —relatedto→ table

chair —relatedto→ desk —isa→ table

chair —atlocation→ cat —atlocation→ table

side chair —atlocation→ room —atlocation→ table

side chair —atlocation→ bedroom —atlocation→ table

.....

side chair : a straight-backed chair without arms.

table : a piece of furniture with tableware for a meal laid out on it.

.....

Figure 1: An example from CommonsenseQA , we retrieve knowledge paths from ConceptNet (Speer et al., 2017) and key entity paraphrases from WordNet (Miller, 1995) and Wiktionary.

neural networks (GNNs) (Scarselli et al., 2009). Related methods generally follow the following steps: (i) Extracting key entities in the QA context using entity recognition methods; (ii) Retrieving relevant knowledge subgraphs in KGs based on key entities; (iii) Initializing subgraph entities using pre-trained word embedding models; and (iv) Designing a GNNs-based reasoning module to perform joint reasoning with LMs. Therefore, subgraphs' quality and the joint method of GNNs and LMs are crucial to the reasoning performance.

Currently, combining LMs and GNNs to solve commonsense QA (CQA) task has proven to be an effective method but still contains some problems: (i) In the key entities-based subgraph extraction method, the goodness of the key entities largely determines the quality of the subgraph. As shown in Figure 1, entities such as "wood", "bank", "top", and "cat" are some noisy knowledge for the current question, and they affect the model's judgment during the inference process. But a part of the noisy knowledge can be solved by optimizing key entities. In the example of Figure 1, "side chair" is a noun phrase, which should be considered as a whole when retrieving knowledge based on it, and this will reduce the introduction of some noisy knowledge; (ii) The knowledge representation of entities in subgraph are mostly obtained by Glove (Pennington et al., 2014), LMs, and so on, ignoring the semantic associations between entities; additionally, the knowledge representations obtained are less effective; (iii) Given that the QA context and subgraph have different structures, existing methods encode QA context and subgraph separately, with shallow interactions only at the GNN layer through message passing (Yasunaga et al., 2021; Zheng and Kordjamshidi, 2022) or at the output layer through attention mechanism (Sun et al., 2022) or MLP (Feng et al., 2020; Zhang et al., 2022; Yasunaga et al., 2022), lacking deep fusion of QA context and subgraph, which will hinder the inference capability of the model.

Based on the above problems, we propose a **D**ynamic **H**eterogeneous-graph reasoning method based on **L**anguage models and **K**nowledge representation learning (DHLK). Specifically, given a question and choice, we first use KeyBERT (Grootendorst, 2020) to extract the candidate entities and introduce WordNet (Miller, 1995) and Wiktionary [1] vocabularies to filter the candidate entities and then obtain the key entities, which can remove some noisy entities in the subgraph retrieval process and realize first-stage pruning of the subgraph. We also incorporate the paraphrases of key entities in the two dictionaries as entities into the subgraph to construct a heterogeneous knowledge graph (HKG). Then, we use LM to encode the QA context and HKG and fuse the QA context and HKG in the encoding process. In addition, we dynamically remove irrelevant entities according to the attention weights of LM to achieve the second-stage pruning of the subgraph. Finally, we combine KRL and Relation Mask Self-Attention (RMSA) to optimize the knowledge representation of HKG and incorporate the path information in the HKG into the QA context. In summary, our contributions are threefold:

- We construct the HKG based on multiple knowledge sources and introduce a two-stage pruning strategy and KRL to optimize the structure and knowledge representation of the HKG.
- We effectively fuse the QA context and HKG in the encoding phase of LM to achieve better reasoning performance.
- We evaluate our method on CommonsenseQA and OpenBookQA, proving the effectiveness of the method through a series of ablation experiments and case studies.

[1] https://www.wiktionary.org/

2 Related Work

Recently, large LMs such as UnifiedQA (Khashabi et al., 2020), T5 (Raffel et al., 2020) and GPT-3 (Brown et al., 2020) have been widely applied in QA tasks, such as open-domain question answering (ODQA) and CQA, driving the development of QA. However, larger LMs result in disproportionate resource consumption and training time. Therefore, many works have enhanced the reasoning ability of machines by introducing external knowledge, hoping to achieve good answering results while reducing resource consumption and training time.

Knowledge-enhanced ODQA. ODQA model utilizes external knowledge to answer questions, typically consisting of a retriever and a reader component. With the development of LMs, Retrieval Augmented Architectures (Lewis et al., 2020; Guu et al., 2020) have become the mainstream method for ODQA. They apply LMs to retriever-reader and conduct joint training of the retriever-reader. However, previous works (Karpukhin et al., 2020; Izacard and Grave, 2021) primarily focused on unstructured knowledge sources, such as Wikipedia. Recently, some works (Min et al., 2019; Zhou et al., 2020; Hu et al., 2022) have started incorporating structured KGs into the retriever-reader architecture to enhance retrieval effectiveness and question answering capabilities. For example, UniK-QA (Oguz et al., 2022) converts KG triplets into text and merges them with unstructured knowledge repositories. KG-FiD (Yu et al., 2022) utilizes KG to establish relational dependencies between retrieved paragraphs and employs GNNs to sort and prune the retrieved paragraphs. Grape (Ju et al., 2022) constructs a localized bipartite graph for each pair of question and article, learning knowledge representations through GNNs.

Knowledge-enhanced CQA. CQA also requires external knowledge to answer questions, but it is more focused on commonsense questions. From the perspective of knowledge and QA context fusion, there are currently two main methods. Some works (Bian et al., 2021; Xu et al., 2021, 2022) feed the retrieved knowledge together with the QA context into the LM, utilizing self-attention to fuse the knowledge. However, the self-attention treats the input knowledge and QA context indiscriminately, which can undermine the semantic information of the QA context. Other works (Lin et al., 2019; Feng et al., 2020; Lv et al., 2020; Yasunaga et al., 2022; Zheng and Kordjamshidi, 2022) combine LM and GNNs to solve CQA. For example, QAGNN (Yasunaga et al., 2021) uses LM to estimate the importance of subgraph entities and considers the QA context as an additional node connected to the subgraph. JointLK (Sun et al., 2022) uses the bidirectional attention module to fuse the two modalities while designing a pruning module to remove irrelevant entities from the subgraph. GREASELM (Zhang et al., 2022) fuses encoding representations from LM and GNNs through multi-layered modality interaction operations. However, these works encode the QA context and KG subgraph in isolation, leading to limited interaction between textual and KG representations. Additionally, they don't consider the influence of key entities and knowledge representations on subgraph retrieval and model inference.

In contrast to previous works, we propose to reduce noisy knowledge by optimizing the set of key entities in the subgraph retrieval process. In addition, we use LM to encode and fuse the two modalities and prune the subgraph according to the attention weights of LM. Meanwhile, during the inference process, we introduce the KRL algorithm to optimize the knowledge representation of the subgraph. Figure 2 shows the overall architecture of our method.

3 Methods

3.1 Task Formulation

We focus on the multi-choice CQA task in this paper. Given a question q and a set of candidate choices $\{c_1, c_2, ..., c_b\}$, we need to select the one that best fits the question's meaning. In general, CQA does not provide the background knowledge related to the question. Therefore, we need to retrieve relevant knowledge from KG and combine it to reason about the answer. In this paper, we retrieve a relevant subgraph from ConceptNet based on key entities in question and choice, and identify the paraphrases of the key entities in WordNet and Wiktionary. Meanwhile, we explicitly take the paraphrases as some additional entities (paraphrase entities) connected to the KG subgraph. Therefore, our method starts with the HKG construction.

3.2 HKG Construction

In the KG-based CQA task, the subgraph needs to be retrieved from the KG based on key entities. Therefore, the key entities determine the quality of the subgraph. We use KeyBERT to identify

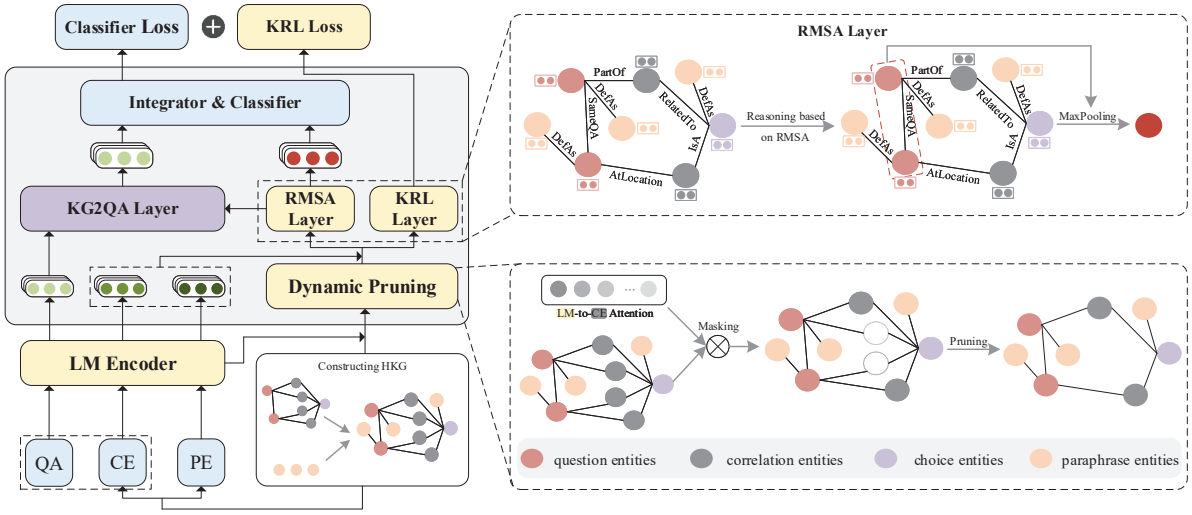

Figure 2: The overall architecture of our proposed DHLK model, which takes as input the QA context (question + choice) and the entities in the HKG. The CE and PE denote concept entities extracted from ConceptNet and paraphrase entities extracted from the dictionaries, respectively.

candidate entities $\hat{E} = \{\hat{e}_1, \hat{e}_2, ..., \hat{e}_n\}$ in question and choice. Meanwhile, we identify phrase entities in $\hat{E}$ based on WordNet and Wiktionary vocabularies, and remove the subwords that constitute phrase entities in $\hat{E}$, to obtain key entities $E = \{e_1, e_2, ..., e_m\}$ and their corresponding paraphrases $P = \{p_1, p_2, ..., p_m\}$. Here n, m denotes the number of candidate entities and key entities, and $n \geq m$.

Following the work of Yasunaga et al. (2021), we retrieve the subgraph in ConceptNet according to E. The subgraph consists of multiple knowledge paths within two-hops, and each path contains at most two triples. Meanwhile, we separately connect the question key entities and choice key entities in the subgraph, and define the relation between them as "SameQA". In addition, we consider P as paraphrase entities and connect them with the corresponding key entities to construct HKG, and define the relation between them as "DefAs". We give all the relations included in HKG in Appendix A. From the knowledge source perspective, HKG contains two types of entities, i.e., concept entities and paraphrase entities.

3.3 LM-Based Encoding

Inspired by K-BERT (Liu et al., 2020), we construct two visible matrices and use RoBERTa (Liu et al., 2019) to encode the QA context, concept entities, and paraphrase entities in HKG. The visible matrix and the encoding process are described further below.

We connect the QA context with the concept entities and construct the visual matrix M according to the following rules:

(i) The tokens contained in the QA context are visible to each other.

(ii) The tokens belonging to the same concept entity are visible to each other.

(iii) The key entities exist in the concept entities, and they are also extracted from the QA context. Therefore, the key entities and the corresponding tokens in the QA context are visible to each other.

The value of $M_{i,j}$ is 0 or 1, where $M_{i,j} = 1$ means that tokens are visible to each other, and $M_{i,j} = 0$ means that tokens are invisible to each other. In RoBERTa model, M is further defined as

$$\tilde{M} = \begin{cases} 0 & M_{i,j} = 1 \\ -\infty & M_{i,j} = 0 \end{cases} \quad (1)$$

Based on $\tilde{M}$, we introduce Mask Self-Attention (MSA) into RoBERTa to encode the QA context and concept entities. Formally, the MSA is defined as

$$\boldsymbol{Q}^{i+1}, \boldsymbol{K}^{i+1}, \boldsymbol{V}^{i+1} = \boldsymbol{h}^i W_q, \boldsymbol{h}^i W_k, \boldsymbol{h}^i W_v \quad (2)$$

$$s^{i+1} = \frac{\boldsymbol{Q}^{i+1} \boldsymbol{K}^{i+1\top}}{\sqrt{d}} \quad (3)$$

$$\alpha^{i+1} = softmax(s^{i+1} + \tilde{M}) \quad (4)$$

$$\boldsymbol{h}^{i+1} = s^{i+1} \boldsymbol{V}^{i+1} \quad (5)$$

where $\boldsymbol{h}^i$ is the hidden state of RoBERTa at i-th layer. W_q, W_k and W_v are trainable model parameters. α^{i+1} is the attention weights after integrating $\tilde{M}$. d denotes the hidden layer size of RoBERTa.

We feed the QA context, concept entities, and M into RoBERTa to obtain the tokens embeddings of the QA context and concept entities: $\{\tilde{\boldsymbol{q}}_i\}_{i=1}^{A} \in \mathbb{R}^d$ and $\{\tilde{\boldsymbol{c}}_i\}_{i=1}^{Z} \in \mathbb{R}^d$. Here A and Z denote the number of tokens of QA context and concept entities, respectively.

Similarly, we construct a visible matrix $\hat{M}$ to prevent the change in paraphrases meaning due to the interaction between different paraphrases. In $\hat{M}$, only the tokens located in the same paraphrase are visible to each other. We connect all the paraphrases and feed them into RoBERTa along with $\hat{M}$ to obtain the tokens embeddings $\{\tilde{\boldsymbol{p}}_i\}_{i=1}^{F} \in \mathbb{R}^d$ of the paraphrase entities. Here F denotes the number of paraphrase tokens.

3.4 Dynamic Pruning

Although we prune the HKG by filtering key entities during its construction, noisy entities persist in the HKG. Therefore, we prune the HKG in the second-stage according to the importance of the concept entities to the QA context.

We take the embedding representation $\tilde{\boldsymbol{q}}_1$ of the [CLS] position in RoBERTa as semantic representation of the QA context. For the concept entities in HKG, we obtain the token-level attention weights $w = \{w_j, w_{j+1}, ..., w_k\}$ of each entity for $\tilde{\boldsymbol{q}}_1$ by equation 3, and then obtain the node-level attention weight $\tilde{w}$ by

$$\hat{w} = \frac{1}{k}\sum_{i=j}^{k} w_i \tag{6}$$

$$\tilde{w} = \frac{\hat{w} - \hat{w}_{min}}{\hat{w}_{max} - \hat{w}_{min}} \tag{7}$$

where $\hat{w}_{max}$, $\hat{w}_{min}$ are the maximum and minimum values of node-level attention weights. Next, we remove the entities with $\tilde{w} < \mu$ in the HKG and remove the edges connected to these entities in the HKG.

3.5 KRL Layer

HKG can be viewed as the knowledge subgraph composed of multiple triples connections. To obtain better entity and relation embeddings, we introduce KRL to optimize knowledge representation and improve the reasoning effect.

Entity and relation embeddings. For a triplet (h, r, t), h,t are the entities in HKG, and r is the concatenated edge between the entities. Based on the tokens embeddings $\{\tilde{\boldsymbol{t}}_i\}_{i=1}^{T} \in \mathbb{R}^d$ of each entity obtained in Section 3.3, we obtain the entity embedding $\tilde{\boldsymbol{e}}$ by

$$\tilde{\boldsymbol{e}} = W_t f_{avg}(\{\tilde{\boldsymbol{t}}_1, \tilde{\boldsymbol{t}}_2, ..., \tilde{\boldsymbol{t}}_T\}) \tag{8}$$

where $W_t \in \mathbb{R}^{d \times d_t}$ is a linear transformation, f_{avg} is an average pooling function. Similarly, we feed all the relations and corresponding paraphrases into RoBERTa to obtain the relation embedding $\tilde{\boldsymbol{r}}$ by equation 8.

For simplicity, we follow TransE (Bordes et al., 2013), combined with a negative sampling strategy to optimize entity and relation embeddings. TransE training objective is

$$\mathcal{L}_{KRL} = \sum_{(h,r,t)\in S} \sum_{(h',r,t')\in S'_{(h,r,t)}} \left[\gamma + d_r(h,t) - d_r(h^{'}, t^{'})\right] \tag{9}$$

$$d_r(h,t) = \|h + r - t\|_p \tag{10}$$

where $\gamma > 0$ is a margin hyperparamet, d_r is the scoring function, we take the norm p as 1, and S' is the samples obtained by negative sampling. For the negative sampling strategy, we randomly sample entity in other HKGs in the same batch to replace the head entity or tail entity.

3.6 RMSA Layer

Inspired by (Wang et al., 2020a; Shao et al., 2020), we introduce the relation into Mask Self-Attention to construct RMSA and combine LM and RMSA for reasoning.

First, we separately obtain the initial embedding representation $\boldsymbol{E}^0 = \{\tilde{\boldsymbol{e}}_i\}_{i=1}^{V} \in \mathbb{R}^{d_t}$ and $\boldsymbol{R}^0 = \{\tilde{\boldsymbol{r}}_i\}_{i=1}^{B} \in \mathbb{R}^{d_t}$ of all entities and the relations between entities in HKG by Section 3.5. Here V and B denote the number of entities and relations, respectively. Then, we apply L-layer RMSA to update the embedding representations of entities and relations in HKG. Specifically, the computation process of the l-th layer RMSA can be formulated as

$$\tilde{\alpha}^{l-1} = (\boldsymbol{E}^{l-1}W_q^e)(\boldsymbol{E}^{l-1}W_k^e + \boldsymbol{R}^{l-1}W_k^r)^\top \tag{11}$$

$$\alpha^{l-1} = softmax(\tilde{\alpha}^{l-1}/\sqrt{d_t} + M_{hkg}) \tag{12}$$

$$\tilde{\boldsymbol{E}}^{l-1} = \alpha^{l-1}(\boldsymbol{E}^{l-1}W_v^e + \boldsymbol{R}^{l-1}W_v^r) \tag{13}$$

$$\boldsymbol{E}^{l} = LayerNorm(\tilde{\boldsymbol{E}}^{l-1}) \tag{14}$$

where W_q^e, W_k^e, W_v^e, W_k^r and W_v^r are trainable model parameters, M_{hkg} is the adjacency matrix of HKG after pruning.

We obtain the HKG graph embedding representation $\tilde{\boldsymbol{g}}$ by

$$\tilde{\boldsymbol{g}} = f_{max}(\tilde{\boldsymbol{e}}^q) \tag{15}$$

where f_{max} is maximum pooling function, $\tilde{\boldsymbol{e}}^q$ is all question entities embeddings.

3.7 Integrator & Answer Prediction

After L-layer RMSA iteration, we obtain the entities and relations embeddings in HKG. Then, we incorporate the path information of HKG into the QA context by a KG2QA layer and then connect it with the $\tilde{\boldsymbol{g}}$ to predict the answer.

KG2QA. HKG is composed of multiple paths $X = \{x_1, x_2, ..., x_y\}$, each of which is a sequence of multiple triples. Same as Lin et al. (2019), we define the k-th path between the i-th question entity $e_i^q \in E_q$ and the j-th choice entity $e_j^c \in E_c$ as

$$X_{i,j}[k] = [(e_i^q, r_0, t_0), ...(t_{n-1}, r_n, e_j^c)] \tag{16}$$

We use GRU to encode X and use the last hidden layer state as X's embedding representation $\tilde{\boldsymbol{X}}$.

Not all paths are helpful for answering questions, so we dynamically select the appropriate paths by the relevance between the paths and the QA context. First, we compute the similarity score s^{pq} between the paths and QA context through the cosine similarity algorithm. Then, we retain top $\beta\%$ of the knowledge paths $\tilde{\boldsymbol{X}}^q$ according to s^{pq}. Finally, we obtain the QA context representation $\tilde{\boldsymbol{Q}}^p$ of the fusion paths information by

$$s_{pq} = softmax\left((\tilde{\boldsymbol{q}}W_q^q)(\tilde{\boldsymbol{X}}^q W_k^p)^\top\right) \tag{17}$$

$$\hat{\boldsymbol{Q}}^p = LayerNorm(s_{pq}\tilde{\boldsymbol{X}}^q W_v^p + \tilde{\boldsymbol{q}}) \tag{18}$$

$$\tilde{\boldsymbol{Q}}^p = f_{avg}(\hat{\boldsymbol{Q}}^p) \tag{19}$$

Here $\tilde{\boldsymbol{q}}$ is the tokens embeddings of the QA context, W_q^q, W_k^p and W_v^p are trainable model parameters.

Finally, we feed $\tilde{\boldsymbol{g}}$, $\tilde{\boldsymbol{Q}}^p$ and $\tilde{\boldsymbol{Q}}^q$ into the MLP to predict the answer probability.

$$p = MLP([\tilde{\boldsymbol{g}}; \tilde{\boldsymbol{Q}}^p; \tilde{\boldsymbol{Q}}^q]) \tag{20}$$

Here $\tilde{\boldsymbol{Q}}^q$ is obtained by averaging the pooling of $\tilde{\boldsymbol{q}}$.

4 Experiment

4.1 Datasets

We evaluate our method on CommonsenseQA (Talmor et al., 2019) and OpenBookQA (Mihaylov et al., 2018). Given that the test set of CommonsenseQA is not public, we conduct experiments on the in-house dataset (IHdata) splitted by (Lin et al., 2019) (specific details of the datasets are in Appendix B).

4.2 Implementation Details

For the CQA tasks, we use two types of external knowledge: knowledge graph and dictionary. Given a question and choice, we extract at most 100 knowledge paths within two-hops in ConceptNet (Speer et al., 2017) based on the question key entities and the choice key entities. We also retrieve the paraphrases of the key entities in WordNet (Miller, 1995) and Wiktionary. In the experiment, we use RoBERTa-large (Liu et al., 2019) as the encoder and Adamw (Loshchilov and Hutter, 2019) as the model optimizer. For some hyperparameters, we set the learning rate to 1e-5, the batch size to $\{4, 5\}$, the epochs to $\{3, 6\}$, RMSA's layer number L=4, dynamic pruning threshold μ=0.38, and knowledge path's retention rate β=40%. Each model is trained using one GPU (NVIDIA_A100), which takes 1.5 hours on average.

4.3 Compared Method

We compare with the mainstream RoBERTa-large+KG methods, including RN (Santoro et al., 2017), RGCN (Schlichtkrull et al., 2018), GconAttn (Wang et al., 2019), KagNet (Lin et al., 2019), MHGRN (Feng et al., 2020), QAGNN (Yasunaga et al., 2021), JointLK (Sun et al., 2022), DRGN (Zheng and Kordjamshidi, 2022), GREASELM (Zhang et al., 2022) and DRAGON (Yasunaga et al., 2022). Meanwhile, we compare our method with DESC-KCR (Xu et al., 2021), which also uses both KG and dictionary types of knowledge. But since DESC-KCR uses ALBEER-xxlarge (Lan et al., 2020) as the encoder, we re-trained the DESC-KCR model in IHdata using RoBERTa-large (Liu et al., 2019) for a fair comparison.

4.4 Main results

Table 1 and Table 2 give the experimental results on CommonsenseQA and OpenBookQA. On both datasets, our method achieves consistent

Methods	IHdev-Acc.(%)	IHtest-Acc.(%)
Fine-tuned LMs (w/o KBs)	73.07 (±0.45)	68.69 (±0.56)
+ RGCN	72.69 (±0.19)	68.41 (±0.66)
+ GconAttn	72.61 (±0.39)	68.59 (±0.96)
+ KagNet	73.47 (±0.22)	69.01 (±0.76)
+ RN	74.57 (±0.91)	69.08 (±0.21)
+ MHGRN	74.45 (±0.10)	71.11 (±0.81)
+ QA-GNN	76.54 (±0.21)	73.41 (±0.92)
+ DESC-KCR	78.21(±0.23)	73.78 (±0.39)
+ DGRN	78.20	74.00
+ GREASELM	78.5(±0.5)	74.20(±0.4)
+ JointLK	77.88 (±0.25)	74.43 (±0.83)
+ DRAGON *	-	**76.00**
+ DRAGON (w/o MLM) *	-	73.80
+ DHLK (ours)	**79.39** (±0.24)	74.68 (±0.26)

Table 1: Performance comparison on Commonsense QA in-house split. The DRAGON model undergoes MLM training on the BookCorpus dataset and requires training on 8 A100 GPUs for 7 days. Meanwhile, our method outperforms DRAGON when the MLM task is removed.

Methods	RoBERTa	AristoRoBERTa
Fine-tuned LMs (w/o KB)	64.80 (±2.37)	78.40 (±1.64)
+ RGCN	62.45 (±1.57)	74.60 (±2.53)
+ GconAttn	64.75 (±1.48)	71.80 (±1.21)
+ RN	65.20 (±1.18)	75.35 (±1.39)
+ MHGRN	66.85 (±1.19)	80.60
+ QAGNN	70.58 (±1.42)	82.77 (±1.56)
+ DESC-KCR *	-	-
+ DGRN	69.60	84.10
+ GREASELM	-	84.80
+ JointLK	70.34 (±0.75)	84.92 (±1.07)
+ DRAGON	72.00	-
+ DRAGON (w/o MLM)	66.40	-
+ DHLK (ours)	**72.20** (±0.40)	**86.00** (±0.79)

Table 2: Accuracy on the test set of OpenBookQA. Methods with AristoRoBERTa use the textual evidence by Clark et al. (2020) as an additional input to the QA context. DESC-KCR does not provide pre-processed data from OpenBookQA. Therefore we cannot train the DESC-KCR model on OpenBookQA.

improvements compared to fine-tuned LM and other LM+KG methods. On CommonsenseQA, DHLK improves 6.32% and 5.99% on IHdev and IHtest compared to fine-tuned RoBERTa, respectively. Compared to other LM+KG methods, DHLK has also achieved highly competitive results. (DRAGON further pre-trained on BookCorpus, so it outperforms us on IHtest.) Similarly, our method achieves better experimental results on OpenBookQA. Compared to the best JointLk method, our method improves by 1.08%.

In Tables 3 and 4, we also compare with similar methods in the leaderboard, and our method achieves competitive results.

Methods	Dev-Acc. (%)	Test-Acc.(%)
RoBERTa (Liu et al., 2019)	78.5	72.1
RoBERTa + FreeLB (Zhu et al., 2020)	78.81	72.19
RoBERTa + HyKAS (Ma et al., 2019)	80.1	73.2
RoBERTa + KE	78.7	73.3
Albert (Lan et al., 2020)	80.5	73.5
RoBERTa + KEDGN (ensemble)	-	74.4
XLNet + Graph Reasoning (Lv et al., 2020)	79.3	75.3
RoBERTa + MHGRN (Feng et al., 2020)	-	75.4
ALBERT + Path Generator (Wang et al., 2020b)	78.42	75.6
RoBERTa + QA-GNN (Yasunaga et al., 2021)	-	76.1
Albert (Lan et al., 2020) (ensemble)	-	76.5
RoBERTa + JointLK (Sun et al., 2022)	-	76.6
RoBERTa + DHLK (ours)	**80.85**	**77.6**

Table 3: Performance comparison on Commonsense QA official leaderboard.

Methods	Test-Acc.(%)
Careful Selection (Banerjee et al., 2019)	72.0
AristoRoBERTa	77.8
KF+SIR (Banerjee and Baral, 2020)	80.0
AristoRoBERTa + PG (Wang et al., 2020b)	80.2
AristoRoBERTa + MHGRN (Feng et al., 2020)	80.6
ALBERT + KB	81.0
AristiRoBERTa + QA-GNN (Yasunaga et al., 2021)	82.8
T5 (Raffel et al., 2020)	83.2
AristoRoBERTa + DRGN (Sun et al., 2022)	84.1
AristoRoBERTa + GREASELM (Zhang et al., 2022)	84.8
AristoRoBERTa + JointLK (Sun et al., 2022)	85.6
UnifiedQA(11B)* (Khashabi et al., 2020)	**87.2**
AristoRoBERTa + DHLK (our)	86.8

Table 4: Accuracy on the OpenBookQA leaderboard test set. All listed methods use the provided science facts as an additional input to the language context. The UnifiedQA (11B params) is 30x larger than our model.

5 Analysis

5.1 Ablation Studies

We conduct ablation studies on the Commonsense IHdev set to further analyze the effectiveness of each module of DHLK.

Impact of DHLK module. Table 5(a) shows the experimental results after ablation of each model of DHLK. Disabling the KG2QA module results in a performance decrease of 1.24%, showing that KG2QA can effectively incorporate the paths information from HKG into the QA context. Removing the KRL module results in a 0.83% decrease in the DHLK's performance, demonstrating that optimizing the knowledge representation of HKG by KRL can improve the reasoning ability of the model. Removing the dynamic pruning module results in 0.66% decrease of DHLK's performance, which proves that there is some unfavorable knowledge in HKG for model reasoning. After removing the visible matrix M in the RoBERTa encoding process, the performance decreases by 3.53%. The reason is that when M is removed, all tokens are visible to each other when encoding QA context and concept

Methods	IHdev-Acc.(%)
DHLK	79.61
- KG2QA	78.37
- KRL	78.78
- Dynamic pruning	78.95
- Visual matrix M	76.08
- Paraphrase entities	79.11

(a) Impact of DHLK module

RMSA Layers	IHdev-Acc.(%)
$L = 2$	78.62
$L = 3$	79.27
$L = 4$ (final)	**79.61**
$L = 5$	79.19
$L = 6$	78.86

(b) Impact of RMSA layers

Pruning threshold μ	IHdev-Acc.(%)
$\mu = 0.34$	77.72
$\mu = 0.36$	78.62
$\mu = 0.38$ (final)	**79.61**
$\mu = 0.40$	79.36
$\mu = 0.42$	78.95

(c) Impact of HKG pruning threshold

Retention rate β (%)	IHdev-Acc.(%)
$\beta = 36$	78.70
$\beta = 38$	79.44
$\beta = 40$ (final)	**79.61**
$\beta = 42$	78.38
$\beta = 44$	78.46

(d) Impact of paths retention rate

Table 5: Ablation results on the CommonsenseQA IHdev set.

entities by RoBERTa, too many concept entities can change the original meaning of QA context and also have an impact on dynamic pruning. Finally, removing paraphrase entities from HKG results in a 0.5% performance degradation, which is due to the fact that paraphrase entities can further enhance the semantic representation of key entities.

Impact of RMSA layers. We further analyze the effect of the number of RMSA layers on DHLK. As shown in Table 5(b), the DHLK performance gradually increases as the number of layers increases, and the best performance is achieved when $L = 4$.

Impact of pruning threshold and retention rate. We analyze the thresholds of the dynamic pruning module and the KG2QA module, respectively (see in Table 5(c) and Table 5(d)). For the dynamic pruning module, DHLK achieves the best performance when we remove entities with node-level attention weights less than 0.38 in the HKG. Similarly, for the KG2QA module, DHLK achieves the best performance when we retain the top 40% of the paths most relevant to the QA context.

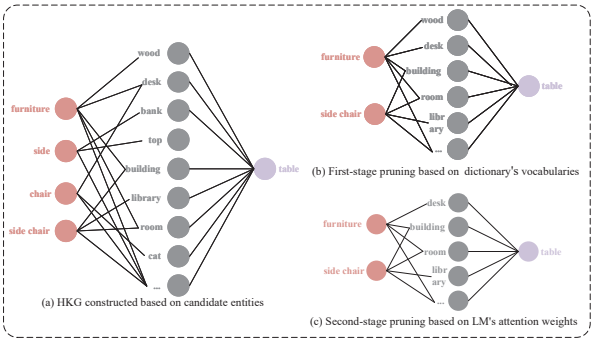

Figure 3: A case study on two-stage pruning. The question and the corresponding answer are "What furniture will you normally find near a side chair?" and "table". For simplicity, we give only part of the entities in the figure and remove the paraphrased entities.

5.2 Case study

We analyze the two-stage pruning strategy of DHLK by a case study. As shown in Figure 3(a), when we extract the subgraph in ConceptNet based on the candidate entities, we introduce some unrelated entities to the question inevitably. For example, "side chair" is a noun phrase and should be considered as a whole. When it is split into "side" and "chair", the "side" has different meanings from it. Meanwhile, "side" and "chair" also introduce some irrelevant entities to the current question, such as "bank", "top", "cat", etc. Therefore, in Figure 3(b) we introduce the dictionary's vocabularies to filter the candidate entities and remove the subwords that make up the phrase entities, so that some irrelevant entities such as "bank", "top" and "cat" can be removed when retrieving the subgraph.

We consider the above process as the first-stage pruning of HKG. However, the subgraph obtained by this process is static and there are still some noisy entities in HKG. We think that the entities that are weakly associated with the QA context should be removed dynamically in the model inference process. Therefore, as shown in Figure 3 (c), in the second-stage pruning, we dynamically remove entities with less relevance to the QA context, e.g., "wood", during the model reasoning based on the LM's attention weights.

5.3 Error Analysis

To further analyze why our model fails on some questions. As shown in Appendix C, we randomly select 50 examples for analysis and classify them into the following classes.

Inappropriate paraphrases. Some entities have multiple paraphrases, even though we extract paraphrases based on entity POS tags in the QA context and the similarity of each paraphrase to the QA context, there are still some entities whose paraphrases are inappropriate. For example, the paraphrase of "fair" in the first example should be "(used of hair or skin) pale or light colored", but the paraphrase we extracted is "a competitive exhibition of farm products", which is inconsistent with the question in the example.

Indistinguishable knowledge paths. When we analyze the error examples, we find that some questions have similar knowledge paths in multiple choices. In such cases, the model predicts answers that are also consistent with human commonsense. For example, in the second example, the

"hedgehog" and "porcupine" have similar knowledge paths and the same paraphrase.

Lack of relevant knowledge. Although we use multiple knowledge sources, there is still much knowledge that is not covered. In the third example, the question is about the content of the self-referential book written by Kramer. This requires some knowledge of Kramer's life to answer, but we did not retrieve this knowledge in ConcetNet or the dictionaries.

Incomprehensible questions. When the question is too long or rather abstract, the model is difficult to make correct judgment. The fourth example asks "The pencil sharpener in the classroom is broken, and the teacher tells the students where they should go to find another.". Although our model retrieves the correct paths and paraphrases, it lacks further understanding of the question and cannot model the current question scenario. The lack of this ability led to our method's unsatisfactory results in answering some complex questions.

6 Conclusions

In this paper, we propose DHLK, a CQA method based on LM and KRL. Our main innovations include: (i) Constructing the HKG based on KG and dictionary, and introducing a two-stage pruning strategy and KRL to optimize the structure and knowledge representation of the HKG; (ii) Deeply fusing the QA context and HKG in the encoding stage of LM, and designing a KG2QA module to incorporate the paths information of HKG into the QA context. The effectiveness of DHLK is demonstrated via experimental results and analysis on CommonsenseQA and OpenBookQA.

Limitations

In this section, we will analyze the limitations of our method. First, we introduce multiple knowledge sources to construct HKG, and encoding this knowledge through LM consumes more GPU resources. Second, some useful knowledge may be removed when retrieving knowledge from key entities optimized by dictionary vocabulary. Then, we experimentally demonstrate that the paraphrase descriptions are effective in improving the reasoning ability of the model, but due to resource constraints, we are unable to incorporate the paraphrases of all entities into HKG. Finally, our method uses the simpler TransE algorithm when optimizing the knowledge representation using KRL due to GPU constraints, which may not be able to model the complex relationships in HKG well.

Acknowledgments

We thank the anonymous reviewers for their helpful comments and suggestions. This work is supported by the National Key Research and Development Program of China (2020AAA0106100), National Natural Science Foundation of China (62176145) and National Natural Science Foundation of China (62076155).

References

Pratyay Banerjee and Chitta Baral. 2020. Knowledge fusion and semantic knowledge ranking for open domain question answering. *CoRR*, abs/2004.03101.

Pratyay Banerjee, Kuntal Kumar Pal, Arindam Mitra, and Chitta Baral. 2019. Careful selection of knowledge to solve open book question answering. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6120–6129. Association for Computational Linguistics.

Ning Bian, Xianpei Han, Bo Chen, and Le Sun. 2021. Benchmarking knowledge-enhanced commonsense question answering via knowledge-to-text transformation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12574–12582. AAAI Press.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 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*, pages 2787–2795.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Peter Clark, Oren Etzioni, Tushar Khot, Daniel Khashabi, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, Niket Tandon, Sumithra Bhakthavatsalam, Dirk Groeneveld, Michal Guerquin, and Michael Schmitz. 2020. From 'f' to 'a' on the N.Y. regents science exams: An overview of the aristo project. *AI Mag.*, 41(4):39–53.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Yanlin Feng, Xinyue Chen, Bill Yuchen Lin, Peifeng Wang, Jun Yan, and Xiang Ren. 2020. Scalable multi-hop relational reasoning for knowledge-aware question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1295–1309, Online. Association for Computational Linguistics.

Maarten Grootendorst. 2020. Keybert: Minimal keyword extraction with bert.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Ziniu Hu, Yichong Xu, Wenhao Yu, Shuohang Wang, Ziyi Yang, Chenguang Zhu, Kai-Wei Chang, and Yizhou Sun. 2022. Empowering language models with knowledge graph reasoning for open-domain question answering. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 9562–9581. Association for Computational Linguistics.

Gautier Izacard and Edouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 874–880. Association for Computational Linguistics.

Mingxuan Ju, Wenhao Yu, Tong Zhao, Chuxu Zhang, and Yanfang Ye. 2022. Grape: Knowledge graph enhanced passage reader for open-domain question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*, pages 169–181. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1896–1907. Association for Computational Linguistics.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard H. Hovy. 2017. RACE: large-scale reading comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*, pages 785–794. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Patrick S. H. 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. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 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, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2829–2839. Association for Computational Linguistics.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-BERT: enabling language representation with knowledge graph. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2901–2908. AAAI Press.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International*

Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019. OpenReview.net.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 8449–8456. AAAI Press.

Kaixin Ma, Jonathan Francis, Quanyang Lu, Eric Nyberg, and Alessandro Oltramari. 2019. Towards generalizable neuro-symbolic systems for commonsense question answering. *CoRR*, abs/1910.14087.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. Can a suit of armor conduct electricity? A new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2381–2391. Association for Computational Linguistics.

George A. Miller. 1995. Wordnet: A lexical database for english. *Commun. ACM*, 38(11):39–41.

Sewon Min, Danqi Chen, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. Knowledge guided text retrieval and reading for open domain question answering. *CoRR*, abs/1911.03868.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Sejr Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. Unik-qa: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1535–1546. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 784–789. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Adam Santoro, David Raposo, David G. T. Barrett, Mateusz Malinowski, Razvan Pascanu, Peter W. Battaglia, and Tim Lillicrap. 2017. A simple neural network module for relational reasoning. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 4967–4976.

Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. 2009. The graph neural network model. *IEEE Trans. Neural Networks*, 20(1):61–80.

Michael Sejr Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *The Semantic Web - 15th International Conference, ESWC 2018, Heraklion, Crete, Greece, June 3-7, 2018, Proceedings*, volume 10843 of *Lecture Notes in Computer Science*, pages 593–607. Springer.

Nan Shao, Yiming Cui, Ting Liu, Shijin Wang, and Guoping Hu. 2020. Is graph structure necessary for multi-hop question answering? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 7187–7192. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 4444–4451. AAAI Press.

Yueqing Sun, Qi Shi, Le Qi, and Yu Zhang. 2022. Jointlk: Joint reasoning with language models and knowledge graphs for commonsense question answering. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 5049–5060. Association for Computational Linguistics.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. Commonsenseqa: A question

answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4149–4158. Association for Computational Linguistics.

Kai Wang, Weizhou Shen, Yunyi Yang, Xiaojun Quan, and Rui Wang. 2020a. Relational graph attention network for aspect-based sentiment analysis. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 3229–3238. Association for Computational Linguistics.

Peifeng Wang, Nanyun Peng, Filip Ilievski, Pedro A. Szekely, and Xiang Ren. 2020b. Connecting the dots: A knowledgeable path generator for commonsense question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4129–4140. Association for Computational Linguistics.

Xiaoyan Wang, Pavan Kapanipathi, Ryan Musa, Mo Yu, Kartik Talamadupula, Ibrahim Abdelaziz, Maria Chang, Achille Fokoue, Bassem Makni, Nicholas Mattei, and Michael Witbrock. 2019. Improving natural language inference using external knowledge in the science questions domain. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7208–7215. AAAI Press.

Yichong Xu, Chenguang Zhu, Shuohang Wang, Siqi Sun, Hao Cheng, Xiaodong Liu, Jianfeng Gao, Pengcheng He, Michael Zeng, and Xuedong Huang. 2022. Human parity on commonsenseqa: Augmenting self-attention with external attention. In *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022*, pages 2762–2768. ijcai.org.

Yichong Xu, Chenguang Zhu, Ruochen Xu, Yang Liu, Michael Zeng, and Xuedong Huang. 2021. Fusing context into knowledge graph for commonsense question answering. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 1201–1207. Association for Computational Linguistics.

Michihiro Yasunaga, Antoine Bosselut, Hongyu Ren, Xikun Zhang, Christopher D. Manning, Percy Liang, and Jure Leskovec. 2022. Deep bidirectional language-knowledge graph pretraining. *CoRR*, abs/2210.09338.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 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: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 535–546. Association for Computational Linguistics.

Donghan Yu, Chenguang Zhu, Yuwei Fang, Wenhao Yu, Shuohang Wang, Yichong Xu, Xiang Ren, Yiming Yang, and Michael Zeng. 2022. Kg-fid: Infusing knowledge graph in fusion-in-decoder for open-domain question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4961–4974. Association for Computational Linguistics.

Xikun Zhang, Antoine Bosselut, Michihiro Yasunaga, Hongyu Ren, Percy Liang, Christopher D. Manning, and Jure Leskovec. 2022. Greaselm: Graph reasoning enhanced language models for question answering. *CoRR*, abs/2201.08860.

Chen Zheng and Parisa Kordjamshidi. 2022. Dynamic relevance graph network for knowledge-aware question answering. In *Proceedings of the 29th International Conference on Computational Linguistics, COLING 2022, Gyeongju, Republic of Korea, October 12-17, 2022*, pages 1357–1366. International Committee on Computational Linguistics.

Mantong Zhou, Zhouxing Shi, Minlie Huang, and Xiaoyan Zhu. 2020. Knowledge-aided open-domain question answering. *arXiv preprint arXiv:2006.05244*.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2020. Freelb: Enhanced adversarial training for natural language understanding. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

A Relation types

Table 6 gives all the relation types used in our method, 19 relations in total. We view all the relations as undirected in our experiments.

Relations	Merged Relation
Antonym DistinctFrom	Antonym
AtLocation LocatedNear	AtLocation
CapableOf	CapableOf
Causes CausesDesire MotivatedByGoal	Causes
CreatedBy	CreatedBy
IsA InstanceOf DefinedAs	ISA
Desires	Desires
HasSubevent HasFirstSubevent HasLastSubevent HasPrerequisite Entails MannerOf	HasSubevent
PartOf HasA	PartOf
HasContext	HasContext
HasProperty	HasProperty
Madeof	Madeof
NotCapableOf	NotCapableOf
NotDesires	NotDesires
ReceivesAction	ReceivesAction
RelatedTo SimilarTo Synonym	RelatedTo
UsedFor	UsedFor
SameQA	SameQA
DefAs	DefAs

Table 6: HKG involves relationship types. We follow the relationship type defined by (Yasunaga et al., 2021) and add "SameQA" and "DefAS" to it, which represent the relationship between key entities and the relationship between key entities and paraphrase entities, respectively.

B Details of Datasets

CommonsenseQA is a multiple-choice QA dataset that requires different types of commonsense knowledge to answer questions, with each question containing one correct choice and four distracting choices. The dataset has a total of 12,102 questions.

OpenBookQA is a QA dataset focusing on scientific facts that require a combination of scientific facts or commonsense knowledge to answer. It contains 5,957 questions, each containing one correct choice and three distracting choices. We conduct experiments on the official split dataset.

The statistics for the datasets are shown in Table 7.

Datasets	Train	Dev	Test
CSQA(Official)	9,741	1,221	1,140
CSQA(IHdata)	8,500	1,221	1,241
OBQA	4,957	500	500

Table 7: Statistics of CommonsenseQA (CSQA) and OpenBookQA (OBQA).

C Error types and Examples

Table 8 gives some examples of error analysis. Each example gives a part knowledge paths and paraphrase descriptions retrieved in multiple knowledge sources.

Error type	Examples	
Inappropriate paraphrases (8/50)	Question	What is another name for the color of the fur of a dog with light colored fur?
	Choices	✓ **fair** | ✗ basket | ✗ **dog hair** | ✗game | ✗sun
	Paths for correct answer	color $\xrightarrow{relatedto}$pale$\xrightarrow{relatedto}$fair; color $\xrightarrow{isa}$white$\xrightarrow{relatedto}$fair; ...
	Paths for predicted answer	fur$\xrightarrow{relateddto}$hair; fur $\xrightarrow{relatedto}$hairball$\xrightarrow{relateddto}$hair;fur$\xrightarrow{partof}$dog$\xrightarrow{madeof}$hair; ...
	Correct paraphrase description	fair: (used of hair or skin) pale or light colored.
	Inappropriate paraphrase description	fair: a competitive exhibition of farm products.
Indistinguishable knowledge paths (17/50)	Question	What animal has quills all over it?
	Choices	✗ feather | ✗ chicken | ✗ calligraphy | ✗**porcupine** | ✓ **hedgehog**
	Paths for correct answer	quill $\xrightarrow{partof}$hedgehog; quill $\xrightarrow{partof}$porcupine$\xrightarrow{relatedto}$hedgehog; ...
	Paths for predicted answer	quill $\xrightarrow{partof}$porcupine; quill $\xrightarrow{partof}$hedgehog$\xrightarrow{relatedto}$porcupine; ...
	Paraphrase description	porcupine: relatively large rodents with sharp erectile bristles mingled with the fur. hedgehog: relatively large rodents with sharp erectile bristles mingled with the fur. ...
Lack of relevant knowledge (13/50)	Question	Kramer wrote a self-referential book. What might that book be about?
	Choices	✗ counter | ✓ **coffee table** | ✗ school room | ✗ **backpack** | ✗ bedside table
	Paths for correct answer	book $\xrightarrow{atlocation}$coffee table; book $\xrightarrow{isa}$magazine$\xrightarrow{relatedto}$coffee table; ...
	Paths for predicted answer	book $\xrightarrow{partof}$backpack; book $\xrightarrow{atlocation}$satchel$\xrightarrow{relatedto}$backpack; ...
	Paraphrase description	coffee table: low table where magazines can be placed and coffee or cocktails are served. backpack: a bag carried by a strap on your back or shoulder. ...
Incomprehensible question (10/50)	Question	The pencil sharpener was broken in the classroom, where did the teacher recommend the student go?
	Choices	✗ **home** | ✓ **library** | ✗ stationery store | ✗ cabinet | ✗ desk drawer
	Paths for correct answer	pencil sharpener $\xrightarrow{atlocation}$library; pencil sharpener $\xrightarrow{atlocation}$desk$\xrightarrow{atlocation}$library; classroom $\xrightarrow{atlocation}$student$\xrightarrow{atlocation}$library; ...
	Paths for predicted answer	classroom $\xrightarrow{atlocation}$ferret $\xrightarrow{atlocation}$home; classroom $\xrightarrow{atlocation}$door$\xrightarrow{relatedto}$home; classroom $\xrightarrow{atlocation}$poet$\xrightarrow{atlocation}$home; ...
	Paraphrase description	classroom: a room in a school where lessons take place. pencil sharpener: a rotary implement for sharpening the point on pencils. ...

Table 8: Error analyse, we divide the error data into four categories

ACL 2023 Responsible NLP Checklist

A For every submission:

☑ A1. Did you describe the limitations of your work?
In Section 7 Limitations

☑ A2. Did you discuss any potential risks of your work?
In Section 7 Limitations

☑ A3. Do the abstract and introduction summarize the paper's main claims?
In Section 1

☒ A4. Have you used AI writing assistants when working on this paper?
Left blank.

B ☑ Did you use or create scientific artifacts?

In Section 1 and 3

☑ B1. Did you cite the creators of artifacts you used?
In Section 1

☐ B2. Did you discuss the license or terms for use and / or distribution of any artifacts?
Not applicable. Left blank.

☐ B3. Did you discuss if your use of existing artifact(s) was consistent with their intended use, provided that it was specified? For the artifacts you create, do you specify intended use and whether that is compatible with the original access conditions (in particular, derivatives of data accessed for research purposes should not be used outside of research contexts)?
Not applicable. Left blank.

☐ B4. Did you discuss the steps taken to check whether the data that was collected / used contains any information that names or uniquely identifies individual people or offensive content, and the steps taken to protect / anonymize it?
Not applicable. Left blank.

☐ B5. Did you provide documentation of the artifacts, e.g., coverage of domains, languages, and linguistic phenomena, demographic groups represented, etc.?
Not applicable. Left blank.

☑ B6. Did you report relevant statistics like the number of examples, details of train / test / dev splits, etc. for the data that you used / created? Even for commonly-used benchmark datasets, include the number of examples in train / validation / test splits, as these provide necessary context for a reader to understand experimental results. For example, small differences in accuracy on large test sets may be significant, while on small test sets they may not be.
Yes, we report the details of the dataset used in Section 4.1.

C ☑ Did you run computational experiments?

In Section 4

☑ C1. Did you report the number of parameters in the models used, the total computational budget (e.g., GPU hours), and computing infrastructure used?
In Section 4.2

The Responsible NLP Checklist used at ACL 2023 is adopted from NAACL 2022, with the addition of a question on AI writing assistance.

☑ C2. Did you discuss the experimental setup, including hyperparameter search and best-found hyperparameter values?
In Section 5.1

☑ C3. Did you report descriptive statistics about your results (e.g., error bars around results, summary statistics from sets of experiments), and is it transparent whether you are reporting the max, mean, etc. or just a single run?
In Section 4.4

☐ C4. If you used existing packages (e.g., for preprocessing, for normalization, or for evaluation), did you report the implementation, model, and parameter settings used (e.g., NLTK, Spacy, ROUGE, etc.)?
Not applicable. Left blank.

D ☒ Did you use human annotators (e.g., crowdworkers) or research with human participants?

Left blank.

☐ D1. Did you report the full text of instructions given to participants, including e.g., screenshots, disclaimers of any risks to participants or annotators, etc.?
No response.

☐ D2. Did you report information about how you recruited (e.g., crowdsourcing platform, students) and paid participants, and discuss if such payment is adequate given the participants' demographic (e.g., country of residence)?
No response.

☐ D3. Did you discuss whether and how consent was obtained from people whose data you're using/curating? For example, if you collected data via crowdsourcing, did your instructions to crowdworkers explain how the data would be used?
No response.

☐ D4. Was the data collection protocol approved (or determined exempt) by an ethics review board?
No response.

☐ D5. Did you report the basic demographic and geographic characteristics of the annotator population that is the source of the data?
No response.