Heterogeneous-Graph Reasoning With Context Paraphrase for Commonsense Question Answering

Yujie Wang, Hu Zhang, Jiye Liang, *Senior Member, IEEE*, and Ru Li, *Member, IEEE*

***Abstract*—Commonsense question answering (CQA) generally means that the machine uses its mastered commonsense to answer questions without relevant background material, which is a challenging task in natural language processing. Existing methods focus on retrieving relevant subgraphs from knowledge graphs based on key entities and designing complex graph neural networks to perform reasoning over the subgraphs. However, they have the following problems: i) the nested entities in key entities lead to the introduction of irrelevant knowledge; ii) the QA context is not well integrated with the subgraphs; and iii) insufficient context knowledge hinders subgraph nodes understanding. In this paper, we present a heterogeneous-graph reasoning with context paraphrase method (HCP), which introduces the paraphrase knowledge from the dictionary into key entity recognition and subgraphs construction, and effectively fuses QA context and subgraphs during the encoding phase of the pre-trained language model (PTLM). Specifically, HCP filters the nested entities through the dictionary's vocabulary and constructs the Heterogeneous Path-Paraphrase (HPP) graph by connecting the paraphrase descriptions[1] with the key entity nodes in the subgraphs. Then, by constructing the visible matrices in the PTLM encoding phase, we fuse the QA context representation into the HPP graph. Finally, to get the answer, we perform reasoning on the HPP graph by Mask Self-Attention. Experimental results on CommonsenseQA and OpenBookQA show that fusing QA context with HPP graph in the encoding stage and enhancing the HPP graph representation by using context paraphrase can improve the machine's commonsense reasoning ability.**

***Index Terms*—Natural language processing, heterogeneous graph, knowledge enhancement, question answering.**

I. Introduction

OVER the past few years, with the advent of large-scale pre-trained language models (PTLMs) [1], [2], [3], [4], Question Answering (QA) tasks have remarkably progressed, surpassing human levels on multiple QA tasks. However, the PTLMs still have a substantial gap with humans in QA tasks that require commonsense knowledge despite achieving good results in some tasks. Humans can use their commonsense knowledge in temporal, science, and society to help them understand the meaning of natural language according to practical situations. For example, if you ask: "Where would you expect to find a pizzeria while shopping?", then we know that a "pizzeria" can make "pizza", and "pizza" is a food. Therefore, inferring that you can "find a pizzeria" in a "food court" is easy. This simple reasoning ability may seem easy to human beings but is beyond the current capacity of natural language understanding systems.

Commonsense is the common daily consensus of most people on the same thing, which is the basis of daily human communication and cooperation. Commonsense can be categorized according to types, including social, temporal, physical commonsense, and so on. Several CQA datasets have recently been built on the basis of different types of commonsense. For example, SocialQA [5], MCTA-CO [6], PIQA [7], and CommonsenseQA [8] are respectively proposed for social commonsense, temporal commonsense, physical commonsense, and general commonsense.

PTLMs also capture general knowledge about the world. However, the representation of knowledge in PTLMs remains unclear, and even the knowledge in PTLMs may be noise for specific questions, affecting the machine's response. Abundant commonsense knowledge is stored in Knowledge Graphs (KGs), such as Freebase [9], ATOMIC [10] and ConceptNet [11] et al. Machine can use these KGs to make sound judgments, and KGs can also provide explicit and explanatory evidence. Therefore, the existing methods introduce KGs when solving CQA tasks to improve the commonsense reasoning capability of machine, including knowledge-aware graph network [12], multi-hop graph relation network [13], QA-GNN [14] and JointLK [15]. The general steps of these methods include: (i) recognize key entities in QA context by entity recognition tools or other methods; (ii) extract knowledge paths and construct subgraphs based on key entities; (iii) design reasoning modules based on graph neural networks (GNNs) [16] and reason answers on subgraphs. Therefore, the recognize key entities are particularly important, which determine the quality and size of the subgraphs.

However, the key entities recognition by the previous methods [12], [13], [14], [15] contain some nested entities. An example is shown in Fig. 1, the "empire state building", "fifth avenue" and "new york city" are noun phrases that should be considered as a whole and do not need to be split into subwords. When retrieving knowledge paths in KG, subwords such as "empire", "building" introduce paths that are weakly associated

Manuscript received 9 November 2022; revised 19 February 2024; accepted 14 July 2024. Date of publication 26 July 2024; date of current version 15 August 2024. This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0106100 and in part by the National Natural Science Foundation of China under Grant 62176145. The associate editor coordinating the review of this article and approving it for publication was Prof. Zoraida Callejas. *(Corresponding author: Hu Zhang.)*

The authors are with the Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, School of Computer and Information Technology, Shanxi University, Taiyuan 030006, China (e-mail: init_wang@foxmail.com; zhanghu@sxu.edu.cn; ljy@sxu.edu.cn; liru@sxu.edu.cn).

[1]The paraphrase descriptions are English explanations of words or phrases in WordNet and Wiktionary.

Digital Object Identifier 10.1109/TASLP.2024.3434469

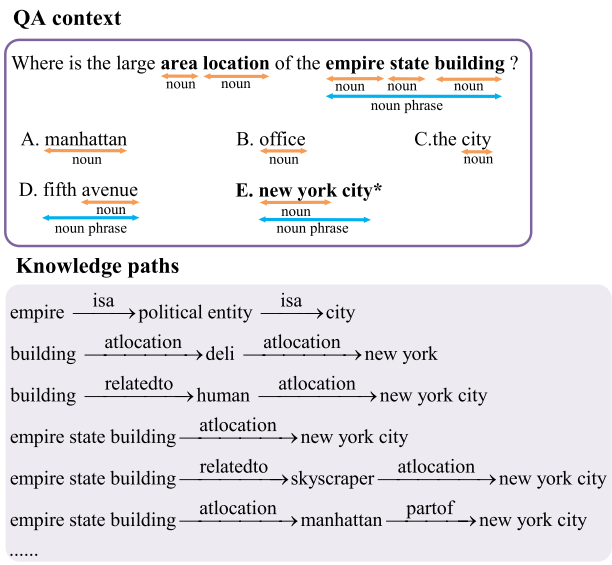


Fig. 1. An example from the CommonsenseQA dataset, where nested entities exist in both question and choices.

with the QA context (e.g. "empire $\xrightarrow{isa}$ political entity $\xrightarrow{isa}$ city", "building $\xrightarrow{relatedto}$ deli $\xrightarrow{atlocation}$ new york", etc.), and these paths may lead to biased machine reasoning. Meanwhile, the machine lacks the understanding of the entity context. In Fig. 1, "empire state building" means "a skyscraper built in new york city in 1931" and "new york city" means "the largest city in new york state and in the United States". Combined with these commonsense knowledge, the answer "E" can be easily obtained. However, the machine is difficult to get the right answer due to the lack of commonsense knowledge. Moreover, the existing methods [14], [15] only interact with the QA context and the subgraphs at a relatively shallow level in the reasoning process, which cannot well incorporate QA context into subgraphs nodes, and may lead to some bias in the reasoning process.

In order to address the above problems, we propose a heterogeneous-graph reasoning with context paraphrase method (HCP). Given the QA context consisting of a question and multiple choices, we first recognize the candidate entities using KeyBert [17], and further filter the candidate entities using WordNet [18] and Wiktionary[2] to obtain the key entities. Meanwhile, we retrieve the paths within k hops in ConceptNet [11] based on the key entities, and retrieve the paraphrases of the key entities in WordNet and Wiktionary. Based on this, we construct a heterogeneous graph based on the relationship between the paths and paraphrases, which is named as Heterogeneous Path-Paraphrase (HPP) graph. Then, we construct two visible matrices to introduce contextual information of the QA context into the HPP graph, and to limit the impact of external knowledge on the QA context. Finally, we feed the paraphrases, QA context, paths and visible matrices into the PTLM to encode the HPP graph and perform reasoning on the HPP graph by a Mask Multi-head Self-Attention (MMSA).

The main contributions of this paper are as follows:

1) We introduce WordNet and Wiktionary to filter nested entities and reduce the irrelevant entities, thus realizing pruning of knowledge paths. Meanwhile, we incorporate the paraphrased descriptions from the two dictionaries into the construction process of HPP graph.
2) We fuse QA context representation into the HPP graph through two visible matrices during the PTLM's encoding phase.
3) We also evaluate our method on multiple CQA datasets, and prove the effectiveness of the proposed method through a series of ablation experiments and case studies.

II. Related Studies

Currently, existing methotds combine PTLMs and structured KGs for commonsense reasoning, providing explicit evidence while improving commonsense reasoning and answering performance. KagNet [12] retrieves the relationship paths between the question and choice entities from ConceptNet [11] and models the relationship between the entity nodes through Graph convolutional networks (GCNs) [19] and LSTMs. MHGRN [13] unifies the reasoning methods based on paths and GNNs to achieve improved interpretability and scalability. QA-GNN [14] uses PTLM to compute the relevance of KG nodes conditioned on the given QA context, then joint reasoning over the QA context and KG. JointLK [15] performs joint reasoning between PTLM and RGAT [20] through a dense bidirectional attention and designs a dynamic pruning module to remove irrelevant nodes. ACP [21] obtains the AMR graph by performing Abstract Meaning Representation (AMR) on the questions, and integrates the AMR graph with the subgraphs retrieved from KG to obtain the ACP graph, which is then used to explain the reasoning path. Above methods extract subgraphs in KGs based on key entities, but these key entities contain some nested entities. The nested entities may introduce some noisy knowledge in the knowledge retrieval process and affect the reasoning effect of the machine.

Although the KGs are rich in structured knowledge, they lack contextual semantic information. Therefore, [22], [23], [24] improve the commonsense reasoning ability of machines by integrating knowledge from multiple knowledge sources. Inspired by the Xu et al. [24], We introduce Wiktionary and WordNet into key entity recognition to filter nested entities and realize pruning of knowledge paths. Meanwhile, we explicitly view the paraphrase descriptions as additional nodes (paraphrase nodes) and connect them to the key entities in the HPP graph.

III. Method

In this section, we introduce the specific details of the HCP method. As shown in Fig. 2, HCP consists of five parts: 1) key entities recognition module, 2) HPP graph construction module, 3) PTLM based encoding module, 4) MMSA based reasoning module, and 5) answer prediction module.

A. Task Definition

For a commonsense question answering task, given a question and multiple choices $A = \{a_1, a_2, ..., a_p\}$, the machine

[2][Online]. Available: https://www.wiktionary.org/

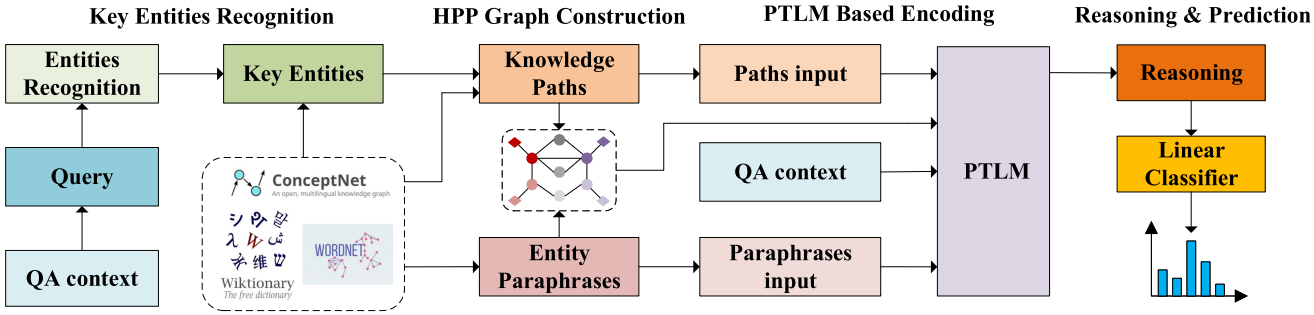


Fig. 2. The Overall Architecture, the paraphrase knowledge is introduced into key entities recognition and HPP graph construction.

needs to choose the correct answer from A without background material. Therefore, the external knowledge bases (KBs) can provide some useful information for commonsense question answering. In this work, we retrieve knowledge paths from ConceptNet based on key entities, retrieve paraphrases of key entities from WordNet and Wiktionary, and construct the HPP graph $g = (V, S)$ based on knowledge paths and paraphrases. Here V is the set of entity nodes and paraphrase nodes, and S is the set of edges between nodes. We need to retrieve relevant knowledge in multiple KBs to construct a HPP graph based on key entities. Therefore, our method starts from a key entities recognition module.

B. Key Entities Recognition

In the process of answering commonsense questions, we need to retrieve knowledge paths from ConceptNet based on key entities. The key entities determine the quality of the knowledge paths. As shown in Fig. 2, we first recognize the key entities in the QA context. Specifically, we replace the interrogative word in the question with the choice to get the Query (e.g. given a question "The kids didn't clean up after they had done **what**?" and choice "play with toys", it is converted to "The kids didn't clean up after they had done **play with toys**."). Then, we use the KeyBERT [17] to recognize the candidate entities $E = \{e_1, e_2, \ldots, e_m\}$ in the Query, and filter the candidate entities through Wiktionary and Wordnet's vocabulary to get the key entities. For a candidate entity e_i, if the length of e_i is greater than 1 and e_i can be retrieved in vocabulary, we remove the subwords that make up e_i in the candidate entities E. Meanwhile, we remove the stop words in the candidate entities. Finally, we obtain the key entities $\tilde{E} = \{\tilde{e_1}, \tilde{e_2}, \ldots \tilde{e_n}\}, n \leq m$, where n, m denotes the number of key entities and candidate entities.

C. HPP Graph Construction

ConceptNet is a large-scale knowledge graph of commonsense comprising relationship-based knowledge in the form of triples, with millions of nodes and relationships. Following the previous study by Yasunaga et al. [14], we retrieve the knowledge paths within k-hops in ConceptNet based on the question and choice entities in the key entities. A path includes a question entity, k relationship descriptions, k-1 correlation entity, and a choice entity. For the example in Fig. 1, "building $\xrightarrow{relatedto}$ deli $\xrightarrow{atlocation}$ new york" is a 2-hop knowledge path, where

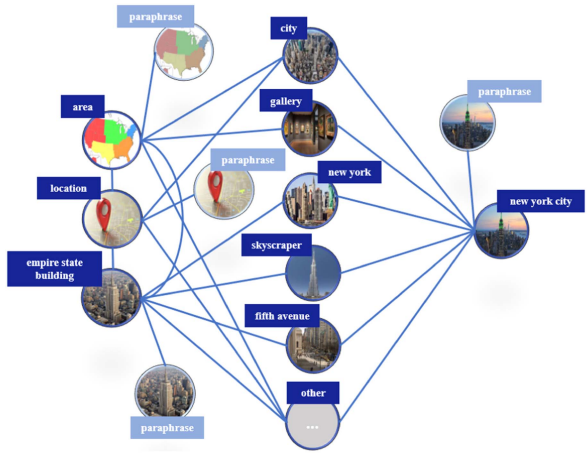


Fig. 3. An example of an HPP graph. For simplicity, we ignore the relationships between entities.

"building" is a question entity, "deli" is a correlation entity, "new york" is a choice entity, "relatedto" and "atlocation" are relationship descriptions.

Understanding the meaning of knowledge paths based solely on the relationship between entities is difficult for the machine. Therefore, we retrieve paraphrase descriptions of key entities in WordNet and Wiktionary to further enhance the representation of key entities. Specifically, if the entity e_i is a word, the paraphrase of e_i in WordNet is retrieved according to the POS tag. Meanwhile, if multiple paraphrases exist, we calculate the similarity between each paraphrase and Query based on Sentence-BERT [25], and select the one with the greatest similarity as the paraphrase of e_i. Similarly, if e_i is a phrase, we retrieve all the paraphrases of e_i in WordNet and select the paraphrase with the greatest similarity to Query as paraphrase of e_i. Furthermore, if e_i does not exist in WordNet, we retrieve it in Wiktionary. Finally, if e_i does not exist in WordNet and Wiktionary, we retrieve the paraphrase of the basic word (select POS tag as noun, proper noun, verb, and adjective in that order) for e_i. After getting the knowledge paths and paraphrase descriptions, we construct the HPP graph based on the relationship between them. The construction of HPP graph is further described below.

The Fig. 3 gives an example of the HPP graph. The HPP graph consists of the relationship between the knowledge paths and the paraphrase descriptions, including question entity nodes, choice entity nodes, correlation entity nodes, and paraphrase nodes. The following rules are used in the construction of the HPP graph.

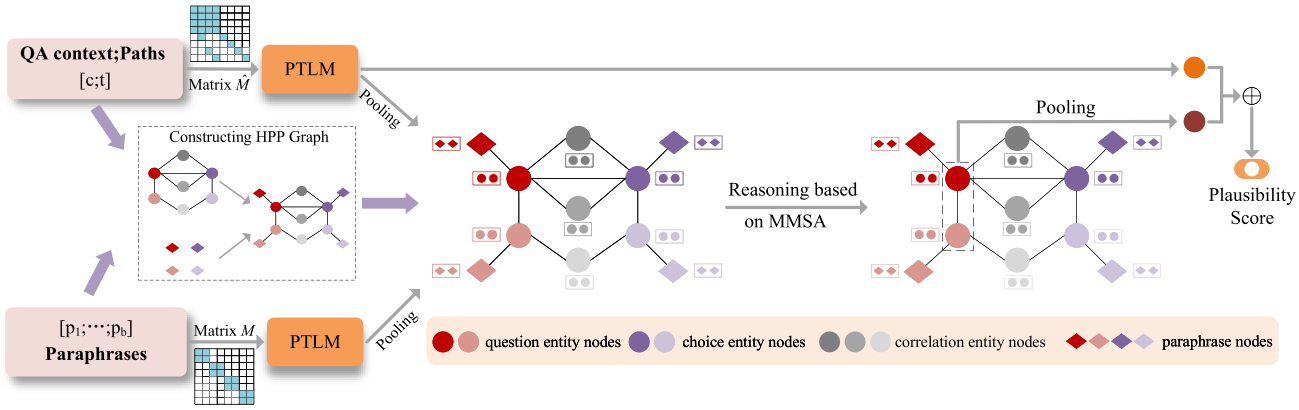


Fig. 4. In our method, the paraphrase descriptions are connected as additional nodes to the key entities nodes in the HPP graph, and the QA context representation is fused with the HPP graph in the PTLM's encoding phase.

1) the question entity nodes, correlation entity nodes and choice entity nodes in the same path are successively connected, e.g.,"area", "city" and "new york city" in Fig. 3 denote question entity, correlation entity and choice entity respectively, which are connected successively;
2) the question entity nodes from the same question are connected to each other, the choice entity nodes from the same choice are also connected to each other, e.g., "area", "location", and "empire state building" in Fig. 3 are question entities from the same question and they are connected to each other;
3) the question entity nodes and choice entity nodes are connected to their corresponding paraphrases nodes, e.g., entities such as "area" and "new york city" in Fig. 3 are connected to their corresponding paraphrase entities, respectively.

D. PTLM Based Encoding

After retrieving the knowledge paths and key entity paraphrases. As shown in Fig. 4, we construct two visible matrices and use the RoBERTa to encode the QA context, paths and key entity paraphrases. The construction of visible matrices and token encoding is further described below.

Visible matrices construction: We fuse QA context representation and HPP graph representation by visible matrices in the PTLM's encoding phase. Meanwhile, the visible matrices can weaken the influence between non-associated tokens in the encoding phase. Inspired by Liu et al. [26], we further construct the QA-paths visible matrix $\hat{\boldsymbol{M}}$ based on the relationship between QA context and knowledge paths. $\hat{\boldsymbol{M}}_{i,j} = 1$ means that two tokens are visible to each other and $\hat{\boldsymbol{M}}_{i,j} = 0$ means that they are not visible. We define the following rules to construct $\hat{\boldsymbol{M}}$.

1) the tokens in the QA context are visible to each other;
2) the question entity q^e and choice entity c^e are extracted from the QA context, and they can see themselves and the tokens in the corresponding positions of the QA context.
3) the correlation entity a^e and relationship description r^e can only be seen by themselves.

Similarly, we also construct the visible paraphrase matrix $\tilde{\boldsymbol{M}}$ to prevent different paraphrases from influencing each other and changing the meaning of paraphrases. In the $\tilde{\boldsymbol{M}}$, the tokens from the same paraphrase are visible to each other.

Token embedding: The visible matrices are incorporated into the RoBERTa model, which not only fuses the QA context with the HPP graph, but also prevents changing the meaning of the other non-associated tokens or even entire QA context due to paths.

In the RoBERTa, any of the visible matrices $\boldsymbol{M}$ is further defined as:

$$\bar{\boldsymbol{M}}_{i,j} = \begin{cases} -10^4 & \boldsymbol{M}_{i,j} = 1 \\ 0 & \boldsymbol{M}_{i,j} = 0 \end{cases}. \tag{1}$$

We incorporate $\bar{\boldsymbol{M}}_{i,j}$ into the RoBERETa by MMSA. MMSA is defined as:

$$s_{i,j} = \frac{\boldsymbol{Q}_t \boldsymbol{K}_t^T}{\sqrt{d}}, \tag{2}$$

$$a_{ij} = Softmax(s_{ij} + \bar{\boldsymbol{M}}_{i,j}), \tag{3}$$

$$\boldsymbol{h}_{t+1} = a_{ij} \boldsymbol{V}_t. \tag{4}$$

The $\boldsymbol{h}_t$ is the hidden state of RoBERTa at t layer. $\boldsymbol{Q}_t$, $\boldsymbol{K}_t$ and $\boldsymbol{V}_t$ are obtained by linear transformation of $\boldsymbol{h}_t$ through three different fully connected layers, a_{ij} is the attention weight after integrating $\bar{\boldsymbol{M}}_{i,j}$, d is the size of RoBERTa hidden states.

We concatenate the QA context C and the paths T, which are fed into the pre-trained RoBERTa model along with the QA-paths matrix $\hat{\boldsymbol{M}}$. The representations $\boldsymbol{C} = \{\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_l\} \in \mathbb{R}^{l \times d}$ and $\boldsymbol{T} = \{\boldsymbol{t}_1, \boldsymbol{t}_2, \ldots, \boldsymbol{t}_w\} \in \mathbb{R}^{w \times d}$ are obtained, where l, w are lengths of QA context and paths.

$$\hat{\boldsymbol{H}} = \mathbf{RoBERTa} = ([\mathrm{C}; \mathrm{T}], \hat{\boldsymbol{M}}). \tag{5}$$

Similarly, we feed the paraphrases P and visible matrix $\tilde{M}$ into the RoBERTa model to obtain representation $\boldsymbol{P} = \{\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_b\} \in \mathbb{R}^{b \times d}$, where b is lengths of paraphrases.

$$\tilde{\boldsymbol{H}} = \mathbf{RoBERTa}([\mathrm{P}], \tilde{\boldsymbol{M}}). \tag{6}$$

E. MMSA Based Reasoning

After encoding the QA context C, paths T and paraphrases P, the remaining challenge is to encode the HPP graph and perform reasoning based on the HPP graph.

Encoding HPP graph: Given a graph structure, it is first necessary to obtain an initial representation of each node in the graph. In Section III-D, we obtain the tokens embedding of QA context, paths, and paraphrases. For each node v in the graph, $v = \{v_i, v_{i+1}, ..., v_k\}$ is a certain segment of the path tokens or paraphrase tokens, and the node representation is obtained through the average pooling method.

$$\boldsymbol{u} = AvergePooling([v_i, v_{i+1}, ..., v_k]). \tag{7}$$

During the reasoning process, we should consider the relationship between two nodes. Therefore, we obtain the embedding $\boldsymbol{u}_r$ of the relation description by averaging pooling and concatenate $\boldsymbol{u}_r$ with its next node embedding $\boldsymbol{u}_v$ by

$$\boldsymbol{u}_{rv} = f(\boldsymbol{u}_r, \boldsymbol{u}_v). \tag{8}$$

where $f : \mathbb{R}^{2\,d} \rightarrow \mathbb{R}^d$ is a 1-layer MLP.

Reasoning: Attention can capture the strength of the association between two nodes. Inspired by the works of Shao et al. [27], we use the MMSA to replace GNNs to perform reasoning on the HPP graph.

Specifically, For the j-th layer MMSA, we obtain the query, value, and key vectors $\boldsymbol{Q}_j$, $\boldsymbol{K}_j$ and $\boldsymbol{V}_j$ by

$$\boldsymbol{Q}_j = f_j^q(\tilde{\boldsymbol{u}}_{j-1}), \tag{9}$$

$$\boldsymbol{K}_j = f_j^k(\tilde{\boldsymbol{u}}_{j-1}), \tag{10}$$

$$\boldsymbol{V}_j = f_j^v(\tilde{\boldsymbol{u}}_{j-1}), \tag{11}$$

where $f_j^q : \mathbb{R}^d \rightarrow \mathbb{R}^d$, $f_j^k : \mathbb{R}^d \rightarrow \mathbb{R}^d$ and $f_j^v : \mathbb{R}^d \rightarrow \mathbb{R}^d$ are linear transformations, $\tilde{\boldsymbol{u}}_{j-1}$ is the nodes embedding representation at the (j-1)-th layer of the HPP graph. For brevity, we formulate the entire computation in the j-th layer as:

$$\tilde{\boldsymbol{u}}_j = \sigma(MMSA(\boldsymbol{Q}_j, \boldsymbol{K}_j, \boldsymbol{V}_j)). \tag{12}$$

where $\tilde{\boldsymbol{u}}_j$ is the nodes embedding of j-th layer, σ is a LayerNorm layer.

We use the N-layer MMSA to update the node representation in the HTP graph and get the embedding $\tilde{\boldsymbol{g}}$ of the HPP graph by

$$\tilde{\boldsymbol{g}} = MaxPooling([\tilde{\boldsymbol{u}}^q]). \tag{13}$$

where $\tilde{\boldsymbol{u}}^q$ is the embedding representation of all question entity nodes in the HPP graph.

F. Answer Prediction

We obtain the QA context representation $\tilde{\boldsymbol{c}}$ by averge pooling and concatenate $\tilde{\boldsymbol{c}}$ and HPP graph embedding $\tilde{\boldsymbol{g}}$, then feed them into a linear classifier to the answer score. The formula is as follows:

$$s = Linear([\tilde{\boldsymbol{c}}; \tilde{\boldsymbol{g}}]). \tag{14}$$

TABLE I
STATISTICS OF COMMONSENSEQA, OPENBOOKQA, HELLASWAG, AND SOCIAL IQA

Datasets	Train	Dev	Test
CommonsenseQA(Official)	9741	1221	1140
CommonsenseQA(IHdata)	8500	1221	1241
OpenBookQA	4957	500	500
HellaSWAG	39 905	10 042	10 003
SOCIAL IQa	33 410	1954	2224

IV. EXPERIMENT

A. Datasets

We evaluate our method using CommonsenseQA [8] and OpenBookQA [28] as the primary datasets, and adopt HellaSWAG [42] and SOCIAL IQa [5] as secondary datasets.

CommonsenseQA is a commonsense question answering dataset with 12,102 multiple-choice questions, requires different types of commonsense knowledge to get the correct answer. Each question contains one correct choice and four interference choices. Since the answers of the official test set are not published, we can only submit the leaderboard once every two weeks. Therefore, we follow the work of Lin et al. [13] and split CommonsenseQA into in-house dataset (IHdata), where the training set is divided into IHtrain and IHtest, and the dev set is divided into IHdev.

OpenBookQA is a question answering dataset based on an open book exam that evaluates human understanding of a topic, and each question must be answered in combination with scientific facts or commonsense knowledge. OpenBookQA contains 5,957 multiple-choice questions, each question contains one correct choice and three interference choices.

HellaSWAG is a question answering dataset designed to assess general language understanding. It consists of 59,950 multiple-choice questions, each with answers that are anomalous or unusual without a thorough understanding of the context.

SOCIAL IQa is a question answering dataset designed for commonsense reasoning in social situations. It comprises 37,588 multiple-choice questions that cover social emotions and wisdom in everyday scenarios.

The statistics for the datasets are shown in Table I.

B. Experimental Setting

We use RoBERTa-large [2] as the encoder and use Adam [29] as the model optimizer. In the data processing, for each data, we extract 80 paths within 2 hops from ConcetNet. In training, we set the learning rate to 1e-5, batch size to 8, epoch to 5, and number of layers (N = 4) of MMSA. Each model is trained using GPU (Tesla P100), which takes 6 hours on average.

C. Compared Method

We use RoBERTa-large to fine-tune our model on CommonsenseQA and OpenBookQA, and compare with existing RoBERTa-large+KBs methods, including relation network (RN) [30], RGCN [31], GconAttn [32], KagNet [12], MHGRN [13], QA-GNN [14], JointLK [15] and GREASELM [33].

TABLE II
PERFORMANCE COMPARISON ON COMMONSENSEQA IN-HOUSE SPLIT, WE USE ROBERTA-LARGE TO FINE-TUNE DEKCOR

Methods	IHdev-Acc.(%)	IHtest-Acc.(%)
Fine-tuned RoBERTa(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)
+ DEKCOR*	78.21(±0.23)	73.78 (±0.39)
+ GREASELM	78.5(±0.5)	74.2(±0.4)
+ JointLK	77.88 (±0.25)	74.43 (±0.83)
+ HCP (ours)	**79.38** (±0.67)	**74.93** (±0.31)

TABLE III
PERFORMANCE COMPARISON ON COMMONSENSEQA OFFICIAL LEADERBOARD

Methods	Test-Acc.(%)
RoBERTa [2]	72.1
Albert [3] (ensemble)	76.5
RoBERTa + FreeLB [34]	72.19
RoBERTa + HyKAS [35]	73.2
RoBERTa + KE	73.3
RoBERTa + KEDGN (ensemble)	74.4
XLNet + Graph Reasoning [22]	75.3
RoBERTa + MHGRN [13]	75.4
ELECTRA + ACP [21]	75.43
ALBERT + Path Generator [36]	75.6
RoBERTa + QA-GNN [14]	76.1
RoBERTa + JointLK [15]	76.6
RoBERTa + HCP (our)	**77.02**

Although DEKCOR [24] also uses ConceptNet and Wiktionary, it uses ALBERT-xxlarge [3] as an encoder. For fair comparison, we retrain the DEKCOR on the IHdata using RoBERTa-large. In addition, to further validate the generalizability of HCP, we conduct secondary experiments on HellaSWAG and SOCIAL IQa. We also perform experiments on ALBERT-xx-large, T5-large and LLama2-7B[3] [43].

D. Main Results

We evaluate our method on CommonsenseQA and OpenBookQA. The specific experimental results are as follows.

CommonsenseQA: Tables II and III show the experimental results on IHdata and the official test set. We can see that the HCP achieves the best performance compared to other methods. Compared to Fine-tuned LMs, HCP improves by 6.31% and 6.24% in IHdev and IHtest. Meanwhile, HCP's performance on IHdata is improved by 1.5% and 0.5% over the best baseline model JointLK. Moreover, HCP's performance is 0.42% better than JointLK in the official leaderboard.

OpenBookQA: Additional experiments on the OpenBookQA dataset are conducted to further demonstrate the effectiveness of the proposed method. Table IV shows that HCP increases 1.12% higher than the RoBERTa + JointLK model and AristoRoBERTa + JointLK increases 0.88%. Table V shows the accuracy on the official leaderboard of OpenBookQA, our method achieves good performance in the OpenBookQA leaderboard.

[3]Due to the large number of parameters in Llama2, we perform lightweight fine-tuning on it based on LoRa.

TABLE IV
PERFORMANCE COMPARISON ON THE TEST SET OF OPENBOOKQA

Methods	RoBERTa	AristoRoBERTa
Fine-tuned LM (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)	80.6
+ QAGNN	70.58 (±1.42)	82.77 (±1.56)
+ GREASELM	-	84.8
+ JointLK	70.34 (±0.75)	84.92 (±1.07)
+ HCP (ours)	**71.46** (±1.22)	**85.8** (±0.89)

Methods with AristoRoBERTa use the textual evidence by Clark et al. [37] as an additional input to the QA context.

TABLE V
PERFORMANCE COMPARISON ON OPENBOOKQA OFFICIAL LEADERBOARD

Methods	Test-Acc.(%)
Careful Selection [38]	72.0
AristoRoBERTa	77.8
KF+SIR [39]	80.0
AristoRoBERTa + Path Generator [36]	80.2
AristoRoBERTa + MHGRN [13]	80.6
ALBERT + KB	81.0
AristiRoBERTa + QA-GNN [14]	82.8
T5-3B [40]	83.2
AristoRoBERTa + GREASELM [33]	84.8
AristoRoBERTa + JointLK [15]	85.6
UnifiedQA(11B)* [41]	87.2
AristoRoBERTa + HCP (our)	**86.6**

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.

TABLE VI
IN DIFFERENT PTLMS, HCP'S PERFORMANCE ON THE IHDEV SET AND IHTEST SET OF COMMONSENSE (CSQA), AND THE TEST SET OF OPENBOOKQA (OBQA)

Methods	CSQA-IHdev	CSQA-IHtest	OBQA-test
T5	69.10	65.97	65.33
T5 + HCP	73.51	69.78	67.60
ALBERT	79.41	75.63	71.67
ALBERT + HCP	82.03	79.37	76.06
Llama	84.28	79.21	84.40
Llama + HCP	84.74	81.71	85.40

To further validate the generalizability of HCP, we report HCP's performance on different PTLMs and other CQA datasets in Tables VI and VII. As can be seen from the tables, HCP basically brings performance improvement on different PTLMs and other CQA datasets, which proves the good generalizability of our proposed method.

V. ANALYSIS

A. Ablation Study

To further evaluate our method, we conduct ablation experiments on the official dev sets of CommonsenseQA and OpenBookQA to evaluate the contribution of each module to our method. Including:

TABLE VII
IN DIFFERENT PTLMS, HCP'S PERFORMANCE ON THE DEV SETS OF HELLASWAG AND SOCIAL IQA

Methods	HellaSWAG	SOCIAL IQa
T5	78.32	69.90
T5 + HCP	79.18	72.00
RoBERTa	83.52	72.72
RoBERTa + HCP	84.90	77.02
ALBERT	88.10	77.20
ALBERT + HCP	89.56	78.92
Llama	94.81	81.68
Llama + HCP	95.16	81.68

TABLE VIII
ABLATION RESULTS ON THE COMMONSENSEQA AND OPENBOOKQA OFFICIAL DEV SET

Methods	CommonsenseQA	OpenBookQA
HCP	80.51	70.6
-PE	78.56	69.2
-PN	79.11	69.6
-VM	76.98	68.2
-QP	77.23	69.8
-HPP	76.16	68.8

1) -PE: The WordNet and Wiktionary are not introduced to process nested entities and use candidate entities to retrieve knowledge paths.
2) -PN: The paraphrase nodes in HPP graph are removed;
3) -VM: The tokens of knowledge paths and QA contexts are mutually visible;
4) -QP: The QA context and key entities in the knowledge path are not visible;
5) -HPP: The HPP graph is removed and only fine-tune on RoBERTa.

Table VIII shows the results of the ablation experiments on CommonsenseQA and OpenBookQA. On CommonsenseQA, if the dictionaries is not introduced to handle nested entities, the model (-PE) performance decreases by 1.95%. This indicates that nested entities introduce some irrelevant paths, such as "empire $\xrightarrow{isa}$ political entity $\xrightarrow{isa}$ city", "building $\xrightarrow{relatedto}$ deli $\xrightarrow{atlocation}$ new york" in Fig. 1, which affect the model's reasoning performance. When we remove the paraphrase nodes from the HPP graph, the model (-PN) results decreased by 1.4%. This is because paraphrases contain further explanations and descriptions of key entities, which can help the model to better understand the knowledge paths, and even some paraphrases can directly establish the relationship with the correct choice. For example, the paraphrase of "empire state building" in Fig. 1 is "a skyscraper built in new york city in 1931...", which directly establishes a relationship with the correct choice "new york city", thus aiding the model in making correct judgments to some extent. When we remove the visible matrice between the QA context and the HPP graph, the QA context and the entities contained in the HPP graph become visible to each other. At this point, the model (-VM) performance decreases by 3.53%, which is due to the fact that the HPP graph contains a large number of entities that may change the original semantics of the QA context, leading to biased model reasoning.

TABLE IX
STATISTICS ON NESTED ENTITIES IN COMMONSENSEQA AND OPENBOOKQA

Datasets	Train	Dev	Test
CommonsenseQA	7 471/9741	921/1221	848/1140
OpenBookQA	2 258/4957	249/500	237/500

TABLE X
PERFORMANCE COMPARISON OF CONTAINING NESTED ENTITIES DATA ON COMMONSENSE (CSQA) AND OPENBOOKQA (OBQA)

Methods	CSQA-dev	OBQA-dev	OBQA-test
RoBERTa + HCP	76.42	54.70	66.90
- PE	74.84	51.28	61.87

Since CSQA did not publish the answers to the test set, we cannot perform experiments on CSQA-test.

Our method establishes associations between key entities in the HPP graph and the QA context through self-attention of the PTLM, thereby achieving deep integration of QA context and knowledge. When we set the key entities to be invisible to the corresponding tokens in the QA context, the model (-QP) performance decreases by 3.28%, which indicates that fusing the QA context with the HPP graph during the encoding process of PTLM is effective. The model (-HPP) performance decreased by 4.35% after removing the HPP graph, proving that the KBs can provide some useful clues to commonsense questions and improve the machine's reasoning performance. Similarly, the model performance decreases after removing each module on OpenBookQA.

B. Case Study

Nested Entity Study: We report the frequency of nested entities in CommonsenseQA and OpenBookQA as shown in Table IX. Nested entities exist in 76.3% and 46.1% of the data in both datasets. For nested entities, we should consider them as a whole, which is more consistent with the semantic meaning of the question. Meanwhile, the introduction of some irrelevant paths can be reduced when retrieving knowledge paths, thus enabling pruning of HPP graph. We train the HCP using data from CommonsenseQA and OpenBookQA that contain nested entities. It is worth noting that while some of the data contain nested entities, these entities do not retrieve any knowledge from the ConceptNet, and we therefore exclude these data.[4] Table X gives the HCP experimental results on CommonsenseQA and OpenBookQA. We can see that adding the handling of nested entities can greatly improve the model performance.

Fig. 5 further shows the impact of nested entities on knowledge paths retrieval. The first example in Fig. 5 inquires about for "the location of empire state building", and "empire state building" is a skyscraper, which is a building. When we retrieve knowledge paths based on candidate entities without processing nested entities, the "empire state building" and "new york city" are cut into multiple subwords. Retrieving paths by these subwords will introduce entities that are less relevant to the question,

[4]The Commonsense and OpenBookQA participate in training and testing with data of 4,926/632 and 1,038/117/139, respectively.

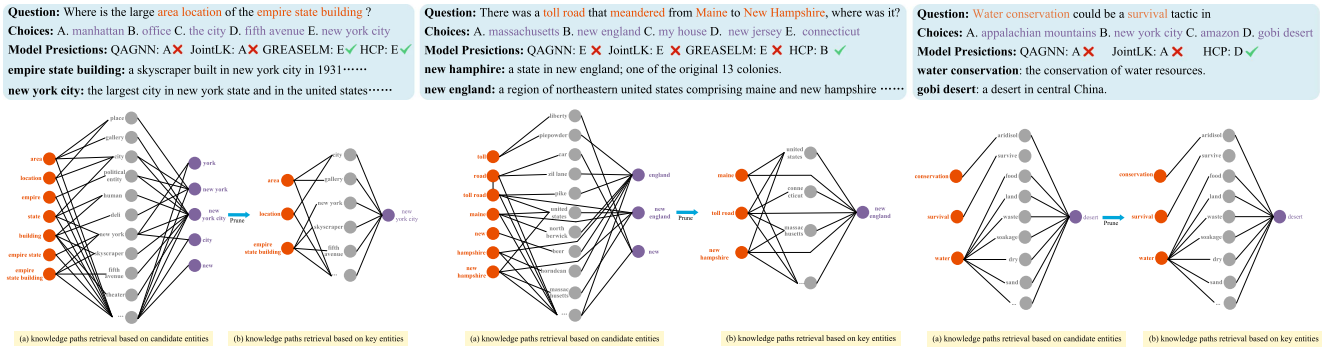


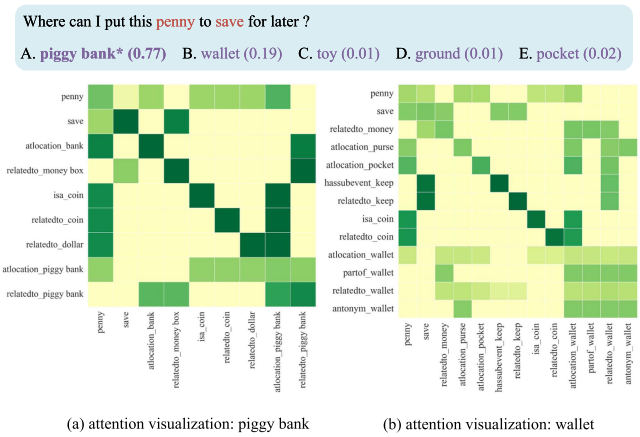
Fig. 5. Knowledge paths retrieval. We prune the paths by incorporating WordNet and Wiktionary, and provide paraphrase descriptions for some entities.

such as "political entity", "deli", "human", etc. Moreover, the meaning of some subwords is not very relevant to the whole entity. For example, "empire" means "the right or territory of a sovereign", which is irrelevant to "empire state building". These entities and subwords that are weakly associated with the question semantic can introduce some irrelevant knowledge paths and lead to bias in the model reasoning process. When we introduce WordNet and Wiktionary to process the nested entities, "empire state building" and "new york city" are considered as a whole without splitting, so that we can reduce the introduction of some irrelevant paths when retrieving knowledge paths, and thus indirectly achieve pruning of HPP graph.

At the same time, we noticed that in some data, filtering nested entity subwords through dictionaries may result in losing some key paths; however, the noise paths introduced by these subwords far exceed the number of key paths. Taking the second example, the subword "road" in "toll road" can find the knowledge path related to the answer "B", but it also introduces more noisy entities, such as "car", "zil lane", "telford" and so on. Additionally, in most cases, key paths can be obtained through nested entities, so filtering subwords of nested entities through dictionaries is feasible and can reduce the introduction of noisy knowledge.

We further consider the paraphrase descriptions of key entities as paraphrase nodes and connect them to key entities, which can guide the model's reasoning. For example, in the second example, all the baseline models predict the answer as "E". We analyze and find that choices "B" and "E" have similar knowledge paths, making it difficult for the model to decide between the two. However, through the paraphrase descriptions, we can ascertain that "maine" and "new hampshire" are both "states within the new England region of the United States". By incorporating paraphrase descriptions, our method yields the correct answer.

In addition, we also found some data that although contain nested entities, these entities cannot retrieve any knowledge in ConceptNet. Therefore, in such cases, we still use the subwords of the nested entities for retrieval. For example, in the third example, "water conservation" and "gobi desert" are both nested entities, but they cannot retrieve any knowledge. Thus, we still use subwords like "water," "conservation," "desert," etc., for retrieval to ensure that key paths are not lost.



(a) attention visualization: piggy bank (b) attention visualization: wallet

Fig. 6. Attention visualization graph of example 1, the darker colors indicating higher attention weights.

Interpretability study: As shown in Table XI, we further analyse the experimental results of the different ablation models with some examples.

In the first example, without introducing any external knowledge, the model (-HPP) gets the wrong answer "**pocket**". After the introduction of ConceptNet, each choice retrieves multiple paths in ConceptNet, and the model could not accurately determine which choice is most relevant to the question. Therefore, the model (-PN) get the wrong answer "**wallet**". Then, we introduce the paraphrase descriptions from WordNet and Wiktionary. The model can known that "**penny** is a coin" and "**piggy bank** can store coins" through paraphrase descriptions. Combined with the question, the model (HCP) can get the correct answer "**piggy bank**". Similarly, in the second example, the model can know from the paraphrase descriptions that "the period of the successive appearances of the moon is 29.531 days", which is closest "30 days". Combining the question, the model (HCP) gives the correct answer "**30 days**". Finally, to further analyze the model, we visualize the process of HCP answering the first example based on the attention weights between nodes in the HPP graph during the reasoning process, as shown in Fig. 6.

Fig. 6 shows the results of the visualization of example 1. Analysis of the two attentional visualizations shows that the attentional weight of one-hop path "*penny* $\xrightarrow{atlocation}$ *piggy bank*

TABLE XI
EXAMPLES FROM COMMONSENSEQA AND OPENBOOKQA, EACH EXAMPLE GIVES KNOWLEDGE PATHS AND ENTITY PARAPHRASES EXTRACTED FROM KBS

CommonsenseQA Question: Where can I put this **penny** to **save** for later ? **Choices:** A.**piggy bank** B.wallet C.toy D.ground E.pocket
Paths: penny $\xrightarrow{atlocation}$ piggy bank; penny $\xrightarrow{relatedto}$ dollar $\xrightarrow{atlocation}$ piggy bank; save $\xrightarrow{relatedto}$ money box $\xrightarrow{relatedto}$ piggy bank; penny $\xrightarrow{atlocation}$ wallet; ...
Paraphrases: **penny:** a coin worth one-hundredth of the value of the basic unit. **save:** accumulate money for future use. **piggy bank:** a child's coin bank (often shaped like a pig). **wallet:** a pocket-size case for holding papers and paper money.
Model prediction: HCP: A ☑ -PN: B ☒ -VM: B ☒ -QP: B ☒ -HPP: E ☒

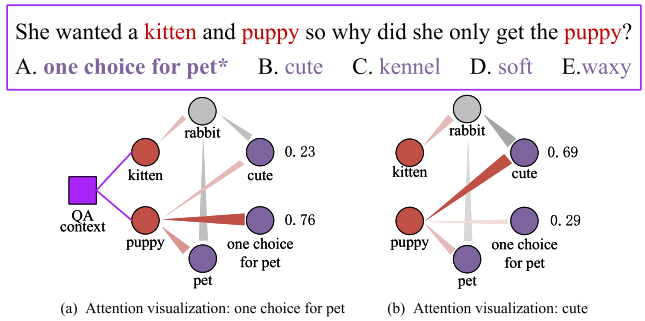
OpenBookQA Question: Each of the **moon**'s phases usually occurs once per **Choices:** A.week B.day C.**30 days** D.year
Paths: moon $\xrightarrow{relatedto}$ calendar month $\xrightarrow{partof}$ week; moon $\xrightarrow{relatedto}$ sky $\xrightarrow{atlocation}$ day; moon $\xrightarrow{relatedto}$ month $\xrightarrow{relatedto}$ days; moon $\xrightarrow{relatedto}$ lunar year $\xrightarrow{isa}$ year; ...
Paraphrases: **moon:** the period between successive new moons (29.531 days). **day:** the period of time taken by a particular planet (e.g. mars) ... **week:** hours or days of work in a calendar week. **year:** a period of time occupying a regular part of a calendar year ...
Model prediction: HCP: C ☑ -PN: A ☒ -VM: B ☒ -QP: D ☒ -HPP: D ☒

CommonsenseQA Question: She wanted a **kitten** and **puppy** so why did she only get the **puppy** ? **Choices:** A.**one choice for pet** B.cute C.kennel D.soft E.waxy
Paths: puppy $\xrightarrow{hasproperty}$ one choice for pet; puppy $\xrightarrow{capableof}$ pet; kitten $\xrightarrow{relatedto}$ rabbit $\xrightarrow{capableof}$ pet; puppy $\xrightarrow{hasproperty}$ cute; kitten $\xrightarrow{relatedto}$ rabbit $\xrightarrow{isa}$cute; ...
Paraphrases: **kitten:** young domestic cat. **puppy:** a young dog. **pet:** a domesticated animal kept for companionship or amusement. **cute:** attractive especially by means of smallness or prettiness or quaintness.
Model prediction: HCP: A ☑ -PN: A ☑ -VM: B ☒ -QP: B ☒ -HPP: B ☒

The each model uses the same knowledge paths and entity paraphrases.

" is greater than that of "*penny* $\xrightarrow{atlocation}$ *wallet* ". The attention weight of Fig. 6(a) is also significantly higher than that of Fig. 6(b) in the two-hop paths (e.g. "*penny* $\xrightarrow{relatedto}$ *coin* $\xrightarrow{atlocation}$ *piggy bank* " and "*penny* $\xrightarrow{relatedto}$ *coin* $\xrightarrow{atlocation}$ *wallet* "). In addition, we can also explain the HCP's reasoning process by analyzing the attention weights among the nodes of the HPP graph.

In the third example, The "kitten and puppy are cute" is more general commonsense knowledge. Without considering the QA context in the reasoning process, the model (-QP) would easily choose "**cute**" as the correct answer. Moreover, when too many knowledge paths are incorporated into the QA context resulting in changing the meaning of the QA context itself, the model (-VM) does not answer the question correctly. Therefore, we incorporate the QA context into the HPP graph through the visible matrix $\hat{M}$ in the PTLM's encoding phrase, so that the model (HCP) and model (-PN) can pay more attention to



(a) Attention visualization: one choice for pet (b) Attention visualization: cute

Fig. 7. Attention visualization graph of example 3, the (a) fuses the representation of the QA context into the HPP graph, and (b) does not fuse the QA context. Darker and thicker edges indicate higher attention weights.

the nodes related to the question during the reasoning process and get the correct answer "**one choice for pet**". Similarly, we visualize example 3 by attention weights as shown in Fig. 7.

C. Error Analysis

We randomly select 100 prediction errors examples and analyse them. As shown in Table XII, there are four main types of error examples.

Inappropriate paraphrase descriptions: For most key entities, their paraphrases are not unique. Although we filtrate the entity paraphrases based on POS tags and QA context, there are still some entities whose paraphrases are inappropriate. For example, the question in example 1 is about "Why does Sarah drop marbles?", and the paraphrase of "marble" in the current context should be "a small ball of glass that is used in various games.", but the paraphrase extracted by our method is "a sculpture carved from marble.", which is obviously inconsistent with the context of the current question.

Incomprehensible questions: When analyzing the error examples, we found that our method is difficult to get the correct answer when the questions are long and the scenes described are complex and abstract. For example, a "kite flying" scene is described in example 2 and asks "Where does the kite string start from?". We retrieves the corresponding paths and entity paraphrases in the KBs, but get the wrong answer. The question seems simple to humans, who can easily model the scene and associate "the string of the kite" to be in the "human hand". But machines lack the ability to understand complex questions and model complex scenes, which may lead to biased reasoning and wrong answers.

Indistinguishable knowledge paths: We found that the knowledge paths between many choices in the error examples are indistinguishable. In some examples, our method predicts answers that also accord with human commonsense. For example, the question in example 3 is about "If a person is a stranger, what might you treat him like?". Our method gives the answer "friend", while the correct answer for the dataset labeling is "family". In the ConceptNet, the corresponding knowledge paths can be found in both "friend" and "family". Meanwhile, both "friend" and "family" seem to accord with human commonsense and cognition.

TABLE XII
ERROR ANALYSIS, WE DIVIDE THE ERROR DATA INTO FOUR CATEGORIES, WHERE ☑ AND ✓ DENOTE CORRECT ANSWER AND MODEL'S PREDICTED ANSWER, RESPECTIVELY

Error type	Examples	
Inappropriate paraphrase (18/100)	Question	Sarah dropped the marble because she wanted to do what?
	Choices	A.game ☑ B.pouch C. home D.store E.jar ✓
	Paths for correct answer	marble $\xrightarrow{causes}$game; marble $\xrightarrow{usedfor}$playing marbles$\xrightarrow{isa}$game;...
	Paths for predicted answer	marble $\xrightarrow{atlocation}$jar; marble$\xrightarrow{atlocation}$store$\xrightarrow{atlocation}$jar;...
	Correct Paraphrase description	marble: a small ball of glass that is used in various games.
	Inappropriate Paraphrase description	marble: a sculpture carved from marble.
Incomprehensible questions (24/100)	Question	James saw a kite flying in the sky. He traced the string back to its origin and found it. Where did the string begin?
	Choices	A.end of line ✓ B.hobby shop C.his hand D.toy store E.child's hand ☑
	Paths for correct answer	kite $\xrightarrow{atlocation}$ child's hand; kite $\xrightarrow{relatedto}$ available $\xrightarrow{relatedto}$hand;...
	Paths for predicted answer	kite $\xrightarrow{atlocation}$ end of line; kite $\xrightarrow{capableof}$fly$\xrightarrow{relatedto}$line;...
	Paraphrase descriptions	kite: plaything consisting of a light frame covered with tissue paper... line: a lightweight cord. ...
Indistinguishable knowledge paths (18/100)	Question	Sam was a stranger. Even so, Mark treated him like what?
	Choices	A.friend ✓ B.known person C.family ☑ D.park E. outsider
	Paths for correct answer	stranger$\xrightarrow{relatedto}$alien$\xrightarrow{relatedto}$family; stranger $\xrightarrow{antonym}$friend $\xrightarrow{relatedto}$family; ...
	Paths for predicted answer	stranger $\xrightarrow{antonym}$friien; stranger $\xrightarrow{relatedto}$buddy$\xrightarrow{relatedto}$friend; ...
	Paraphrase descriptions	family: a person having kinship with another or others. friend: a person with whom you are acquainted. ...
Lack of relevant knowledge (36/100)	Question	Where may you be if you're buying pork chops at a corner shop?
	Choices	A. england ✓ B. town C. desert D. kentucky E.iowa ☑
	Paths for correct answer	corner shop $\xrightarrow{atlocation}$iowa; corner shop $\xrightarrow{atlocation}$minnesota $\xrightarrow{relatedto}$ iowa;...
	Paths for predicted answer	corner shop $\xrightarrow{atlocation}$ england; corner shop $\xrightarrow{atlocation}$ street $\xrightarrow{relatedto}$ england;...
	Paraphrase descriptions	corner shop:a small retail store, often in a residential area, that carries... iowa: a state in midwestern united states, with capital des moines. ...

Lack of critical knowledge: Although we use multiple KBs, they still does not cover all the knowledge. For example, example 4 mentions "where you might be when you buy pork chops?". The choices A and E both retrieve knowledge paths in ConceptNet, and it is difficult to distinguish which choice is correct by paraphrase's description. We can know that the iowa pork chop is very famous and the iowa pork chop is called "the pork version of the porterhouse steak" by using knowledge from the internet. Therefore, the choice E is the most likely correct answer. However, the ConceptNet, WordNet, and Wiktionary do not contain this knowledge.

With the above error analysis, we think that introducing higher quality external knowledge and improving the machine's ability to model and understand complex questions are essential to improve the machine's commonsense reasoning level.

VI. CONCLUSION

In this paper, we propose a heterogeneous-graph reasoning with context paraphrase method, which solves the CQA task by introducing multiple knowledge bases. Our key innovations include (i) introducing Wiktionary and WordNet into key entities recognition and HPP graph construction, reducing the irrelevant knowledge paths and pruning the HPP graph. Meanwhile, the paraphrase descriptions in Wiktionary and Wordnet provides context knowledge for HPP graph, which better facilitates the model to understand the HPP graph; (ii) fusing QA context and HPP graph in the PTLM's encoding phase, which achieves more efficient joint reasoning. Experiments and analyses on multiple CQA datasets demonstrate the effectiveness and generalisability of our proposed method. Moreover, our method also has some limitations, including: (i) the QA context, paths and paraphrases are encoded using PTLMs, which can consume more GPU resources; (ii) the entity embedding and paraphrase embedding are obtained through a simple pooling network, and the implicit knowledge in PTLMs is not fully utilized; (iii) some entity paraphrase is not appropriate for the currently given QA context. In the future, we will use Prompt-learning [44] and Adapter [45] to learn better entity embedding and paraphrase embedding. At the same time, we will also adopt more effective paraphrase extraction method to obtain more appropriate paraphrase descriptions.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2019, vol. 1, pp. 4171–4186.

[2] Y. Liu et al., "RoBERTa: A robustly optimized BERT pretraining approach," 2019, *arXiv:1907.11692*.

[3] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: A lite BERT for self-supervised learning of language representations," in *Proc. 8th Int. Conf. Learn Representations*, 2020, pp. 1–17.

[4] Y. Cui, W. Che, T. Liu, B. Qin, and Z. Yang, "Pre-training with whole word masking for chinese BERT," *IEEE/ACM Trans. Audio. Speech Lang. Process.*, vol. 29, pp. 3504–3514, 2021, doi: 10.1109/TASLP.2021.3124365.

[5] M. H. Sap, D. Rashkin, R. C. Bras, and Y. Choi, "Social IQa: Commonsense reasoning about social interactions," in *2019 Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 4463–4473.

[6] B. Zhou, D. Khashabi, Q. Ning, and D. Roth, "Going on a vacation takes longer than going for a walk: A study of temporal commonsense understanding," in *2019 Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 3363–3369.

[7] Y. Bisk, R. Zellers, R. LeBras, J. Gao, and Y. Choi, "PIQA: Reasoning about physical commonsense in natural language," in *Proc. 34th AAAI Conf. Artif. Intell.*, 2020, pp. 7432–7439.

[8] A. Talmor, J. Herzig, N. Lourie, and J. Berant, "CommonsenseQA: A question answering challenge targeting commonsense knowledge," in *Proc. 2019 Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, vol. 1, 2019, pp. 4149–4158.

[9] K. Bollacker, C. Evans, P. Paritosh, T. Sturge, and J. Taylor, "Freebase: A collaboratively created graph database for structuring human knowledge," in *2008 ACM SIGMOD Int. Conf. Manage. Data*, 2008, pp. 1247–1250.

[10] M. Sap et al., "ATOMIC: An atlas of machine commonsense for if-then reasoning," in *Proc. 33rd AAAI Conf, Artif. Intell, 2019, 31st Innov. Appl. Artif. Intell. Conf. 9th AAAI Symp. Educ. Adv. Artif. Intell.*, 2019, pp. 3027–3035.

[11] R. Speer, J. Chin, and C. Havasi, "ConceptNet 5.5: An open multilingual graph of general knowledge," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017, pp. 4444–4451.

[12] B. Lin, X. J. C. Chen, and X. Ren, "KagNet: Knowledge-aware graph networks for commonsense reasoning," in *Proc. Conf. Empirical Methods Natural Lang. Process., 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 2829–2839.

[13] Y. Feng et al., "Scalable multi-hop relational reasoning for knowledge-aware question answering," in *Proc. Conf. Empir. Methods Natural Lang. Process.*, 2020, pp. 1295–1309. [Online]. Available: https://aclanthology.org/2020.emnlp-main.99

[14] M. Yasunaga, H. Ren, A. Bosselut, P. Liang, and J. Leskovec, "QA-GNN: Reasoning with language models and knowledge graphs for question answering," in *2021 Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2021, pp. 535–546. [Online]. Available: https://aclanthology.org/2021.naacl-main.45

[15] Y. Sun, Q. Shi, L. Qi, and Y. Zhang, "JointLK: Joint reasoning with language models and knowledge graphs for commonsense question answering," in *2022 Conf. North Amer. Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, 2022, pp. 5049–5060.

[16] F. Scarselli, M. Gori, A. C. Tsoi, M. Hagenbuchner, and G. Monfardin, "The graph neural network model," *IEEE Trans. Neural Netw.*, vol. 20, no. 1, pp. 61–80, Jan. 2009.

[17] M. Grootendorst, "KeyBERT: Minimal keyword extraction with BERT," Zenodo, 2020. Accessed: Jul. 11, 2022, doi: 10.5281/zenodo.4461265.

[18] GA. Miller, "WordNet: A lexical database for english," *ACM Commun.*, vol. 38, pp. 39–41, 1995, doi: 10.1145/219717.219748.

[19] T. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," in *Proc. 5th Int. Conf. Learn. Representations*, 2017, pp. 1–14.

[20] K. Wang, W. Shen, Y. Yang, X. Quan, and R. Wang, "Relational graph attention network for aspect-based sentiment analysis," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 3229–3238.

[21] J. Lim, D. Oh, Y. Jang, K. Yang, and H. Lim, "I know what you asked: Graph path learning using AMR for commonsense reasonin," in *Proc. 28th Int. Conf. Comput. Linguistics*, 2020, pp. 2459–2471.

[22] S. Lv et al., "Graph-based reasoning over heterogeneous external knowledge for commonsense question answering," in *Proc. 34th AAAI Conf. Artif. Intell., 32nd Innov. Appl. Artif. Intell. Conf., 10th AAAI Symp. Educ. Adv. Artif. Intell.*, 2020, pp. 8449–8456.

[23] Q. F. Chen, H. J. Chen, and Y. Zhang, "Improving commonsense question answering by graph-based iterative retrieval over multiple knowledge sources," in *Proc. 28th Int. Conf. Comput.*, 2020, pp. 8449–8456.

[24] Y. C. Xu, R. Zhu, Y. Xu, M. L. Zeng, and X. Huang, "Fusing context into knowledge graph for commonsense reasoning," in *Proc. Findings Assoc. Comput. Linguistics, ACL-IJCNLP 2021*, 2020, pp. 1201–1207. [Online]. Available: https://aclanthology.org/2021.findings-acl.102.pdf

[25] N. Reimers and I. Gurevych, "Sentence-BERT: Sentence embeddings using siamese BERT-Networks," in *2019 Conf. Empirical Methods Natural Lang. Process. 9th Int. Joint Conf. Natural Lang. Process.*, 2019, pp. 3980–3990.

[26] W. Liu et al., "K-BERT: Enabling language representation with knowledge graph," in *Proc. 34th AAAI Conf. Artif. Intell. 2020, 32nd Innov. Appl. Artif. Intell. Conf. 2020, 10th AAAI Symp. Educ. Adv. Artif. Intell.*, 2020, pp. 2901–2908.

[27] N. Y. Shao, T. Cui, S. L. Wang, and G. Hu, "Is graph structure necessary for multi-hop question answering?," in *2020 Conf. Empirical Methods Natural Lang. Process.*, 2020, pp. 7187–7192. [Online]. Available: https://aclanthology.org/2020.emnlp-main.583

[28] T. Mihaylov, P. Clark, T. Khot, and A. Sabharwal, "Can a suit of armor conduct electricity? a new dataset for open book question answering," in *2018 Conf. Empirical Methods Natural Lang. Process.*, 2018, pp. 2381–2391.

[29] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *Proc. 7th Int. Conf. Learn. Representations*, 2019, pp. 1–18.

[30] A. Santoro et al., "A simple neural network module for relational reasoning," in *Proc. Adv. Neural Inf. Process. Syst.*, 2017, pp. 4967–4976.

[31] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. Berg, I. Titov, and M. Welling, "Modeling relational data with graph convolutional networks," in *Proc. Semantic Web Int. Conf.*, 2018, vol. 10843, pp. 593–607.

[32] X. Wang et al., "Improving natural language inference using external knowledge in the science questions domain," in *Proc. 33rd AAAI Conf. Artif. Intell, 2019 31st Innov. Appl. Artif. Intell. Conf. 9th AAAI Symp. Educ. Adv. Artif. Intell.*, 2019, pp. 7208–7215.

[33] X. Zhang et al., "GreaseLM: Graph reasoning enhanced language models for question answering," in *Proc. 10th Int. Conf. Learn. Representations*, 2022, pp. 1–16.

[34] C. Zhu, Y. Cheng, Z. Gan, S. Sun, T. Goldstein, and J. Liu, "FreeLB: Enhanced adversarial training for natural language understanding," in *Proc. 8th Int. Conf. Learn Representations*, 2020, pp. 1–14.

[35] K. Ma, J. Francis, Q. Lu, E. Nyberg, and A. Oltramari, "Towards generalizable neuro-symbolic systems for commonsense question answering," in *Proc. 1st Workshop Commonsense Inference Natural Lang. Process.*, 2019, pp. 22–32.

[36] P. Wang, N. Peng, F. Ilievski, P. Szekely, and X. Ren, "Connecting the dots: A knowledgeable path generator for commonsense question answering," in *Proc. Findings Assoc. Comput. Linguistics, EMNLP 2020*, 2020, pp. 4129–4140. [Online]. Available: https://aclanthology.org/2020.findings-emnlp.369.pdf

[37] P. Clark et al., "From 'F' to 'A' on the N. Y. regents science exams: An overview of the Aristo project," *AI Mag.*, vol. 41, pp. 39–53, 2020.

[38] P. Banerjee, K. Kumar Pal, A. Mitra, and C. Baral, "Careful selection of knowledge to solve open book question answering," in *Proc. 57th Conf. Assoc. Comput. Linguistics*, 2019, vol. 1, pp. 6120–6129.

[39] P. Banerjee and C. Baral, "Knowledge fusion and semantic knowledge ranking for open domain question answering," 2020, *arXiv:2004.03101*.

[40] C. Raffel et al., "Exploring the limits of transfer learning with a unified text-to-text transformer," *J. Mach. Learn. Res.*, vol. 21, 2020, pp. 1–67.

[41] D. Khashabi, "Unifiedqa: Crossing format boundaries with a single QA system," in *Proc. Findings Assoc. Comput. Linguistics, EMNLP 2020*, 2020, pp. 1896–1907. [Online] Available: https://aclanthology.org/2020.findings-emnlp.171

[42] R. Zellers, A. Holtzman, Y. Bisk, A. Farhadi, and Y. Choi, "HellaSwag: Can a machine really finish your sentence?," in *Proc. 57th Conf. Assoc. Comput. Linguistics*, 2019, pp. 4791–4800.

[43] H. Touvron et al., "Llama 2: Open foundation and fine-tuned chat models," 2023, *arXiv:2307.09288*.

[44] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, and G. Neubig, "Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing," in *Proc. ACM Comput. Surv.*, vol. 55, 2023, pp. 195:1–195:35.

[45] J. He, C. Zhou, X. Ma, and T. B. Kirkpatrick, and G. Neubig, "Towards a unified view of parameter-efficient transfer learning," in *Proc. 10th Int. Conf. Learn. Representations*, 2022, pp. 1–15.

Yujie Wang received the B.S. degree from the Zhengzhou University of Light Industry, Zhengzhou, China, in 2019. He is currently working toward the Ph.D. degree with the School of Computer and Information Technology, Shanxi University, Taiyuan, China. His research interests include natural language processing, information retrieval, and deep learning.

Hu Zhang received the Ph.D. degree from Shanxi University, Taiyuan, China. He is currently a Professor with the School of Computer and Information Technology, Shanxi University. He has authored or coauthored more than 80 papers in his research fields, including ACL, EMNLP, Semantic Web, TALLIP, ICONIP. His research interests include Big Data analysis, natural language processing, and artificial intelligence.

Jiye Liang (Senior Member, IEEE) received the Ph.D. degree from Xian Jiaotong University, Xi'an, China. He is currently a Professor with the Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, the School of Computer and Information Technology, Shanxi University, Taiyuan, China. He has authored or coauthored more than 300 papers in his research fields, including AI, JMLR, IEEE TPAMI, IEEE TKDE, ML, NeurIPS, ICML, and AAAI. His research interests include artificial intelligence, granular computing, data mining, and machine learning.

Ru Li (Member, IEEE) received the Ph.D. degree from Shanxi University, Taiyuan, China. She is currently a Professor with the Key Laboratory of Computational Intelligence and Chinese Information Processing of Ministry of Education, the School of Computer and Information Technology, Shanxi University. She has authored or coauthored more than 100 papers in her research fields, including IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, ACL, IJCAI, AAAI, EMNLP, and COLING. Her research interests include artificial intelligence, natural language processing, and frame semantics.