Information Sciences 660 (2024) 120134

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Information bottleneck based knowledge selection for commonsense reasoning

Zhao Yang [a,b], Yuanzhe Zhang [a,b], Pengfei Cao [a,b], Cao Liu [c], Jiansong Chen [c], Jun Zhao [a,b], Kang Liu [a,b,d,*]

[a] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing, China
[b] The Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[c] Meituan, Beijing, China
[d] Shanghai Artificial Intelligence Laboratory, Shanghai, China

ARTICLE INFO

Keywords:
Commonsense reasoning
Knowledge selection
Information bottleneck
KG-augmented model

ABSTRACT

KG-augmented models usually endow existing models with external knowledge graphs, which achieve promising performance in various knowledge-intensive tasks, such as commonsense reasoning. Existing methods mainly first exploited heuristic ways for retrieving the relevant knowledge subgraphs according to the input, and then utilized some effective encoders, such as GNNs, to encode the symbolic knowledge into the neural reasoning networks. However, whether the whole retrieved knowledge subgraphs are really relevant or useful for the reasoning process was seldom considered. Actually, according to our observations and analysis, most retrieved knowledge is noisy and useless to the reasoning models, which would hurt the final performance. To remedy this, this paper proposes information bottleneck based knowledge selection (**IBKS**), which is able to select useful knowledge from the retrieved knowledge subgraph. Expectedly, the selected knowledge could better improve the commonsense reasoning ability of the model. Moreover, **IBKS** is model-agnostic and could be plugged into any existing KG-augmented model. Extensive experimental results show that IBKS could effectively improve commonsense reasoning performance.

1. Introduction

Knowledge underpins reasoning. To verify such ability, researchers recently proposed the task of commonsense reasoning [15,30,21,39] and designed various knowledge graph (KG) augmented models to solve this task [16,6,40]. They usually leverage external commonsense knowledge from existing KGs to empower existing neural reasoning models with enough knowledge background [45].

The basic architecture of existing KG-augmented models, as illustrated in Fig. 1, follows a 3-step injection paradigm for incorporating external knowledge into the neural models [45]. 1) Knowledge Retrieving: they usually attempted to design heuristic methods [16] to retrieve a knowledge subgraph that is relevant to the original textual input. 2) Knowledge Encoding: they utilized encoders like graph neural networks (GNNs) to encode the retrieved knowledge subgraph to obtain the knowledge representations. 3) Knowl-

* Corresponding author at: The Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

E-mail addresses: zhao.yang@nlpr.ia.ac.cn (Z. Yang), yzzhang@nlpr.ia.ac.cn (Y. Zhang), pengfei.cao@nlpr.ia.ac.cn (P. Cao), liucao@meituan.com (C. Liu), chenjiansong@meituan.com (J. Chen), jzhao@nlpr.ia.ac.cn (J. Zhao), kliu@nlpr.ia.ac.cn (K. Liu).

https://doi.org/10.1016/j.ins.2024.120134
Received 29 June 2023; Received in revised form 24 November 2023; Accepted 10 January 2024

Available online 15 January 2024

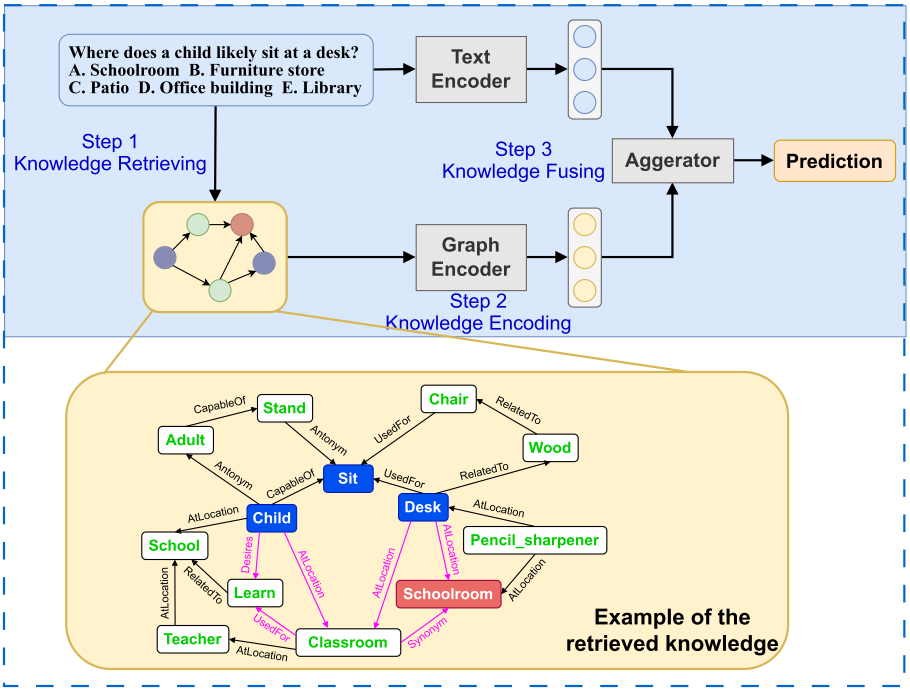


Fig. 1. Existing KG-augmented models could be summarized as this architecture. Step 1 finds relevant knowledge according to the text. Step 2 encodes the knowledge subgraph to obtain the knowledge representations and step 3 fuses the knowledge representations and text representations. And we show the part of the obtained knowledge using MHGRN [6].

Table 1
Annotation results of the 50 samples. Total concept refers to the number of introduced concepts of existing heuristic methods. Useful concept denotes the number of annotated potentially useful concepts. Accuracy (Ori) and Accuracy (Useful) stand for the accuracy of the 50 sampled examples in terms of providing total concepts and useful concepts, respectively.

Method	Total Concept	Useful Concept	Ratio	Accuracy (Ori)	Accuracy (Useful)
MHGRN [6]	572.84	152.76	0.27	72.0	78.0
QA-GNN [40]	534.95	133.84	0.25	80.0	84.0

edge Fusing: they aggregated the knowledge representations and original textual representations together for predictions. In this process, learning better knowledge-augmented representations is very important for reasoning. To this end, previous studies mainly focused on the last two steps (knowledge encoding and knowledge fusing), i.e. obtaining better knowledge representations [16,6,40] and fusing the knowledge representations with the textual representations well [43,45].

However, previous studies [16,6,40,43] all ignore the quality of the retrieved knowledge subgraph (i.e. step 1), which is actually unsatisfactory. As shown in Fig. 1, the knowledge subgraph obtained by heuristic methods is extensive and redundant. Only the subgraph connected with the pink line is useful and the other parts are useless for reasoning. To further illustrate, we choose *CommonsenseQA* [30] which is a widely used dataset for commonsense reasoning to make a quantitative analysis. We analyze the results from existing heuristic KG-augmented methods [6,40] by randomly sampling 50 instances and inviting three postgraduates are invited to annotate the potentially useful concepts. Table 1 presents the statistics of the sampled instances and the annotation results. We observe that less than 30% of the introduced concepts are useful, which demonstrates that the retrieved subgraph contains a substantial amount of useless knowledge. Intuitively, if the quality of the selected knowledge is poor, it will not be helpful for the subsequent reasoning models. Thus we test these 50 examples with these two KG-augmented models. Accuracy (Ori) and Accuracy (Useful) represent the accuracy when providing the total concepts and the useful concepts, respectively. The performance gaps verify that the redundant knowledge indeed hurts the following reasoning performance, which is also consistent with the findings in [2].

To filter the retrieved knowledge noises, existing KG-augmented reasoning models try some solutions in the knowledge encoding step (step 2). For example, they employed GNNs or attention mechanism on knowledge selection. However, recent studies have shown that GNNs are not competent according to their core message-passing mechanism [41,27], i.e., the message of all k-hop neighbors would be encoded into the node representation for a k-layer GNN. Moreover, as for the most anticipated attention mechanism, recent studies also proved that the soft attention value could not filter information but scale information in GNNs [41,42,20]. As a result, the existing efforts in step 2 [16,6,40,43] could only encode retrieved knowledge instead of filtering the noisy information.

Different from previous approaches, this paper focuses on selecting useful knowledge in the first step, i.e. knowledge retrieving. To achieve this, we borrow the idea of the information bottleneck [31,1], which seeks a representation that is maximally informative about the prediction while being minimally informative about the original input data. That is to say, information bottleneck (IB) provides a convenient mechanism for penalizing an information-theoretic measure of redundant information in the original inputs. Based on this, IB is expected to filter useless knowledge which is irrelevant to the prediction. Thus, this paper proposes an Information Bottleneck based Knowledge Selection (**IBKS**) method. Specifically, motivated by Yu et al. [42] and Miao et al. [20], we apply IB on the graph data.[1] However, unlike their tasks, selecting knowledge subgraphs should consider both the textual inputs and the original retrieved graph. The textual inputs could provide more sufficient contextual information for precise knowledge selection. For example, for two different questions that have the same concepts, if we do not consider the contextual information, the selected subgraph would be the same, which is obviously unreasonable. To this end, we add a textual constraint in our selection model to select knowledge more precisely. Moreover, we additionally introduce task-related prior information in the selection process, that is the edges which could connect the concepts in the original text would be more important [9]. Accordingly, the reasoning performance could be improved further. Furthermore, considering the optimization objective of our proposed **IBKS** is not tractable, this paper utilizes variational inference to obtain its tractable upper bound for optimization. Besides, our proposed **IBKS** is model-agnostic, and we expatiate how to plug it into existing KG-augmented models in § 3.2.

Our contributions can be summarized as:

- In this paper, we first illustrate the importance of knowledge selection for commonsense reasoning. To better fit our task, we further propose **i**nformation **b**ottleneck based **k**nowledge **s**election (**IBKS**), which could effectively filter irrelevant and redundant noises from retrieved knowledge subgraphs and be beneficial for the subsequent reasoning models.
- **IBKS** could be easily plugged into any KG-augmented model and we plug it into three typical KG-augmented models. Extensive experimental results show our proposed method could improve the commonsense reasoning performance of existing KG-augmented models with only about 30% of knowledge preserved.

2. Related work

2.1. KG-augmented models for commonsense reasoning

KG-augmented models are proposed to address the problem of lacking enough commonsense knowledge in existing models. These models first employ heuristic methods to obtain a knowledge subgraph that is associated with the question and the answer from a huge external knowledge graph such as `ConceptNet` [28]. The heuristic methods usually match the tokens in the questions and answers to the mentioned concepts in the external knowledge graph. With these matched concepts, they can find a relevant subgraph covering all these concepts via subgraph matching [7] and path finding [16].

After obtaining the relevant knowledge sub-graph, KG-augmented models usually utilize GNNs like RGCN [26], and Gconattn [35] to encode the knowledge subgraph and fuse the knowledge representations with the textual representations which are encoded by PLMs. Relation Networks (RN) [25] is originally proposed to solve questions about the relations between multiple objects in an image. And the concepts in the inputs can be seen as objects and RN could be easily transferred into modeling the relations between concepts in external knowledge graphs. RN could model single-hop triplets well, which results in better knowledge representations. To introduce multi-hop relations, Kagnet [16] modeled the multi-hop relations by extracting relational paths from KGs and then encoding paths with LSTM, which leads to big performance improvements. To further model these multi-hop relations, MHGRN [6] modeled relational paths as multi-hop message passing with multi-layer graph attention networks. Previous studies all learn the knowledge representations in isolation, neglecting the critical role of text representation. To solve this problem, Yasunaga et al. [40] introduced the QA context node to represent text representation and added this node into the knowledge subgraph, which could obtain text-enhanced graph representations. To further enhance the interaction of the text representation and the knowledge representation, GREASELM [43] fused the text representations and the knowledge representations through multi-layered modality interaction operations, JointLK [29] utilized the bidirectional attention module to fuse these two representations. More recently, there are also some studies focusing on the quality of the retrieved knowledge. DGRN [44] added relevant edges to help in finding the chain of reasoning when there are missing edges in external KG. Our concurrent work DHLK [36] pruned the noisy knowledge according to the attention weights.

Previous studies ignore the quality of the heuristically obtained knowledge subgraph, which contains numerous useless knowledge. As a result, the obtained knowledge representations would be filled with noise information, which constrains the model performance [36].

2.2. Information bottleneck

Information bottleneck (IB) is originally proposed to find a short code of the input signal but preserve maximum information in signal processing [31]. Then Tishby et al. [32] first applied it in deep learning. And Alemi et al. [1] further proposed variational information bottleneck (VIB) to bridge the gap between IB and deep learning. In summary, IB aims to seek a trade-off between

[1] Our retrieved knowledge subgraphs are actually the graph data.

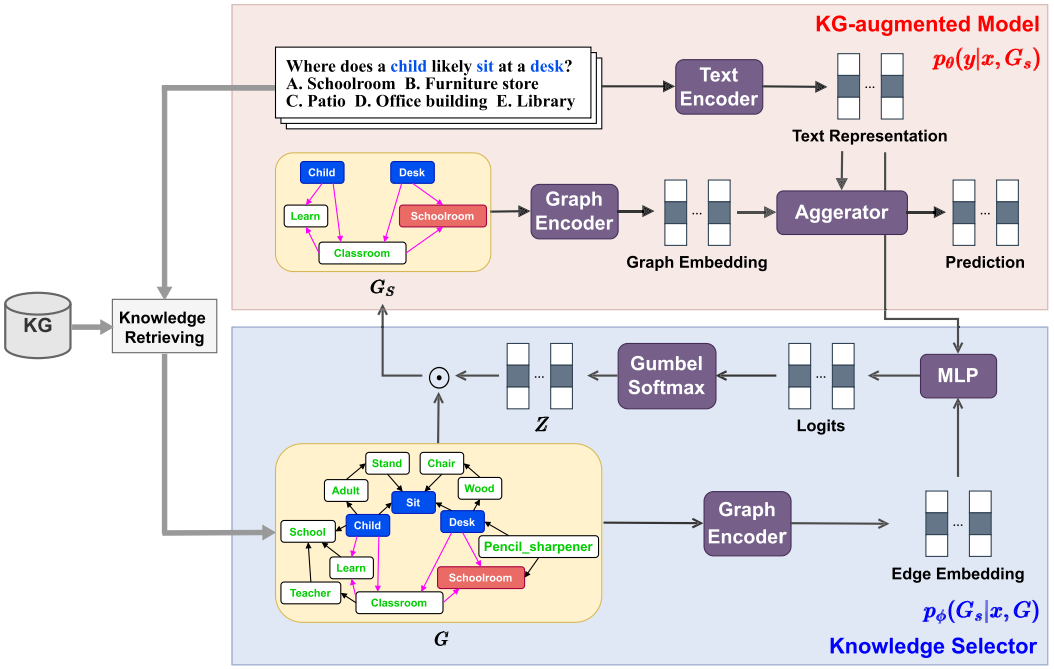


Fig. 2. Overall model architecture of **IBKS**. The blue box denotes the knowledge selector, which could select G_s with the constraints of text and G. The red box refers to the KG-augmented model.

maximizing predictive accuracy and minimizing the representation complexity, which could be applied to improving the model robustness and generalization [5,34]. More recently, [37] and [42] extended the general IB to irregular graph data and proposed graph information bottleneck (GIB), which could be applied to subgraph recognition problem [20]. However, in KG-augmented models, previous studies are not applicable because of the completely different probabilistic graph and the extra constraints of text. Fig. 3 presents the differences between these previous studies and our proposed method in detail.

3. Method

In this section, we first introduce the background of the task and the formulation of existing KG-augmented models. Then we illustrate how to implement our method. Finally, we present the target of **IBKS** and deduce a tractable lower bound for optimization.

3.1. Task definition of commonsense reasoning

In this paper, we focus on the multiple-choice commonsense reasoning task, which requires choosing the correct answer y from N candidate answers $\{a_1, a_2, \cdots, a_N\}$ based on the question q. And we denote the question and all answers as the textual input x. The target is to maximize $p(y|x)$.

Existing KG-augmented models usually introduce relevant knowledge G from the external knowledge graph $\mathcal{G}$ to help reason. These models share a similar architecture (Fig. 1) and the target of these models is to maximize $p(y|x, G)$.

3.2. Overview of KG-augmented models with IBKS

In this part, we would introduce the implementation of **IBKS**. Fig. 2 shows the model architecture, which consists of the knowledge selector and KG-augmented model. This paper aims to select G_S from the original G to improve existing models. The target of the knowledge selector is $p_\phi(G_S|G, x)$. Therefore, the original target $p(y|x, G)$ could be decomposed as $p_\phi(G_S|G, x)p_\theta(y|x, G_S)$, where ϕ and θ refer to the parameters of the knowledge selector and the KG-augmented model, respectively. And we introduce these two components in the following parts.

Knowledge Selector is depicted in the blue box, which could be formulated as $p_\phi(G_S|G, x)$. This module aims to select a knowledge subgraph G_S[2] from the original retrieved knowledge graph G.

[2] In this paper, we obtain G_S by selecting useful edges. And the isolated node would naturally be filtered. Compared to selecting useful nodes, edge selection is more refined and more effective [42,20].

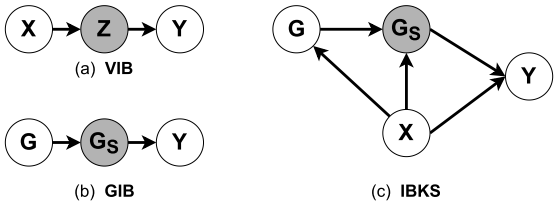


Fig. 3. Probabilistic graphical model of VIB [1], GIB [42] and IBKS.

Firstly, we utilize the pre-trained language model to encode the textual input x to obtain the textual representation $\boldsymbol{T}$. Then we use the graph encoder[3] to obtain the edge embedding[4] $\boldsymbol{E} = (\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_{N_e})$, where N_e refers the number of edges. Then we denote a binary embedding $\boldsymbol{Z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \cdots, \boldsymbol{z}_{N_e})$ to represent which edges would be reserved, where $\boldsymbol{z}_i = 1$ indicates the corresponding i-th edge would be reserved. This binary embedding $\boldsymbol{Z}$ can be computed as follows:

$$\boldsymbol{Z} = \text{Gumbel\_Softmax}(\text{MLP}(\boldsymbol{E}, \boldsymbol{T})) \tag{1}$$

where MLP refers to a 3-layer perception and Gumbel_Softmax refers to the **reparameterization method** [8]. With this binary embedding $\boldsymbol{Z}$, we can easily obtain the corresponding subgraph $G_S = \boldsymbol{Z} \odot G$. However, the modification of the graph input does not fit the end-to-end training paradigm and has adverse effects on GNNs [24]. Thus we do not modify the original knowledge graph input and turn to modify the message-passing process, which is the core mechanism of GNN. In specific, we let the message passed by the i-th edge be zero when $z_i = 0$ and keep the passing message unchanged when $z_i = 1$.

KG-augmented Model is shown in the red box, which refers to $p_\theta(y|x, G_S)$. And existing KG-augmented models all can be summarized as this formulation. With this part, we can judge the quality of the selected G_S according to the loss between the output probability and the ground truth label y. This signal could help the knowledge selector adaptively select a more label-relevant subgraph.

Obviously, the knowledge selector could be plugged into any previous KG-augmented model. Therefore, **IBKS** is model-agnostic and could be applied to improving any KG-augmented model.

3.3. Information bottleneck based knowledge selector

As shown in Fig. 3b, original GIB [42] seeks G_S from G and requires G_S minimally informative about G. The optimization objective of GIB could be formulated as follows:

$$\min -I(G_S; y) + \beta I(G; G_S) \tag{2}$$

where I denotes the mutual information and β is the weight to adjust the two terms. Obviously, this optimization objective ignores the impact of the textual inputs, which is not suitable for our task. Therefore, we need to introduce the textual constraint in the selection process.

IB with Textual Constraint As shown in Fig. 3c, we introduce the textual constraint to better model our task, which named **IBKS**. In specific, G_S is selected from G with the constraint of the text input x, and we expect G_S to contain as little information about G as possible. G_S would be fed into KG-augmented models to check its quality, and we expect G_S could help maximize the probability of obtaining the correct answer y. Therefore, the overall optimization target of **IBKS** can be formulated as:

$$\min -I(x, G_S; y) + \beta I(G; G_S|x) \tag{3}$$

where I denotes the mutual information and β is the weight to adjust the two terms. The first term $I(x, G_S; y)$ requires the mutual information between the label y and the input x fused with the selected knowledge G_S to be big enough. The second term $I(G; G_S|x)$ constrains the conditional mutual information between the original G and the selected knowledge subgraph G_S given the textual input x. Combining these two terms, **IBKS** could select optimal G_S from G, which satisfies that G_S entails less information of G but could provide more information about the label Y, which can be seen as useful knowledge.

The above optimization objective is obviously not tractable because of the mutual information terms. Thus, we utilize **variational inference** to obtain its tractable upper bound for optimization. In specific, we need to deduce the lower bound for $I(x, G_S; y)$ and the upper bound for $I(G; G_S|x)$. And we illustrate these details in the following parts.

For the first term, we expect G_S could add relevance between the input x and the ground truth label y. This term can be formulated as follows:

[3] Graph encoder in knowledge selector shares parameters with the graph encoder in KG-augmented model.

[4] Considering some GNNs do not have separate edge representations, we uniformly average the node representations of the start node and the end node as the edge representations.

$$\begin{aligned} I(X, G_S; Y) &= \int p(x, G_S, y) \log \frac{p(y|x, G_S)}{p(y)} dx dy dG_S \\ &= \int p(x, G_S, y) \log \frac{p_\theta(y|x, G_S)}{p(y)} dx dy dG_S \\ &\quad + \mathrm{KL}(p(y|x, G_S)||p_\theta(y|x, G_S)) \\ &\geq \int p(x, G_S, y) \log p_\theta(y|x, G_S) dx dy dG_S \end{aligned} \tag{4}$$

where $p(y|x, G_S)$ can not be estimated, thus we utilize $p_\theta(y|x, G_S)$ to be a varitional estimation of this distribution. And $p_\theta(y|x, G_S)$ could be seen as any existing KG-augmented model and θ denotes the model parameter.

Considering G_S is selected from G, we introduce G into the probability density to form $p(x, G_S, y, G)$. According to the probabilistic graph of **IBKS** (Fig. 3 (c)), we could decompose it as $p(x, y)p_\phi(G_S|x, G)p(G|x)$. Based on this, we could simplify the above term as follows:

$$\begin{aligned} & I(X, G_S; Y) \\ \geq & \int p(x, G_S, y, G) log p_\theta(y|x, G_S) dx dy dG_S dG \\ = & \int p(x, y)p_\phi(G_S|x, G)p(G|x) \log p_\theta(y|x, G_S) dx dy dG_S dG \end{aligned} \tag{5}$$

As for the second term $I(G; G_S|x)$, it requires the conditional mutual information between G and the selected knowledge subgraph G_S given the textual input x to be small. This term constrains the selected G_S to contain less information from the original G, which can be formulated as:

$$\begin{aligned} & I(G; G_S|X) \\ = & \int p(G, G_S, x) \log \frac{p(G_S|x, G)}{p(G_S|x)} dx dG dG_S \\ = & \int p(G, G_S, x) \log [\frac{p(G_S|x, G)}{r(G_S|x)}] dx dG dG_S \\ & - \mathrm{KL}[p(G_S|x)||r(G_S|x)] \\ \leq & \int p(G, G_S, x) \log \frac{p(G_S|x, G)}{r(G_S|x)} dx dG dG_S \end{aligned} \tag{6}$$

where the marginal distribution of $p(G_S|x)$ is difficult to compute. Thus we utilize a prior distribution $r(G_S|X)$ to be the variational estimation of this marginal distribution. In specific, $r(G_S|x) = \int r(G_S|G, x)p(G|x)dG$. For the heuristic method, $p(G|x)$ could be seen as a one-point distribution, which means G is deterministic for a given x. Therefore, $r(G_S|x) = r(G_S|G, x) = r(G * Z_G|G, x)$, where Z_G is a binary embedding that denotes whether the edge should be reserved.

Task-related Prior We illustrate the prior distribution of whether to reserve an edge in this part. For edge e, we sample $\alpha_e \sim \mathrm{Bern}(p)$, where $\mathrm{Bern}(p)$ denotes the Bernoulli distribution which sample 1 with the probability p. Then we retain e when $\alpha_e = 1$ and drop e when $\alpha_e = 0$. A recent study [9] has shown the edges which could connect question and answer would be more important. Inspired by this new finding, we further design the prior distribution as follows: For edge e which connects the question concept and answer concept, we sample $\alpha_e \sim \mathrm{Bern}(p_1)$. For the edges that do not serve as connectors, we sample $\alpha_e \sim \mathrm{Bern}(p_2)$, where p_1 is bigger than p_2. With this sampling strategy, the edges that could connect question concepts and answer concepts would be reserved with higher probability.

Considering the golden label y could help to judge the quality of G_S, we introduce y to form $p(G, G_S, x, y)$. Following Fig. 3 (c), we can decompose it as $p(x, y)p_\phi(G_S|x, G)p(G|x)$. Thus the above formula could be simplified as:

$$\begin{aligned} & I(G; G_S|X) \\ \leq & \int p(G, G_S, x, y) \log \frac{p(G_S|x, G)}{r(G_S|x)} dx dy dG dG_S \\ = & \int p(x, y)p_\phi(G_S|x, G)p(G|x) \log \frac{p_\phi(G_S|x, G)}{r(G_S|x)} dx dy dG dG_S \end{aligned} \tag{7}$$

where $p_\phi(G_S|x, G)$ denotes the knowledge selector, which could select G_S with the constraints of text x and original knowledge G. ϕ refers to the parameter of the knowledge selector.

Table 2
Stastics of the datasets *CommonsenseQA* and *OpenBookQA*.

Dataset	Train	Valid	Test
CommonsenseQA(IH)	8500	1221	1241
OpenbookQA	4957	500	500

Therefore, the original intractable optimization target could be computed as follows:

$$\begin{aligned} &-I(X,G_S;Y)+\beta I(G;G_S|X) \\ \le &-\int p(x,y)p_\phi(G_S|x,G)p(G|x)(\log p_\theta(y|x,G_S) \\ &+\beta\log\frac{p_\phi(G_S|G,x)}{r(G_S|x)})dxdydGdG_S \end{aligned} \tag{8}$$

In practice, we can approximate $p(x,y)$ with its empirical data distribution $p(x,y)\approx\frac{1}{N}\sum_{i=1}^{N}\delta_{x_i}(x)\delta_{y_i}(y)$. With the existing deterministic heuristic methods, G_i is also deterministic for a given x_i. Based on this, our optimization target could be formulated as:

$$\begin{aligned} &-I(X,G_S;Y)+\beta I(G;G_S|X) \\ \le &-\frac{1}{N}\sum_{i=1}^{N}[p_\phi(G_{S_i}|x_i,G_i)\log p_\theta(y_i|x_i,G_{S_i})+\beta\mathrm{KL}[p_\phi(G_{S_i}|x_i,G_i)||r(G_{S_i}|x_i)]] \end{aligned} \tag{9}$$

Combining these two terms, the selected G_S would have less information about the original G but could help the KG-augmented model more. Generally speaking, for a chosen KG-augmented model, the selected G_S is the label-relevant knowledge in the whole knowledge subgraph G.

According to Fig. 3, our designed **IBKS** diverge from existing representative IB studies, including VIB [1] and GIB [42]. The fundamental difference is the constraints of the textual input. When selecting G_S, we should consider the synergy of the textual input X and the original knowledge G. When judging the quality of G_S, we should also consider the mapping from the combination of G_S and X to Y.

4. Experiments

Firstly, we introduce the basic experimental setups in § 4.1. Then we list the corresponding experimental results and show analysis in § 4.2.

4.1. Experimental setup

4.1.1. Datasets

We evaluate all models on the two widely used commonsense reasoning datasets: *CommonsenseQA* [30] and *OpenBookQA* [26].

CommonsenseQA is a 5-way multiple choice QA task that requires reasoning with commonsense knowledge. And there are 12,102 questions in the dataset. Following previous studies, we conduct experiments on the in-house (IH) data splits [16]. *OpenBookQA* is a 4-way multiple choice QA task that requires reasoning with elementary science knowledge. There are 5,957 questions in the dataset. We conduct experiments on the official data splits [21]. And Table 2 presents the specific statistics of the data splits in our following experiments.

4.1.2. External knowledge graph

Following previous work, we utilize `ConceptNet` [28] as the external commonsense knowledge graph resource in our experiments. `ConceptNet` is a general-domain knowledge graph and consists of 799,273 nodes and 2,487,810 edges, which has been regarded as a good commonsense knowledge origin in various tasks.

4.1.3. Models

Existing KG-augmented models mainly share a similar architecture and we select three typical methods.

(1) RGCN [26] is one of the most typical KG-augmented models, which utilizes RGCN to encode the retrieved knowledge. Compared to common GCN [22], RGCN considers the important role of relations in a multi-relation graph additionally, which is better suited for the knowledge graph.

(2) MHGRN [6] addresses the limitation of previous methods that only modeled single-hop relationships and proposes multi-layer graph attention networks [33] to encode the multi-hop relation paths.

(3) QA-GNN [40] highlights the issue with previous methods that focused solely on learning isolated representations of knowledge while neglecting the role of text representations. QA-GNN introduces a context node to represent text representation and adds this node into the knowledge subgraph, which could obtain text-enhanced graph representations. With the text-enhanced knowledge representations, QA-GNN achieves the SOTA performances.

Table 3
The hyperparameter settings on the *CommonsenseQA* and *OpenBookQA* datasets.

Hyperparameter	CommonsenseQA	OpenbookQA
Learning Rate for BERT-base	3e-5	3e-5
Learning Rate for BERT-large	2e-5	2e-5
Learning Rate for Roberta-large	1e-5	1e-5
Learning Rate for Knowledge Encoder	1e-3	3e-4
Batch Size	32	32

Table 4
Performance comparison on *CommonsenseQA* [30]. We report in-house Dev (**IHdev**) and Test (**IHtest**) accuracy using the data splits of Lin et al. [16]. All results are reported with the mean and standard derivation of five runs.

	BERT-base		BERT-large		RoBerta-large	
	IHdev	IHtest	IHdev	IHtest	IHdev	IHtest
RGCN	56.94 (±0.38)	54.50 (±0.56)	62.98 (±0.82)	57.13 (±0.36)	72.69 (±0.19)	68.41 (±0.66)
+IBKS	58.68 (±1.02)	55.97 (±1.44)	64.34 (±1.01)	58.28 (±1.36)	74.01 (±1.27)	69.72 (±1.56)
MHGRN	60.36 (±0.23)	57.23 (±0.82)	63.29 (±0.51)	60.59 (±0.58)	74.45 (±0.10)	71.11 (±0.81)
+IBKS	61.37 (±0.86)	57.94 (±1.18)	64.42 (±0.97)	60.99 (±1.07)	75.42 (±0.83)	71.78 (±1.23)
QA-GNN	61.92 (±0.46)	58.85 (±0.89)	65.24 (±0.40)	61.34 (±0.72)	76.54 (±0.21)	73.41 (±0.92)
+IBKS	63.81 (±1.22)	59.87 (±1.65)	66.97 (±1.08)	62.33 (±1.53)	78.47 (±1.14)	74.47 (±1.77)

Table 5
Performance comparison on *OpenbookQA* [21]. We report the accuracy of the official dev and test datasets. All results are reported with the mean and standard derivation of five runs.

	BERT-base		BERT-large		RoBerta-large	
	Dev	Test	Dev	Test	Dev	Test
RGCN	51.12 (±2.22)	48.96 (±0.85)	58.20 (±1.30)	56.24 (±1.11)	64.65 (±1.96)	62.45 (±1.57)
+IBKS	52.57 (±1.97)	50.02 (±1.66)	59.33 (±1.55)	57.62 (±1.47)	66.47 (±2.04)	63.92 (±1.79)
MHGRN	55.77 (±1.13)	53.83 (±1.02)	59.46 (±0.55)	58.46 (±1.16)	68.10 (±1.02)	66.85 (±1.19)
+IBKS	56.65 (±1.08)	54.40 (±1.47)	60.05 (±1.02)	59.11 (±1.27)	69.02 (±1.15)	67.41 (±1.53)
QA-GNN	57.88 (±0.50)	56.20 (±1.77)	61.60 (±1.23)	60.22 (±1.55)	69.60 (±1.06)	67.51 (±0.58)
+IBKS	58.55 (±1.14)	56.31 (±1.75)	63.08 (±1.36)	61.17 (±1.84)	71.58 (±2.12)	68.93 (±1.86)

For these three KG-augmented models, we choose BERT-base, BERT-large, and RoBERTa-large as the text encoder, respectively. We plug our proposed **IBKS** on these selected models to show its effectiveness.

4.1.4. Implementation details

Following the previous studies [16,6], we reproduce the typical KG-augmented models RGCN, MHGRN, and QA-GNN with the official implementations. We use the Adam optimizer [12] to train our model and list the best-performing values of hyperparameters in Table 3. For β in formula (9), we select β from $\{1e-1, 1e-2, 1e-3, 1e-4, 1e-5\}$. As for the prior distribution, we select p_1 from $\{0.5, 0.6, 0.7, 0.8, 0.9\}$ and choose p_2 from $\{0.4, 0.3, 0.2\}$. All hyper-parameters are selected based on the validation set through a grid search. All experiments are conducted with an NVIDIA GeForce RTX 3090 Ti.

4.2. Experimental results

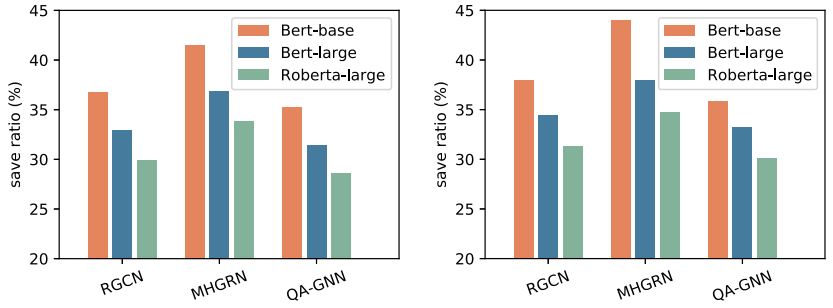
We divide the experiments into two parts. Firstly, we conduct various experiments on different KG-augmented models and different text encoders on different datasets. According to these experiments, we could illustrate the effectiveness of IBKS on these different conditions. Secondly, we adhere to the settings of the text encoder in the leaderboard and compare to a series of KG-augmented models. Besides, we also compare to the large language models (GPT-3.5 family) [23].

4.2.1. Performance improvements of IBKS

Table 4 and Table 5 show the experimental results on *CommonsenseQA* and *OpenbookQA*, respectively. From these results, we could observe that **IBKS** could further improve these three KG-augmented models across different text encoders and different datasets.

In specific, for *CommonsenseQA* (Table 4), we utilize the in-house data split [16]. For the best text encoder Roberta-large, **IBKS** could improve 1.31%, 0.67%, and 1.06% on IHtest for RGCN, MHGRN, and QA-GNN, respectively. For the recent SOTA method QA-GNN [40], **IBKS** could bring 1.02%, 0.99%, 1.06% improvements for BERT-base, BERT-large, and Roberta-large, respectively.

As for *OpenbookQA* (Table 4), we conduct experiments on official data splits. For Roberta-large, **IBKS** outperforms the baseline methods 1.47%, 0.56%, 1.42% on the test dataset for RGCN, MHGRN, QA-GNN, respectively.



(a) Save Ratio on *CommonsenseQA* (b) Save Ratio on *OpenbookQA*

Fig. 4. Average edge save ratio of IHDev on *CommonsenseQA* and *OpenbookQA*. We present the results of three KG-augmented models across the three different text encoders: BERT-base, BERT-large, and Roberta-large.

Table 6
Performance comparison on *CommonsenseQA* in-house split. We choose QA-GNN and GREASELM and plug IBKS on these two KG-augmented models. Besides, we also show the performance of large language models. We select `text-davinci-002`, `text-davinci-003`, and `gpt-3.5-turbo` and show average 4-shot performances of 3 runs.

Methods	IHDev-Acc(.%)	IHTest-Acc(.%)
RoBERTa-large [19]	73.07 (±0.45)	68.69 (±0.56)
RoBERTa-large+RGCN [26]	72.69 (±0.19)	68.41 (±0.66)
RoBERTa-large+GconAttn [35]	72.61 (±0.39)	68.59 (±0.96)
RoBERTa-large+Kagnet [16]	73.47 (±0.22)	69.01 (±0.76)
RoBERTa-large+RN [25]	74.57 (±0.91)	69.08 (±0.21)
RoBERTa-large+MHGRN [6]	74.45 (±0.10)	71.11 (±0.81)
RoBERTa-large+QA-GNN [40]	76.54 (±0.21)	73.41 (±0.92)
RoBERTa-large+DESC-KCR [38]	78.21 (±0.23)	73.78 (±0.39)
RoBERTa-large+DGRN [44]	78.20	74.00
RoBERTa-large+GREASELM [43]	78.50 (±0.50)	74.20 (±0.40)
RoBERTa-large+JointLK [29]	77.78 (±0.25)	74.43 (±0.83)
RoBERTa-large+DHLK [36]	79.39 (±0.24)	74.68 (±0.26)
`text-davinci-002`	79.44 (±0.56)	74.78 (±0.58)
`text-davinci-003`	79.03 (±0.61)	74.70 (±0.62)
`gpt-3.5-turbo` (ChatGPT)	74.61 (±0.49)	72.63 (±0.52)
RoBERTa-large+QA-GNN+IBKS	78.47 (±1.14)	74.47 (±1.77)
RoBERTa-large+GREASELM+IBKS	**79.75** (±1.37)	**75.32** (±1.29)

Besides the performance improvement, we also present the edge save ratio of **IBKS** for these three KG-augmented models. As shown in Fig. 4, for these three KG-augmented models, **IBKS** could filter numerous useless knowledge. For different text encoders, we could observe that the stronger text encoder needs less extra knowledge for the same KG-augmented method. This is because the stronger text encoder like Roberta-large has more parameters and could save more knowledge in the pre-training phase. As for different KG-augmented methods, we do not find obvious conclusions and the results are just for reference, this is because the difference between these methods is too big, including completely different graph encoders and different fusion ways of the text representations and knowledge representations. Especially for MHGRN, which is based on a multi-hop message-passing mechanism, the average save ratio is obviously higher than the others. Considering our **IBKS** is based on edge selection, a multi-hop path would fail to pass messages when any edge in this path is dropped. In contrast, the similar multi-hop path would fail only when all the edges are filtered for RGCN and QA-GNN.

4.2.2. Comparison with SOTA methods

In this part, we select QA-GNN [40] and GREASELM [43] as the basic KG-augmented model and plug our IBKS on these two models. To compare with SOTA methods, we follow the settings in previous studies. In specific, we choose the text encoder as RoBERTa-large [19] and AristorRoBERTa [3] for *CommonsenseQA* and *OpenbookQA*, respectively. And we compare to the mainstream KG-augmented models in recent years, including RGCN [26], GconAttn [35], Kagnet [16], RN [25], MHGRN [6], QA-GNN [40], DESC-KCR [38], DGRN [44], GREASELM [43], JointLK [29], and DHLK [36]. And we also compare to the large language models. In specific, we compare to the GPT-3.5 series models, including `text-davinci-002`, `text-davinci-003`, and `gpt-3.5-turbo` (ChatGPT). For these three large language models, we perform 4-shot in-context learning and report the average performance of 3 runs due to the high API costs. Table 6 and Table 7 present the corresponding results, respectively.

Table 7
Performance comparison on *OpenbookQA* test set. We choose QA-GNN and GREASELM and plug IBKS on these two KG-augmented models. Besides, we also show the performance of large language models. We select `text-davinci-002`, `text-davinci-003`, and `gpt-3.5-turbo` and show average 4-shot performances of 3 runs.

Methods	Test-Acc(.%)
AristoRoBERTta [3]	78.40 (±1.64)
AristoRoBERTta+RGCN [26]	74.60 (±2.53)
AristoRoBERTta+GconAttn [35]	71.80 (±1.21)
AristoRoBERTta+RN [25]	75.35 (±1.39)
AristoRoBERTta+MHGRN [6]	80.60 (±0.10)
AristoRoBERTta+QA-GNN [40]	82.77 (±1.21)
AristoRoBERTta+DGRN [44]	84.10
AristoRoBERTta+GREASELM [43]	84.80 (±0.50)
AristoRoBERTta+JointLK [29]	84.92 (±1.07)
AristoRoBERTta+DHLK [36]	86.00 (±0.79)
`text-davinci-002`	80.43 (±0.54)
`text-davinci-003`	84.27 (±0.57)
`gpt-3.5-turbo` (ChatGPT)	79.67 (±0.42)
AristoRoBERTta+QA-GNN+IBKS	84.18 (±1.02)
AristoRoBERTta+GREASELM+IBKS	**86.12** (±1.33)

Table 8
Performance comparison between **IBKS** and random selection. And the random selection keeps the same save ratio with **IBKS**. We report the average performance of IHDev on *CommonsenseQA*. And ori refers to the original KG-augmented model without any knowledge selection.

	RGCN	MHGRN	QA-GNN
ori	72.69 (±0.19)	74.45 (±0.10)	76.54 (±0.21)
IBKS	74.01 (±1.27)	75.42 (±0.83)	78.47 (±1.14)
random selection	71.87 (±2.65)	73.59 (±2.15)	76.36 (±2.49)

For *CommonsenseQA*, when we plug IBKS on QA-GNN, we can get comparative performance with the recent SOTA methods and the large language models. To further illustrate the effectiveness of IBKS, we also add IBKS into a stronger KG-augmented model GREASELM, which follows a similar idea to QA-GNN. IBKS brings 1.25% and 1.12% on IHDev and IHTest on GREASELM, which achieve the best performance on *CommonsenQA*. For *OpenbookQA*, IBKS also improves 1.41% and 1.32% for QA-GNN and GREASELM, respectively. With the help of IBKS, GREASELM could achieve the best performance on *OpenbookQA*.

4.3. Effectiveness of information bottleneck for knowledge selection

In the previous experiments, corresponding results show that **IBKS** could improve existing KG-augmented models. In this section, we discuss the effectiveness of alternative knowledge selection methods. First, we apply random selection with the same save ratio with **IBKS** to illustrate the impact of the reduction of the knowledge scale. Then, we compare with the sparsity-based selection methods [20,1], which is an effective selection method and has been widely applied to rationale selection [10].

4.3.1. Impact of the reduction of knowledge scale

In this part, we explore whether the performance improvement in previous experiments simply comes from the reduction of the knowledge scale. To answer this question, We follow the save ratio in Fig. 4 and generate the corresponding random selection. In this way, we could set the scale of the introduced knowledge the same. And we select Roberta-large as the text encoder and conduct experiments on *CommonsenseQA*. Table 8 presents the results of the dev set. From these experimental results, we observe that the random selection brings big performance degradation. Concretely, compared to **IBKS**, the random selection brings 2.14%, 1.83%, and 2.11% performance drops on RGCN, MHGRN, and QA-GNN, respectively. Moreover, the random selection even leads to worse performance compared to the original KG-augmented models. These experimental results indicate that the performance improvement of **IBKS** is not only attributed to the reduction in the size of the subgraph and how selecting useful knowledge is the key to performance improvement.

4.3.2. Comparision with sparsity-based methods

In this section, we explore whether the IB-based knowledge selection frame is superior to other approaches. To address this problem, we compare **IBKS** with the sparsity-based knowledge selection method [20]. Let us go back to the formula (3), the second term $I(G;G_S|X)$ utilizes the mutual information to constrain the information from G to G_S. Sparsity-based methods usually adopt

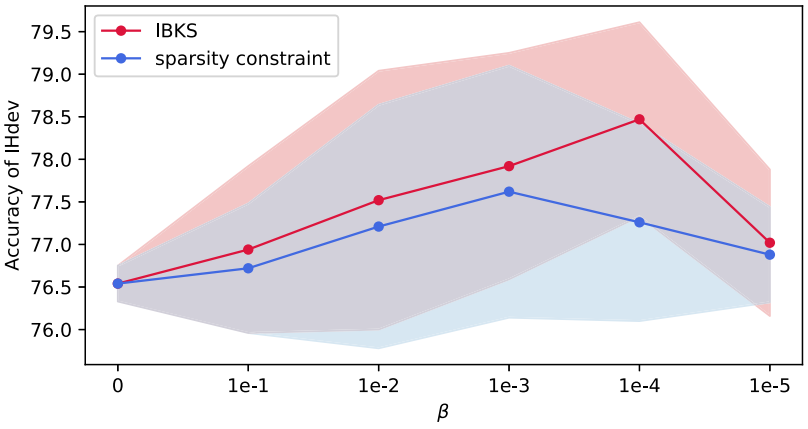


Fig. 5. Effect of different β. We show the average performance and the standard derivations on IHDev of *CommonsenseQA*. The red line indicates the information constraint, which is applied to our **IBKS**. The blue line presents the sparsity constraint, which is formulated in Formula (10). And $\beta = 0$ refers to the performance of the original QA-GNN and we take this as a baseline.

Table 9
Ablation study on text constraints, different prior distribution, and random selection using Roberta-large as the text encoder. We report the average performance of IHDev on *CommonsenseQA*.

	QA-GNN	RGCN
IBKS	78.47 (±1.14)	74.01 (±1.27)
w/o text constraint	78.43 (±1.17)	73.67 (±1.31)
w new prior	78.02 (±1.33)	73.54 (±1.36)

the sparsity loss to constrain information passing, which has been widely used in rationale selection [14,10]. Thus we utilize the sparsity constraint in our task for comparison, the specific formulation is:

$$\min -I(X, G_S; Y) + \beta \frac{\text{count\_edge}(G_S)}{\text{count\_edge}(G)} \tag{10}$$

where count_edge$(\cdot)$ refers to computing the number of edges for a given graph. Actually, count_edge(G_S) could be seen as L0-norm. For fair comparisons, we perform normalization to constrain the scale of the sparsity loss to be the same as the original conditional mutual information loss.

For comprehensive comparisons, we also conduct experiments using the sparsity constraints across different β. Fig. 5 presents the corresponding results. We conduct experiments on QA-GNN using Roberta-large. The red line presents the results of different β and the blue line shows the results of the sparsity-based method on different β.

Considering $\beta = 0$ refers to the performance of the original QA-GNN, we could find that both **IBKS** and the sparsity-based method could achieve better performance for different β, which validates the effectiveness of knowledge selection. Besides, we observe that the information constraint we used in **IBKS** outperforms the sparsity constraint for different β. From these results, we conclude that the mutual information constraints (**IBKS**) could perform better in knowledge selection compared to sparsity constraints. Thus the proposed IB method is more suitable for knowledge selection.

In summary, from the above experiments, we find that the performance improvements do not come from the reduction of the knowledge scale. Only reducing the size of the introduced knowledge even destroys the original KG-augmented models, which indicates that selecting which knowledge is the most important part. And the proposed methods utilize the mutual information constraints to help select knowledge, which is more effective than other methods like the sparsity-based method.

5. Discussions

5.1. Ablation study

In this section, we discuss the impact of the two important components in **IBKS**: the text constraint and the designed prior distribution. To illustrate their impacts, we remove the text constraint and design a new prior distribution. And we conduct the ablation experiments on RGCN and QA-GNN.

Impact of the Text Constraint. In Formula (1), we generate the binary embedding with the extra text constraint. Without this text constraint, it could be seen as GIB (Fig. 3(b)). We show the corresponding performance in Table 9. The text constraint leads to a 0.34% accuracy drop on RGCN. However, there is only a 0.04% performance drop on QA-GNN. This is because QA-GNN has

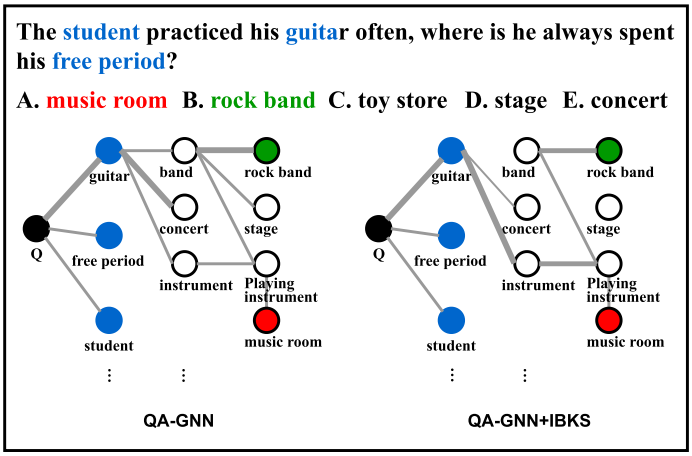


Fig. 6. The reasoning chain from the question node to the candidate answer node of QA-GNN on *CommonsenseQA*. Q refers to the question node and the blue nodes are entity nodes in the question. The red node refers to the golden answer and the green node refers to the original prediction answer. And the thicker edges indicate higher relevance between these two nodes, which could be seen as a reasoning process. After plugging **IBKS** into QA-GNN, the edge (`guitar`, `band`) and (`band`, `stage`) are seen as useless knowledge and filtered. As a result, the model would change the prediction from `rock band` to `music room`.

introduced a text node to the knowledge subgraph, which is equal to introducing the text constraint. Therefore, the text constraint only introduces performance drops on RGCN.

Impact of Different Prior Distribution. In the formula (6), we utilize the prior distribution $r(G_S|x)$ to estimate the inestimable distribution $p(G_S|x)$. We design other prior distribution which is associated with distance. The edge is sampled from Bern(p) and p is bigger when the edge is closer to the question node and answer node. We present the performance with this prior distribution in Table 9. This new prior distribution brings 0.45% and 0.47% drops in performance for QA-GNN and RGCN.

5.2. Example analysis

To further clearly demonstrate how **IBKS** improves the model, we analyze a specific example.

We take the SOTA method QA-GNN with Roberta-large as an example, which could provide a reasoning chain like Fig. 6. The thicker edges refer to higher relevance between the nodes. Therefore, according to the left figure, the reasoning path could be seen as Q → `guitar` → `band` → `rock band`. As a result, the original model predicts `rock band` as the answer, which is wrong.

After adding **IBKS**, the edge (`guitar`, `band`) and (`band`, `stage`) are dropped as useless knowledge and the new knowledge subgraph would be fed into QA-GNN. Correspondingly, as shown in the right figure, the model would generate a new reasoning chain without these two edges. i.e., Q → `guitar` → `instrument` → `playing instrument` → `music room`. Therefore, the model predicts `music room` as the answer, which is consistent with the golden answer.

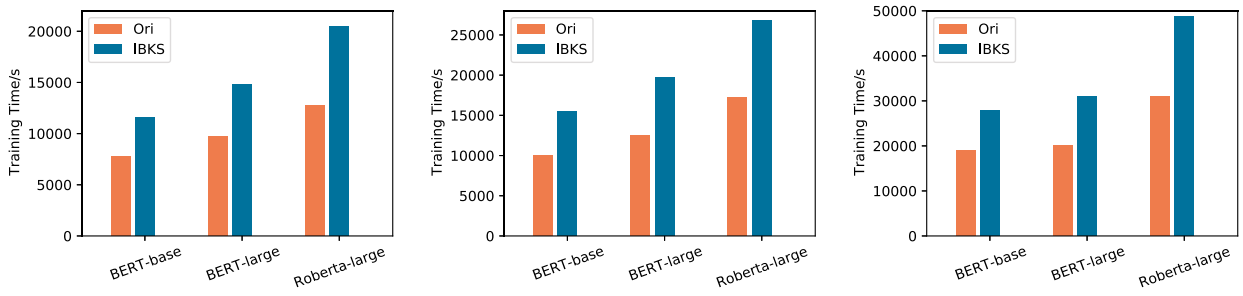
In summary, we think the performance improvement of **IBKS** comes from the filtration of useless knowledge, including irrelevant knowledge and the knowledge associated with the wrong options. Dropping irrelevant knowledge would reduce the scale of the subgraph. Filtering the knowledge associated with the wrong options would reduce interference information. Therefore, the new knowledge subgraph could provide more useful knowledge which is relevant to the ground truth answers.

5.3. Efficiency analysis

Considering our proposed method requires the training of the original KG-augmented models, thus we perform efficiency analysis in this part.

Firstly, we discuss the extra parameter costs of our proposed method. Considering **IBKS** is a plug-in module, the extra costs are only located on the knowledge selector in § 3.2. In our designed knowledge selector, the parameter costs are only the MLP in Formula (1), which transform the 100-dimension representations into 2-dimension representations in our experiments. As for the Gumbel-Softmax sampling, this sampling strategy is efficiently implemented in PyTorch and does not require extra parameter costs.

Then we analyze the training time costs of **IBKS**. Considering the extra knowledge selector module and the end-to-end training frame, **IBKS** is bound to incur additional training time consumption. We conduct experiments on all three KG-augmented models across the three text encoders. Then we compare the training time with and without **IBKS** and Fig. 7 presents the corresponding results of the training time. On average, **IBKS** increases about 50% extra training time for the three KG-augmented models across the three text encoders. Compared to the negligible extra parameter costs of the added knowledge selector module, the additional time costs are indeed not small. However, compared to designing and training a new method in recent studies [43], we think these costs are still acceptable. Besides, **IBKS** could plug into any KG-augmented models and bring enough performance improvements, which is more attractive than designing a new model.



(a) RGCN (b) MHGRN (c) QAGNN

Fig. 7. Training time comparison of whether to add **IBKS**. We present the training time of the three KG-augmented models across the three text encoders on CommensenseQA.

Table 10
Test accuracy on MedQA-USMLE.

Methods	Test-Acc(.%)
BERT-base [4]	34.3
BioBERT-base [13]	34.1
BERT-large [4]	35.0
BioBERT-large [13]	36.7
SapBERT [17]	37.2
SapBERT+QA-GNN [40]	38.0
SapBERT+GREASELM [43]	38.5
SapBERT+QA-GNN+IBKS	38.7
SapBERT+GREASELM+IBKS	**39.1**

5.4. Effectiveness on domain-oriented task

In previous parts, we demonstrate the effectiveness of IBKS in the general commonsense reasoning domain. To further show the domain generality, we explore whether IBKS could boost KG-augmented models on other domains. In specific, following previous studies [40,43], we test on MedQA-USMLE dataset [11], which is a 4-way multiple choice QA task that requires biomedical and clinical knowledge.

Following [40,43], we also utilize SpaBERT [17] as the text encoder. As shown in Table 10, IBKS brings 0.7% and 0.6% improvements for QA-GNN and GREASELM, respectively. According to these results, we can find that IBKS could also boost existing KG-augmented models in domain-oriented tasks.

6. Conclusion and future work

In this paper, we first illustrate the importance of knowledge selection for existing KG-augmented models. To select useful knowledge, we extend existing IB methods and propose **IBKS**, which is model-agnostic and could be plugged into any existing KG-augmented model. Extensive experimental results show the effectiveness of our method. **IBKS** has high applicability due to its model-agnostic nature, which can be used to enhance the performance of existing KG-augmented models. In addition, the selected knowledge can assist us in better understanding the reasoning process of the model, thereby providing guidance for model design.

There are some interesting future research directions to extend our work. First, although the proposed **IBKS** is mode-agnostic, these models are essentially still discriminative models. The recent study [18] applies the generative method to solve this task. It is desirable to further design knowledge selection methods in the generative setting. Secondly, IBKS is designed for knowledge selection in KG, which is graph structure. Therefore, IBKS could only work for KG-augmented models, which is only one of the mainstream QA methods. Expanding our approach to other types of question-answering methods is worth researching. Finally, large language models (LLMs) attract much attention currently. These LLMs contain numerous knowledge and it would be interesting to perform knowledge selection in LLMs for the downstream tasks.

CRediT authorship contribution statement

Zhao Yang: Conceptualization, Methodology, Software, Writing – original draft. **Yuanzhe Zhang:** Writing – review & editing. **Pengfei Cao:** Writing – review & editing. **Cao Liu:** Writing – review & editing. **Jiansong Chen:** Writing – review & editing. **Jun Zhao:** Funding acquisition, Writing – review & editing. **Kang Liu:** Funding acquisition, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

This work was supported by the National Key R&D Program of China (2022ZD0160503) and the National Natural Science Foundation of China (No. 62276264). This work is also supported by the Strategic Priority Research Program of Chinese Academy of Sciences (Grant No. XDA27020100), the Youth Innovation Promotion Association CAS, Yunnan Provincial Major Science and Technology Special Plan Projects (No. 202202AD080004) and Meituan.

References

[1] A.A. Alemi, I. Fischer, J.V. Dillon, K. Murphy, Deep variational information bottleneck, in: International Conference on Learning Representations (ICLR2016).
[2] P. Banerjee, K.K. Pal, A. Mitra, C. Baral, Careful selection of knowledge to solve open book question answering, in: Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, pp. 6120–6129.
[3] P. Clark, O. Etzioni, T. Khot, D. Khashabi, B. Mishra, K. Richardson, A. Sabharwal, C. Schoenick, O. Tafjord, N. Tandon, et al., From 'f' to 'a' on the ny regents science exams: an overview of the aristo project, AI Mag. 41 (2020) 39–53.
[4] J. Devlin, M.W. Chang, K. Lee, K. Toutanova, 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, Volume 1 (Long and Short Papers), pp. 4171–4186.
[5] Y. Du, J. Xu, H. Xiong, Q. Qiu, X. Zhen, C.G. Snoek, L. Shao, Learning to learn with variational information bottleneck for domain generalization, in: European Conference on Computer Vision, Springer, pp. 200–216.
[6] Y. Feng, X. Chen, B.Y. Lin, P. Wang, J. Yan, X. Ren, Scalable multi-hop relational reasoning for knowledge-aware question answering, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pp. 1295–1309.
[7] M.R. Garey, D.S. Johnson, The rectilinear Steiner tree problem is np-complete, SIAM J. Appl. Math. 32 (1977) 826–834.
[8] E. Jang, S. Gu, B. Poole, Categorical reparametrization with gumble-softmax in: International Conference on Learning Representations (ICLR 2017).
[9] J. Jiang, K. Zhou, J.R. Wen, X. Zhao, *great truths are always simple*: a rather simple knowledge encoder for enhancing the commonsense reasoning capacity of pre-trained models, in: Findings of the Association for Computational Linguistics: NAACL 2022, pp. 1730–1741.
[10] Z. Jiang, Y. Zhang, Z. Yang, J. Zhao, K. Liu, Alignment rationale for natural language inference, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pp. 5372–5387.
[11] D. Jin, E. Pan, N. Oufattole, W.H. Weng, H. Fang, P. Szolovits, What disease does this patient have? A large-scale open domain question answering dataset from medical exams, Appl. Sci. 11 (2021) 6421.
[12] D.P. Kingma, J. Ba Adam, A method for stochastic optimization, arXiv preprint arXiv:1412.6980, 2014.
[13] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C.H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2020) 1234–1240.
[14] T. Lei, R. Barzilay, T. Jaakkola, Rationalizing neural predictions, in: Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, pp. 107–117.
[15] H.J. Levesque, E. Davis, L. Morgenstern, The winograd schema challenge, in: Proceedings of the Thirteenth International Conference on Principles of Knowledge Representation and Reasoning, pp. 552–561.
[16] B.Y. Lin, X. Chen, J. Chen, X. Ren 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), pp. 2829–2839.
[17] F. Liu, E. Shareghi, Z. Meng, M. Basaldella, N. Collier, Self-alignment pretraining for biomedical entity representations, in: Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 4228–4238.
[18] J. Liu, A. Liu, X. Lu, S. Welleck, P. West, R. Le Bras, Y. Choi, H. Hajishirzi, Generated knowledge prompting for commonsense reasoning, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 3154–3169.
[19] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: a robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692, 2019.
[20] S. Miao, M. Liu, P. Li, Interpretable and generalizable graph learning via stochastic attention mechanism, in: International Conference on Machine Learning, PMLR, pp. 15524–15543.
[21] T. Mihaylov, A. Frank, Knowledgeable reader: Enhancing cloze-style reading comprehension with external commonsense knowledge, in: Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 821–832.
[22] M. Niepert, M. Ahmed, K. Kutzkov, Learning convolutional neural networks for graphs, in: International Conference on Machine Learning, PMLR, pp. 2014–2023.
[23] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al., Training language models to follow instructions with human feedback, Adv. Neural Inf. Process. Syst. 35 (2022) 27730–27744.
[24] M. Raman, A. Chan, S. Agarwal, P. Wang, H. Wang, S. Kim, R. Rossi, H. Zhao, N. Lipka, X. Ren, Learning to deceive knowledge graph augmented models via targeted perturbation, in: International Conference on Learning Representations (ICLR2020).
[25] A. Santoro, D. Raposo, D.G. Barrett, M. Malinowski, R. Pascanu, P. Battaglia, T. Lillicrap, A simple neural network module for relational reasoning, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, pp. 4974–4983.
[26] M. Schlichtkrull, T.N. Kipf, P. Bloem, R.v.d. Berg, I. Titov, M. Welling, Modeling relational data with graph convolutional networks, in: European Semantic Web Conference, Springer, pp. 593–607.
[27] M.S. Schlichtkrull, N. De Cao, I. Titov, Interpreting graph neural networks for nlp with differentiable edge masking, in: International Conference on Learning Representations (ICLR2020).
[28] R. Speer, J. Chin, C. Havasi, Conceptnet 5.5: an open multilingual graph of general knowledge, in: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, AAAI2017, pp. 4444–4451.

[29] Y. Sun, Q. Shi, L. Qi, Y. Zhang 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, pp. 5049–5060.
[30] A. Talmor, J. Herzig, N. Lourie, J. Berant, 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, Volume 1 (Long and Short Papers), pp. 4149–4158.
[31] N. Tishby, F.C. Pereira, W. Bialek, The information bottleneck method, arXiv preprint arXiv:physics/0004057, 2000.
[32] N. Tishby, N. Zaslavsky, Deep learning and the information bottleneck principle, in: 2015 IEEE Information Theory Workshop (ITW), IEEE, pp. 1–5.
[33] P. Veličković, G. Cucurull, A. Casanova, A. Romero, P. Liò, Y. Bengio, Graph attention networks, in: International Conference on Learning Representations (ICLR 2018).
[34] B. Wang, S. Wang, Y. Cheng, Z. Gan, R. Jia, B. Li, J. Liu, Infobert: improving robustness of language models from an information theoretic perspective, in: International Conference on Learning Representations (ICLR2020).
[35] X. Wang, P. Kapanipathi, R. Musa, M. Yu, K. Talamadupula, I. Abdelaziz, M. Chang, A. Fokoue, B. Makni, N. Mattei, et al., Improving natural language inference using external knowledge in the science questions domain, in: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 7208–7215.
[36] Y. Wang, H. Zhang, J. Liang, R. Li, Dynamic heterogeneous-graph reasoning with language models and knowledge representation learning for commonsense question answering, in: Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pp. 14048–14063.
[37] T. Wu, H. Ren, P. Li, J. Leskovec, Graph information bottleneck, Adv. Neural Inf. Process. Syst. 33 (2020) 20437–20448.
[38] Y. Xu, C. Zhu, R. Xu, Y. Liu, M. Zeng, X. Huang, Fusing context into knowledge graph for commonsense question answering, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pp. 1201–1207.
[39] J. Yang, G. Xiao, Y. Shen, W. Jiang, X. Hu, Y. Zhang, J. Peng, A survey of knowledge enhanced pre-trained models, arXiv preprint arXiv:2110.00269, 2021.
[40] M. Yasunaga, H. Ren, A. Bosselut, P. Liang, J. Leskovec 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, pp. 535–546.
[41] R. Ying, D. Bourgeois, J. You, M. Zitnik, J. Leskovec, Gnnexplainer: generating explanations for graph neural networks, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, pp. 9244–9255.
[42] J. Yu, T. Xu, Y. Rong, Y. Bian, J. Huang, R. He, Graph information bottleneck for subgraph recognition, in: International Conference on Learning Representations (ICLR2020).
[43] X. Zhang, A. Bosselut, M. Yasunaga, H. Ren, P. Liang, C.D. Manning, J. Leskovec, Greaselm: Graph reasoning enhanced language models, in: International conference on learning representations (ICLR2022).
[44] C. Zheng, P. Kordjamshidi, Dynamic relevance graph network for knowledge-aware question answering, in: Proceedings of the 29th International Conference on Computational Linguistics, pp. 1357–1366.
[45] C. Zhu, Y. Xu, X. Ren, B.Y. Lin, M. Jiang, W. Yu, Knowledge-augmented methods for natural language processing, in: Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts, pp. 12–20.