Fusing Context Into Knowledge Graph for Commonsense Question Answering

Yichong Xu*, Chenguang Zhu*, Ruochen Xu, Yang Liu, Michael Zeng, Xuedong Huang
Microsoft Cognitive Services Research Group
{yicxu,chezhu,ruox,yaliu10,nzeng,xdh}@microsoft.com

Abstract

Commonsense question answering (QA) requires a model to grasp commonsense and factual knowledge to answer questions about world events. Many prior methods couple language modeling with knowledge graphs (KG). However, although a KG contains rich structural information, it lacks the context to provide a more precise understanding of the concepts. This creates a gap when fusing knowledge graphs into language modeling, especially when there is insufficient labeled data. Thus, we propose to employ external entity descriptions to provide contextual information for knowledge understanding. We retrieve descriptions of related concepts from Wiktionary and feed them as additional input to pre-trained language models. The resulting model achieves state-of-the-art result in the CommonsenseQA dataset and the best result among non-generative models in OpenBookQA.

1 Introduction

One critical aspect of human intelligence is the ability to reason over everyday matters based on observation and knowledge. This capability is usually shared by most people as a foundation for communication and interaction with the world. Therefore, commonsense reasoning has emerged as an important task in natural language understanding, with various datasets and models proposed in this area (Ma et al., 2019; Talmor et al., 2018; Wang et al., 2020; Lv et al., 2020).

While massive pre-trained models (Devlin et al., 2018; Liu et al., 2019) are effective in language understanding, they lack modules to explicitly handle knowledge and commonsense. Also, structured data like knowledge graph is much more efficient in representing commonsense compared with unstructured text. Therefore, there have been multiple methods coupling language models with various forms of knowledge graphs (KG) for commonsense reasoning, including knowledge bases (Sap et al., 2019; Yu et al., 2020b), relational paths (Lin et al., 2019), graph relation network (Feng et al., 2020) and heterogeneous graph (Lv et al., 2020). These methods combine the merits of language modeling and structural knowledge information and improve the performance of commonsense reasoning and question answering.

However, there is still a non-negligible gap between the performance of these models and humans. One reason is that, although a KG can encode topological information between the concepts, it lacks rich context information. For instance, for a graph node for the entity "Mona Lisa", the graph depicts its relations to multiple other entities. But given this neighborhood information, it is still hard to infer that it is a painting. On the other hand, we can retrieve the precise definition of "Mona Lisa" from external sources, e.g. the definition of Mona Lisa in Wiktionary is "*A painting by Leonardo da Vinci, widely considered as the most famous painting in history*". To represent structured data that can be seamlessly integrated into language models, we need to provide a panoramic view of each concept in the knowledge graph, including its neighboring concepts, relations to them, and a definitive description of it.

Thus, we propose the DEKCOR model, i.e. DEscriptive Knowledge for COmmonsense question answeRing, to tackle multiple choice commonsense questions. Given a question and a choice, we first extract the contained concepts. Then, we extract the edge between the question concept and the choice concept in ConceptNet (Speer et al., 2017). If such an edge does not exist, we compute a relevance score for each knowledge triple (subject, relation, object) containing the choice concept, and select the one with the highest score. Next, we

[*] Equal contribution

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1201–1207
August 1–6, 2021. ©2021 Association for Computational Linguistics

retrieve the definition of these concepts from Wiktionary via multiple criteria of text matching. Finally, we feed the question, choice, selected triple and definitions into the language model ALBERT (Lan et al., 2019) to produce a score indicating how likely this choice is the correct answer.

We evaluate our model on CommonsenseQA (Talmor et al., 2018) and OpenBookQA (Mihaylov et al., 2018). On CommonsenseQA, it outperforms the previous state-of-the-art result by 1.2% (single model) and 3.8% (ensemble model) on the test set. On OpenBookQA, our model outperforms all baselines other than two large-scale models based on T5 (Raffel et al., 2019). We further conduct ablation studies to demonstrate the effectiveness of fusing context into the knowledge graph.

2 Related work

Several different approaches have been investigated for leveraging external knowledge sources to answer commonsense questions. Min et al. (2019) addresses open-domain QA by retrieving from a passage graph, where vertices are passages and edges represent relationships derived from external knowledge bases and co-occurrence. Sap et al. (2019) introduces the ATOMIC graph with 877k textual descriptions of inferential knowledge (e.g. if-then relation) to answer causal questions. Lin et al. (2019) projects questions and choices to the knowledge-based symbolic space as a schema graph. It then utilizes path-based LSTM to give scores. Feng et al. (2020) adopts the multi-hop graph relation network (MHGRN) to perform reasoning unifying path-based methods and graph neural networks. Lv et al. (2020) proposes to extract evidence from both structured knowledge base such as ConceptNet and Wikipedia text and conduct graph-based representation and inference for commonsense reasoning. Wang et al. (2020) employs GPT-2 to generate paths between concepts in a knowledge graph, which can dynamically provide multi-hop relations between any pair of concepts.

Several studies have utilized knowledge descriptions for different tasks. Yu et al. (2020a) uses description text from Wikipedia for knowledge-text co-pretraining. Xie et al. (2016) encodes the semantics of entity descriptions in knowledge graphs to improve the performance on knowledge graph completion and entity classification. Chen et al. (2018) co-trains the knowledge graph embeddings and entity description representation for cross-lingual entity alignment. Concurrent with our work, Chen et al. (2020) also insert knowledge descriptions into commonsense question answering. Compared with our work, the proposed method in Chen et al. (2020) is much more complex, e.g. involving training additional rankers on retrieved text, while our result outperforms Chen et al. on CommonsenseQA.

3 Method

3.1 Knowledge Retrieval

Problem formulation. In this paper, we focus on the following QA task: given a commonsense question q, select the correct answer from several choices $c_1, ..., c_n$. In most cases, the question does not contain any mentions of the answer. Therefore, external knowledge sources can be used to provide additional information. We adopt ConceptNet (Speer et al., 2017) as our knowledge graph $G = (V, E)$, which contains over 8 million entities as nodes and over 21 million relations as edges. In the following, we use triple to refer to two neighboring nodes and the edge connecting them, i.e. $(u \in V, p = (u, v) \in E, v \in V)$, with u being the subject, p the relation, and v the object.

Suppose the question mentions an entity $e_q \in V$ and the choice contains an entity $e_c \in V$[1]. We then employ the KCR method (Lin, 2020) to select relation triples. If there is a direct edge r from e_q to e_c in G, we choose this triple (e_q, r, e_c). Otherwise, we retrieve all the N triples containing e_c. Each triple j is assigned a score s_j which is the product of its triple weight w_j provided by ConceptNet and relation type weight t_{r_j}:

$$s_j = w_j \cdot t_{r_j} = w_j \cdot \frac{N}{N_{r_j}} \tag{1}$$

Here, r_j is the relation type of the triple j, and N_{r_j} is the number of triples among the retrieved triples that have the relation type r_j. In other words, this process favors rarer relation types. Finally, the triple with the highest weight is chosen.

3.2 Contextual information

The retrieved entities and relations from the knowledge graph are described by their surface form. Without additional context, it is hard for the language model to understand its exact meaning, especially for proper nouns.

[1] CommonsenseQA provides the question/choice entity. For OpenBookQA, we choose from the extracted entities that are most frequent in retrieved facts. See Appendix for details.

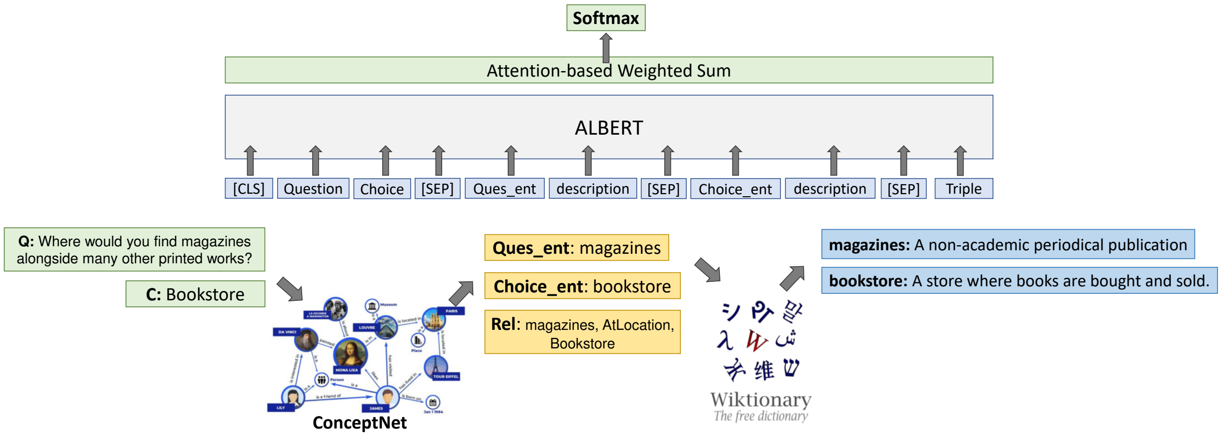
Softmax
Attention-based Weighted Sum
ALBERT
[CLS]
Question
Choice
[SEP]
Ques_ent
description
[SEP]
Choice_ent
description
[SEP]
Triple
Q: Where would you find magazines alongside many other printed works?
C: Bookstore
ConceptNet
Ques_ent: magazines
Choice_ent: bookstore
Rel: magazines, AtLocation, Bookstore
Wiktionary
The free dictionary
magazines: A non-academic periodical publication
bookstore: A store where books are bought and sold.

Figure 1: In our model, the input to ALBERT includes the question, choice, entity names, description text and triple. An attention-based weighted sum and a softmax layer process the output from ALBERT to produce the prediction.

Therefore, we leverage large-scale online dictionaries to provide definitions as context. We use a dump of Wiktionary[2] which includes definitions of 999,614 concepts. For every concept, we choose its first definition entry in Wiktionary as the description. For every question/choice concept, we find its closest match in Wiktionary by using the following forms in order: i) original form; ii) lemma form by Spacy (Honnibal and Montani, 2017); iii) base word (last word). For example, the concept "taking notes" does not appear in its original form in Wiktionary, but its lemma form "take notes" is in Wiktionary and we get its description text: "*To make a record of what one hears or observes for future reference*". In this way, we find descriptions of all entities in our experiments. The descriptions of the question and choice concept are denoted by d_q and d_c, respectively.

Finally, we feed the question, choice, descriptions and triple (from Section 3.1) into the ALBERT model (Lan et al., 2019) in the following format: [CLS] q c [SEP] e_q: d_q [SEP] e_c: d_c [SEP] triple.

3.3 Reasoning

On top of ALBERT, we leverage an attention-based weighted sum and a softmax layer to generate the relevance score for the question-choice pair. In detail, suppose the output representations of ALBERT is $(\boldsymbol{x}_0, ..., \boldsymbol{x}_m)$, where $\boldsymbol{x}_i \in R^d$. We compute a weighted sum of these embeddings based on attention:

$$q_i = \boldsymbol{u}^T \boldsymbol{x}_i \tag{2}$$

$$\alpha_i = \text{softmax}(q_i) \tag{3}$$

$$\boldsymbol{v} = \sum_{i=0}^{m} \alpha_i \boldsymbol{x}_i, \tag{4}$$

where $\boldsymbol{u}$ is a parameter vector. The relevance score between the question and the choice is then $s = \text{softmax}(\boldsymbol{v}^T \boldsymbol{b})$, where $\boldsymbol{b} \in R^d$ is a parameter vector and the softmax is computed over all choices for the cross-entropy loss function.

Table 1: Statistics of CommonsenseQA (CSQA) and OpenBookQA (OBQA).

Dataset	Train	Dev	Test	Choices
CSQA	9,741	1,221	1,140	5
OBQA	4,957	500	500	4

The architecture of our model DEKCOR and the construction of input is shown in Fig. 1.

4 Experiments

4.1 Datasets and baselines

We evaluate our model on two benchmark datasets of multiple-choice questions for commonsense question answering: CommonsenseQA (Talmor et al., 2018) and OpenBookQA (Mihaylov et al., 2018). CommonsenseQA creates questions from ConceptNet entities and relations; OpenBookQA probes elementary science knowledge from a book of 1,326 facts. The statistics of the datasets is provided in Table 1. For OpenBookQA, we follow prior approaches (Wang et al., 2020) to append top

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

Table 2: Accuracy on the test set of CommonsenseQA.

Methods	**Single**	**Ensemble**
BERT+OMCS	62.5	-
RoBERTa	72.1	72.5
RoBERTa+HyKAS	73.2	-
XLNet+DREAM	-	73.3
RoBERTa+KE	73.3	-
RoBERTa+KEDGN	-	74.4
XLNet+GraphReason	75.3	-
ALBERT	-	76.5
RoBERTa+MHGRN	75.4	76.5
ALBERT+PG-Full	75.6	78.2
T5	78.1	-
ALBERT+KRD	78.4	-
UnifiedQA	79.1	-
ALBERT+KCR	79.5	-
DEKCOR (ours)	**80.7**	**83.3**

Table 3: Accuracy on the test set of OpenBookQA.

Methods	**Accuracy**
BERT + Careful Selection	72.0
AristoRoBERTa	77.8
ALBERT + KB	81.0
ALBERT + PG-Full	81.8
TTTTT (T5-3B)	83.2
UnifiedQA (T5-11B)	**87.2**
DEKCOR (ours)	82.4

5 retrieved facts provided by Aristo team (Clark et al., 2019) to the input. We also pre-train our OpenBookQA model on CommonsenseQA's training set as we find it helps to boost the performance.

We compare our models with state-of-the-art baselines, which all employ pre-trained models including RoBERTa (Liu et al., 2019), XLNet (Yang et al., 2019), ALBERT (Lan et al., 2019) and T5 (Raffel et al., 2019) and some adopt additional modules to process knowledge information. A detailed description of the baselines is in the Appendix.

4.2 Results

CommonsenseQA. Table 2 shows the accuracy on the test set of CommonsenseQA. For a fair comparison, we categorize the results into single models and ensemble models. Our ensemble model consists of 7 single models with different initialization random seeds, and its output is the majority of choices selected by these single models. More implementation details are shown in the Appendix.

Table 4: Ablation results on the dev sets of CommonsenseQA and OpenBookQA.

Methods	**CSQA**	**OBQA**
DEKCOR	84.7	82.2
Triple Only	82.0	80.0
Description Only	80.3	81.8
No Context	78.9	80.0

Our proposed DEKCOR outperforms the previous state-of-the-art result by 1.2% (single model) and 3.8% (ensemble model). This demonstrates the effectiveness of the usage of knowledge description to provide context.

Furthermore, we notice two trends based on the results. First, the underlying pre-trained language model is important in commonsense QA quality. In general, we observe this order of accuracy among these language models: BERT<RoBERTa<XLNet<ALBERT<T5. Second, the additional knowledge module is critical to provide external information for reasoning. For example, RoBERTa+KEDGN outperforms the vanilla RoBERTa by 1.9%, and our model outperforms the vanilla ALBERT model by 6.8% in accuracy.

OpenBookQA. Table 3 shows the test set accuracy on OpenBookQA. All results are from single models. Note that the two best-performing models, i.e. UnifiedQA (Khashabi et al., 2020) and TTTTT (Raffel et al., 2019), are based on the T5 generation model, with 11B and 3B parameters respectively. Thus, they are computationally very expensive. Except these T5-based systems, DEKCOR achieves the best accuracy among all baselines.

Ablation study. Table 4 shows that the usage of concept descriptions from Wiktionary and triple from ConceptNet can help improve the accuracy of DEKCOR on the dev set of CommonsenseQA by 2.7% and 4.4% respectively. We observe similar results on OpenBookQA. This demonstrates that additional context information can help with fusing knowledge graph into language modeling for commonsense question answering.

Case Study. Table 5 shows two examples from CommonsenseQA and OBQA respectively. In the first example, without additional description the model would not know relevant information about bats, like they are insectivorous, leading to the wrong answer "eating bugs". With the description, the model knows that bats eat bugs, so it chooses "laying eggs" as the answer. Similarly, for the sec-

CommonsenseQA Question: Bats have many quirks, with the exception of ___ . **Question entity description:** bat: Any of the flying mammals of the order Chiroptera, usually small and nocturnal, insectivorous or frugivorous.
Model w/o description chooses: eating bugs **Model w/ description chooses:** laying eggs
OBQA Question: Alligators ___ . **Question entity description:** alligator: Either of two species of large amphibious reptile, ..., which have sharp teeth and very strong jaws...
Model w/o description chooses: eat gar **Model w/ description chooses:** are warm-blooded

Table 5: Examples from CommonsenseQA and OBQA dataset showing the effectiveness of entity descriptions.

ond question, the "sharp teeth and very strong jaws" in the description hint that alligators are likely carnivorous, and reptiles are likely cold-blooded. The entity description leads to the correct answer of "eat gar".

5 Conclusions

In this paper, we propose to fuse context information into knowledge graphs for commonsense question answering. As a knowledge graph often lacks descriptions for the contained entities and relations, we leverage Wiktionary to provide definitive text for each entity as additional input to the pre-trained language model ALBERT. The resulting DEKCOR model achieves state-of-the-art results on the benchmark datasets CommonsenseQA and OpenBookQA. Ablation studies demonstrate the effectiveness of the proposed usage of knowledge description and knowledge triple information in commonsense question answering.

Acknowledgements

We thank the anonymous reviewers for their valuable comments.

References

Pratyay Banerjee, Kuntal Kumar Pal, A. Mitra, and Chitta Baral. 2019. Careful selection of knowledge to solve open book question answering. In *ACL*.

Muhao Chen, Yingtao Tian, Kai-Wei Chang, Steven Skiena, and Carlo Zaniolo. 2018. Co-training embeddings of knowledge graphs and entity descriptions for cross-lingual entity alignment. *arXiv preprint arXiv:1806.06478*.

Qianglong Chen, Feng Ji, Haiqing Chen, and Yin Zhang. 2020. Improving commonsense question answering by graph-based iterative retrieval over multiple knowledge sources. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2583–2594, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Peter Clark, Oren Etzioni, Daniel Khashabi, Tushar Khot, Bhavana Dalvi Mishra, Kyle Richardson, Ashish Sabharwal, Carissa Schoenick, Oyvind Tafjord, Niket Tandon, et al. 2019. From'f'to'a'on the ny regents science exams: An overview of the aristo project. *arXiv preprint arXiv:1909.01958*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yanlin Feng, Xinyue Chen, Bill Yuchen Lin, Peifeng Wang, Jun Yan, and Xiang Ren. 2020. Scalable multi-hop relational reasoning for knowledge-aware question answering. *arXiv preprint arXiv:2005.00646*.

Matthew Honnibal and Ines Montani. 2017. spacy 2: Natural language understanding with bloom embeddings, convolutional neural networks and incremental parsing. *To appear*, 7(1).

Daniel Khashabi, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system. *arXiv preprint arXiv:2005.00700*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. Kagnet: Knowledge-aware graph networks for commonsense reasoning. *arXiv preprint arXiv:1909.02151*.

Jession Lin. 2020. Knowledge chosen by relations. `https://github.com/jessionlin/csqa/blob/master/Model_details.md`.

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

Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2020. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering.

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

Todor Mihaylov, Peter Clark, Tushar Khot, and A. Sabharwal. 2018. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *EMNLP*.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A Smith, and Yejin Choi. 2019. Atomic: An atlas of machine commonsense for if-then reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 3027–3035.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2018. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*.

Peifeng Wang, Nanyun Peng, Pedro A. Szekely, and X. Ren. 2020. Connecting the dots: A knowledgeable path generator for commonsense question answering. In *EMNLP*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, et al. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, pages arXiv–1910.

Ruobing Xie, Zhiyuan Liu, Jia Jia, Huanbo Luan, and Maosong Sun. 2016. Representation learning of knowledge graphs with entity descriptions.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. pages 5753–5763.

Donghan Yu, Chenguang Zhu, Yiming Yang, and Michael Zeng. 2020a. Jaket: Joint pre-training of knowledge graph and language understanding. *arXiv preprint arXiv:2010.00796*.

Wenhao Yu, Chenguang Zhu, Zaitang Li, Zhiting Hu, Qingyun Wang, Heng Ji, and Meng Jiang. 2020b. A survey of knowledge-enhanced text generation. *arXiv preprint arXiv:2010.04389*.

A Implementation Details

Identification of e_q and e_c. CommonsenseQA specifies the question entity in each question and each answer choice is also an entity in ConceptNet. We use them as e_q and e_c. For OpenBookQA, we identify all ConceptNet entities in the question and answer text and count their number of occurrences in the retrieved text. For a triple (e_q, r, e_c), we define its weight as $n_{e_q} + n_{e_c}$, where n_e is the number of occurrences in retrieved text. The edge with the largest weight is picked. If no edge is found between question and answer entities, we use the answer entity with the most occurrences to find triples. For Wiktionary descriptions, we find descriptions for e_q and e_c with the most occurrences as well.

Using ConceptNet. Since ConceptNet contains a lot of weak relations, we only use the following relations for our triples: CausesDesire, HasProperty, CapableOf, PartOf, AtLocation, Desires, HasPrerequisite, HasSubevent, Antonym, Causes.

Optimization. We use the AdamW (Loshchilov and Hutter, 2017) optimizer with a learning rate of 2e-5. The batch size is 8. We limit the maximum length of the input sequence to 192 tokens. The model is trained for 10 epochs. We use the Huggingface (Wolf et al., 2019) implementation for the ALBERT model.

B Baseline Methods

GraphReason (Lv et al., 2020) retrieves knowledge from both structured knowledge base and plain text.

PG-FULL (Wang et al., 2020) fine-tunes GPT-2 on ConceptNet to generate knowledgeable paths between knowledge graph concepts.

UnifiedQA (Khashabi et al., 2020) pre-trains T5 on a variety of QA datasets for general QA tasks.

MHGRN (Feng et al., 2020) adopts the multi-hop graph relation network to perform reasoning.

HyKAS (Ma et al., 2019) employs an option comparison network to consume ConceptNet triples.

ALBERT+KRD retrieves commonsense knowledge from Open Mind Common Sense and then uses a self-attention module to compute a weighted sum of these triple representations.

BERT + Selection (Banerjee et al., 2019) improves the result on OpenBookQA via abductive information retrieval , information gain based re-ranking, passage selection and weighted scoring.

ALBERT+KB also improves retrieval results on OpenBookQA by token-based and embedding-based retrieval. TTTTT (Raffel et al., 2019) fine-tunes the T5 language generation model on OpenBookQA.