Incorporating Commonsense Knowledge Graph in Pretrained Models for Social Commonsense Tasks

Ting-Yun Chang[1], Yang Liu[2], Karthik Gopalakrishnan[2], Behnam Hedayatnia[2],
Pei Zhou[3], Dilek Hakkani-Tür[2] *

[1] Academia Sinica, Taiwan;
[2] Alexa AI, Amazon, USA;
[3] USC, USA

r06922168@ntu.edu.tw, {yangliud,karthgop,behnam,hakkanit}@amazon.com

Abstract

Pretrained language models have excelled at many NLP tasks recently; however, their social intelligence is still unsatisfactory. To enable this, machines need to have a more general understanding of our complicated world and develop the ability to perform commonsense reasoning besides fitting the specific downstream tasks. External commonsense knowledge graphs (KGs), such as ConceptNet, provide rich information about words and their relationships. Thus, towards general commonsense learning, we propose two approaches to *implicitly* and *explicitly* infuse such KGs into pretrained language models. We demonstrate our proposed methods perform well on SocialIQA, a social commonsense reasoning task, in both limited and full training data regimes.

1 Introduction

Empowering machines with commonsense has become a hot topic recently. Past research efforts for this problem include the construction of various data sets and models. Several commonsense data sets have been commonly used in past work to develop machines' commonsense capability (Talmor et al., 2019; Huang et al., 2019; Zellers et al., 2019; Sap et al., 2019b; Sakaguchi et al., 2019; Gordon et al., 2012; Rajani et al., 2019). In particular, SocialIQA (Sap et al., 2019b) is a multiple-choice QA data set for probing machine's emotional and social intelligence in a variety of everyday situations, which is the data set used in this study. To improve the modeling approaches for the SocialIQA and other commonsense tasks, Shwartz et al. (2020) and Bosselut and Choi (2019) focused on zero-shot setting using pretrained language models. Khashabi et al. (2020) reformulated the multi-choice setup used in most data sets as a generation task and

achieved impressive performance by fine-tuning T5 (Raffel et al., 2019). Recently there is an increasing effort to utilize external knowledge bases to incorporate commonsense information underlying the text (Shwartz et al., 2020; Mitra et al., 2019; Ji et al., 2020a,b).

While most prior work on SocialIQA utilized large pretrained language models (Devlin et al., 2019; Liu et al., 2019; Radford et al., 2018, 2019; Raffel et al., 2019), we argue that such a challenging task requires commonsense reasoning of social events, and simply fine-tuning the model to fit the task is insufficient. We believe it would be beneficial if the model can learn from knowledge-rich resources such as ConceptNet (Liu and Singh, 2004), and thus have a broader and deeper understanding of the information not present in the provided context and answer candidates.

In this paper, we propose two approaches tailored to large pretrained language models to utilize existing knowledge graph (KGs) for downstream commonsense tasks. The first approach leverages the KGs *implicitly* by pretraining on the relevant tuples to the SocialIQA task, while the second one maintains a dynamic knowledge base during fine-tuning, utilizing KGs *explicitly* via an attention mechanism. Our experiments demonstrate the effectiveness of both approaches on SocialIQA under limited and full training data regimes, and the critical role of relevant knowledge.

2 Problem Formulation and Baseline

In SocialIQA, given a context C of an event and a corresponding question Q, the goal is to select the correct choice from the answer set $A = (A_1, A_2, A_3)$. An example is shown in Figure 1, with the provided context, question, and three answer candidates.

A typical approach (as used in Sap et al.

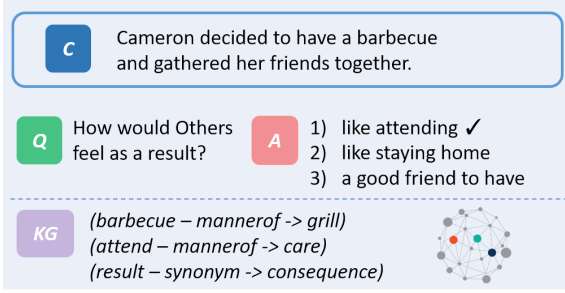‡ Work was done while Ting-Yun Chang and Pei Zhou were interns at Amazon.

Figure 1: An instance in SocialIQA and our retrieved tuples from ConceptNet.

(2019b)) for solving this kind of multiple-choice problems with a pretrained Transformer-based language model is by concatenating C, Q, and A_i with a separator token, and then letting the model output a score via a multi-layer perceptron (MLP) built on top of the final hidden representation of the classifier token $[CLS]$. Finally, scores for each data point are normalized across all (C, Q, A_i) instances with softmax, and cross-entropy loss is applied for model training.

Since RoBERTa (Liu et al., 2019) has shown competitive performance on the SocialIQA task, we use it as a baseline model in this study. Furthermore, in addition to multiple-choice classification, we perform masked language modeling (MLM) (Devlin et al., 2019), masking 15% tokens in the concatenation of C, Q, and the correct answer A_{y^*}, when fine-tuning on the SocialIQA task.

3 Incorporating Commonsense Knowledge Graph

In this section, we introduce two methods to incorporate a given KG into our pretrained model. We experiment with both ATOMIC (Sap et al., 2019a) and ConceptNet (Liu and Singh, 2004) as our KGs.

- **ATOMIC** focuses on inferential knowledge of everyday situations. Each node in ATOMIC is a social event, containing 9 *if-then* relation types. Note that though SocialIQA is derived from ATOMIC, it has been rewritten by crowd workers (Sap et al., 2019b).

- **ConceptNet** represents general words and phrases that people use and the commonsense relationships between them, such as *IsA, At-Location, Desires, Synonym*.

3.1 Querying Knowledge Graph

For both methods, we first extract keywords in the input data to query ConceptNet using only the lemmatized *noun, verb, adjective* words in (C, Q, A) as queries, with stop words excluded. First, we find the node corresponding to each query in the KG, and retrieve all the connected tuples *(query, relation with weight, tail)* within one hop. We then sort all the retrieved tuples by their $relation\ weight \times query's\ idf$, and keep the top-$k$ tuples for each data point.[1] In Figure 1, the bottom shows examples of retrieved tuples from ConceptNet for an instance in the SocialIQA data. Note that since ATOMIC highly overlaps with SocialIQA, we do not extract keywords to query the KG but pretrain the model on the entire KG.

3.2 Pretraining Language Models on Retrieved Concepts

In the first approach, we leverage the KG via infusing it into the pretraining step. Using the SocialIQA data as queries, we first retrieve tuples from the KGs as described above, and then convert them to textual forms. To enable this conversion, we hand-crafted templates for different relations. For example, a tuple in ConceptNet *(barbecue, hascontext, cooking)* would be converted into "*barbecue is a word used in the context of cooking.*" When using ATOMIC, because there are some blanks and unknown names such as "*PersonX meets _ for lunch*", we replace *PersonX* and *PersonY* with two different common *last names* to avoid gender bias, and following Mitra et al. (2019), we utilize the pretrained BERT-large's MLM head to fill in the blanks. After these steps, we build a corpus derived from concepts in the KGs relevant to the SocialIQA task.

We then train our RoBERTa-based models using such a corpus with the MLM loss (Devlin et al., 2019), masking either the head or tail entities, e.g., *barbecue* or *cooking*. Further training the pretrained models on such a corpus is expected to implicitly learn commonsense knowledge in the KGs that is relevant to SocialIQA. Finally, we continue to fine-tune the model on the SocialIQA task, similar to the baseline described in Section 2.

---

[1]Initially, we tried to extract the shortest path between keywords in the KG similar to Shwartz et al. (2020). However, as ConceptNet does not disambiguate word senses, we observed that such paths usually deviate from the original semantics. For example, consider "*C: Cameron decided to have a barbecue and gathered her friends together. A_1: like attending.*" The path we found between *barbecue* and *attend* is: *barbecue–isa→dish–synonym→serve←synonym–attend*. Similarly, since we found that some of the retrieved tuples within one hop are already irrelevant, we did not use more hops to retrieve relevant tuples.

3.3 Modeling Concepts Via Attention

In the second approach, we treat the retrieved tuples as items in a cached external knowledge base (KB), which dynamically changes based on every input instance. The model can then decide the importance of each item and leverage them accordingly.

KG Attentive Representations Motivated by previous work on question answering (Seo et al., 2017; Zhu et al., 2018; Wang et al., 2018; Huang et al., 2019), which uses attention among different segments of the input, here we treat the knowledge tuples as a new segment. Specifically, we concatenate the top-k retrieved tuples and map them into the space of RoBERTa's final hidden representations as an additional segment, and then attend to it using RoBERTa's last hidden representation to generate a new KG-attentive sentence representation.

Formally, let d be the hidden dimension, l be the sequence length of the input, $H_R \in \mathbb{R}^{l \times d}$ be RoBERTa's final hidden representation for the SocialIQA input sequence for a given candidate, and $H_{KG} \in \mathbb{R}^{k \times d}$ is the representation of the k encoded tuples. We attend to H_{KG} from H_R:

$$\hat{H}_R = H_R W_1 + \mathbf{1} * b_1^T,$$
$$\hat{H}_{KG} = H_{KG} W_1 + \mathbf{1} * b_1^T,$$
$$H_R^{KG} = Softmax(\frac{\hat{H}_R \hat{HKG}^T}{\sqrt{d}}) H_{KG} \quad (1)$$

where $W_1 \in \mathbb{R}^{d \times d}$, $\mathbf{1} \in \mathbb{R}^l$ (a vector of all-ones), $b_1 \in \mathbb{R}^d$, and $H_R^{KG} \in \mathbb{R}^{l \times d}$ is the KG-attentive sentence representation.

Encoding Knowledge Tuples To obtain H_{KG}, we need to represent the tuples and project them to the RoBERTa's hidden representation space. We first convert the tuples into fixed embeddings with three different approaches:

- Pretrained KG embeddings based on Concept-Net via TransE (Bordes et al., 2013; Zhou et al., 2018).

- Pretrained word embeddings retrofitted by ConceptNet (Speer et al., 2017), where its training adjusts a word's embeddings to be close to those of its neighbors in the graph.

- Encoded tuple-converted text with templates and pretrained universal sentence encoder (USE) (Cer et al., 2018), a Transformer-based

sentence encoder that transforms text into vectors that can be used for text classification and semantic similarity.

Then we transform these embeddings of the top-k tuples using a linear transformation that is learned during training, and then concatenate all of them to form the knowledge representation $H_{KG} \in \mathbb{R}^{k \times d}$.

Fusion Layer We then combine H_R and H_R^{KG} with a *fusion layer*. Formally,

$$\tilde{H}_R = [H_R \oplus H_R^{KG}] W_2 + \mathbf{1} * b_2^T,$$
$$\tilde{h}_R = max\{\tilde{H}_R\},$$
$$S = MLP(\tilde{h}_R), \quad (2)$$

where we first transform the concatenation (denoted by $\oplus$) of H_R and H_R^{KG} to $\tilde{H}_R \in \mathbb{R}^{l \times d}$, and then perform max-pooling along the sequence dimension to obtain the condensed representation $\tilde{h}_R \in d$ for classification. Finally, we get the score $S \in \mathbb{R}^1$ for each answer via a multilayer perceptron (MLP). The model architecture is illustrated in Figure 2.
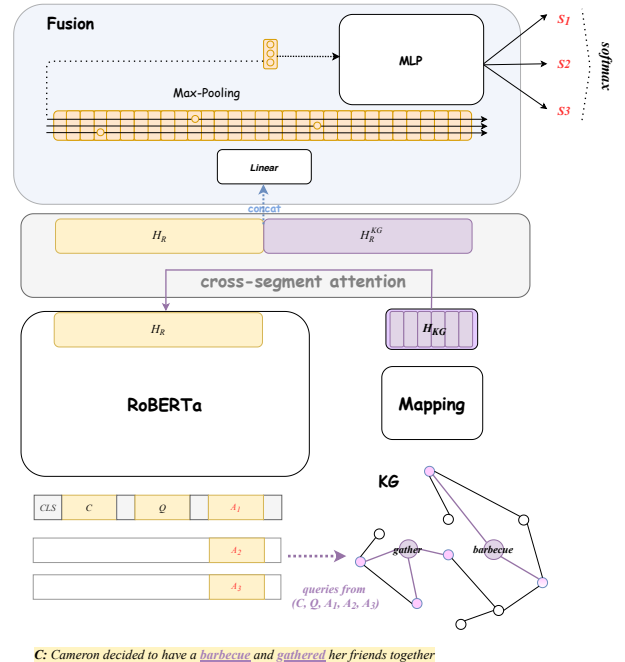

Figure 2: Illustration of the proposed model incorporating external KGs for SocialIQA.

4 Experiments and Results

4.1 Experimental Setup

We use Hugging Face's `transformers` toolkit[2] and train our models on the 33k SocialIQA training instances, running hyper-parameters search over the learning rate in $\{5e-6, 1e-5, 2.5e-5\}$, and the *effective batch size* (number of GPUs $\times$ batch size per GPU $\times$ gradient accumulation steps) in $\{8, 16, 32\}$ for the proposed models and baselines respectively, and report their best performance on the dev set. We set the maximum returned tuples of each instance to $k = 30$.

4.2 Results

Table 1 shows the results on the dev set using different methods. We can see the first pretraining method using ATOMIC performs well, which is not surprising since it is the partial source of SocialIQA, and it is likely that the model may have seen related information about dev/test set during pretraining. On the other hand, the performance of ConceptNet-pretrain suggests that without a sophisticated design, exposing too many irrelevant tuples from ConceptNet in pretraining may compromise the model's performance on the downstream task. This finding is consistent with Gururangan et al. (2020). In our analysis, we did find that some of the extracted tuples are noisy, mainly because ConceptNet is comprehensive, but it does not contain annotations for different word senses.

Model	Accuracy (%)
Baseline Sap et al. (2019b)	78.0
ConceptNet-pretrain	76.8
ATOMIC-pretrain	**79.1**
ConceptNet-attention-TransE	78.5
ConceptNet-attention-Retrofit	78.7
ConceptNet-attention-USE	**79.2**

Table 1: Comparison of different models on the SocialIQA dev set.

The second method, however, seems promising. Among its three variants (see Section 3.3), the one using TransE as knowledge embeddings performed the worst, possibly because of its much smaller dimension of the pretrained TransE embeddings ($\in \mathbb{R}^{100}$) we adopted[3], compared to the other two

variants (300 and 512) and RoBERTa-large's hidden dimension $d = 1024$. These results indicate that the second method of using ConceptNet is less sensitive to the noisy tuples because of the explicit attention mechanism, which allows the model to utilize the items in the KG selectively.

4.3 Few-Shot Learning

To demonstrate the effective utilization of external KGs, we now investigate performance in the limited training data regime. We fine-tuned our model on $5\%, 10\%,$ and 20% of SocialIQA's training instances, respectively. We compare both the explicit method, ConceptNet-attention-USE, and the implicit method, ATOMIC-pretrain, with the typical implementation of RoBERTa as the baseline. We apply MLM on SocialIQA fine-tuning (Section 2) in all the three models, since we have found it helps stabilize the training.

The results in Table 2 show that ATOMIC-pretrain performs especially well, even though it only relies on the pretraining phase to infuse the ATOMIC graph, reaching 72.9% when only 5% of training instances are used. ConceptNet-attention-USE performs the worst on the 5% setting, but better than the baseline on the other two settings. Note that a BERT-base model trained on the full training set only achieves 63.3% (Sap et al., 2019b), showing that RoBERTa model may already learn some commonsense in its pretraining phase. Furthermore, our proposed methods demonstrate that with the external knowledge graphs on relevant domains, we can obtain even better results when only a small number of annotated training instances for the downstream task are available.

	5%	10%	20%
RoBERTa + MLM	70.3	72.3	73.0
ATOMIC-pretrain	72.9	73.3	76.0
ConceptNet-attention-USE	69.7	73.3	74.6

Table 2: Results (accuracy %) when using few training instances for model fine-tuning.

5 Conclusion

In this paper, we propose two methods to introduce KGs into pretrained language models for commonsense tasks. The first one implicitly infuses relevant knowledge into MLM pretraining, while the second method uses the attention mechanism to allow pretrained language models to explicitly utilize the

dynamic query tuples. Our experiments on the SocialIQA task show that leveraging external KGs via attention outperforms the baseline pretrained language models, and the quality of the relevant graphs matters for downstream task performance. Our work can be further improved by designing better algorithms for KG retrieval in the future. Although our experiments have focused on SocialIQA with ConceptNet and ATOMIC, our method can be generalized to other similar tasks to leverage knowledge graphs.

References

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Duran, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in neural information processing systems*, pages 2787–2795.

Antoine Bosselut and Yejin Choi. 2019. Dynamic knowledge graph construction for zero-shot commonsense question answering. *arXiv preprint arXiv:1911.03876*.

Daniel Cer, Yinfei Yang, Sheng-yi Kong, Nan Hua, Nicole Limtiaco, Rhomni St John, Noah Constant, Mario Guajardo-Cespedes, Steve Yuan, Chris Tar, et al. 2018. Universal sentence encoder for english. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 169–174.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4171–4186.

Andrew Gordon, Zornitsa Kozareva, and Melissa Roemmele. 2012. Semeval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In ** SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 394–398.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. *arXiv preprint arXiv:2004.10964*.

Lifu Huang, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Cosmos qa: Machine reading comprehension with contextual 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)*, pages 2391–2401.

Haozhe Ji, Pei Ke, Shaohan Huang, Furu Wei, and Minlie Huang. 2020a. Generating commonsense explanation by extracting bridge concepts from reasoning paths. *AACL-IJCNLP*.

Haozhe Ji, Pei Ke, Shaohan Huang, Furu Wei, Xiaoyan Zhu, and Minlie Huang. 2020b. Language generation with multi-hop reasoning on commonsense knowledge graph. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

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*.

Hugo Liu and Push Singh. 2004. Conceptnet—a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226.

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*.

Arindam Mitra, Pratyay Banerjee, Kuntal Pal, Swaroop Mishra, and Chitta Baral. 2019. How additional knowledge can improve natural language commonsense question answering? *arXiv preprint arXiv:1909.08855*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

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*.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. Winogrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 8732–8740.

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

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019b. Social iqa: Commonsense reasoning about social interactions. 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)*, pages 4453–4463.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *International Conference on Learning Representations*.

Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. Unsupervised commonsense question answering with self-talk. *arXiv preprint arXiv:2004.05483*.

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*.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL)*, pages 4149–4158.

Liang Wang, Meng Sun, Wei Zhao, Kewei Shen, and Jingming Liu. 2018. Yuanfudao at semeval-2018 task 11: Three-way attention and relational knowledge for commonsense machine comprehension. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pages 758–762.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018. Commonsense knowledge aware conversation generation with graph attention. In *IJCAI*.

Haichao Zhu, Furu Wei, Bing Qin, and Ting Liu. 2018. Hierarchical attention flow for multiple-choice reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*.