Commonsense Question Answering: A Survey

Jun Yan
University of Southern California
yanjun@usc.edu

1 Introduction

When humans use their languages to communicate with each other, they often rely on broad implicit assumptions. Humans learn and use this kind of assumptions in everyday life, which makes their language concise without lacking precision. However, machines by nature don't have such background knowledge. Machine learning models can't accumulate human's commonsense through interacting with the environment. Therefore, empowering Natural Language Processing (NLP) techniques with commonsense knowledge is one of the major long-term goals for Artificial Intelligence (AI).

Question Answering (QA) is a Natural Language Understanding (NLU) task requiring both language processing and knowledge reasoning. When commonsense knowledge outside the given text is needed to answer the question, the task is called Commonsense Question Answering. Therefore, the main focus of the commonsense question answering task is how to incorporate commonsense knowledge and conduct reasoning.

2 Resources

In this section, we introduce some recent representative datasets and knowledge resources. We can't cover all of them due to limited space.

2.1 Datasets

Created by Zellers et al. (2018), **SWAG** contains 113k multiple choice questions. Given a partial description, the task is to select the most probable next action. **Commonsense QA** (Talmor et al., 2018) contains 12k multiple choice questions asking for a target concept from ConceptNet (Speer et al., 2017). **CODAH** (Chen et al., 2019) is adversarially-constructed with 2.8k multiple choice questions that make pretrained models struggle to answer.

2.2 Commonsense Knowledge

Commonsense knowledge is mainly available in two forms. One is unstructured large-scale corpora, which implicitly encodes human's world knowledge. Vast numbers of corpora like Wikipedia for training large scale language models can be leveraged. The other is structured knowledge graphs (bases). For example, ConceptNet (Speer et al., 2017) is a knowledge graph whose nodes are concepts in the form of words or phrases and edges are relations between connected nodes. Atomic (Sap et al., 2019) is a knowledge graph about events and their if-then relations.

3 Approaches Overview

Popular and effective approaches can be easily found on the leaderboard of each task. Approaches can be categorized into two classes: (1) data-centric (2) model-centric.

Data-centric methods assume knowledge can directly learned from large copora in an unsupervised way. For example, simply finetuing RoBERTa (Liu et al., 2019) on the commonsense QA dataset can beat many carefully-designed models. Some work (Zhang et al., 2019) also tries to incorporate entity knowledge into the training phase of the model. These methods want to use pretrained language models (e.g. BERT (Devlin et al., 2018), XLNet (Yang et al., 2019), RoBERTa (Liu et al., 2019)) as an unified knowledge encoder to solve the question answering task. While they have continuously set new records on various NLP tasks by training with larger corpora and more advanced hardware support as well as tuning tricks, some analysis paper (Niven and Kao, 2019) argues that they are proficient at memorizing facts and finding "spurious statistical cues", which are results of overfitting. What's more, it's believed that

the language model can not really do reasoning, which limits its potential towards solving NLU tasks that relys on reasoning.

Model-centric methods still rely on powerful pretrained models to learn the language semantics. The difference is that they extend the underlying pretrained models with structures which are designed to explicitly incorporate knowledge that are helpful for specific tasks. The typical workflow is as: A. encoding the input; B. extracting evidence; C.reasoning over evidence. Pretrained language models can be used as the input encoder. Therefore, the main challenge here is how to extract relevant evidence and how to perform reasoning over the extracted evidence. We will introduce some methods in the following sections.

4 Extracting Evidence

There are two main sources to extract knowledge: plain text and knowledge graph. Chen et al. (2017) propose DrQA which consists of a document retriever and a document reader to locate and incorporate helpful knowledge in the Wikipedia. Many knowledge-augmented works (Bauer et al., 2018; Lin et al., 2019) directly use a knowledge graph in the related domain. Lv et al. (2019) extract helpful knowledge from both ConceptNet and Wikipedia. While text data like Wikipeida is has high coverage, structured data like knowledge graph can provide relation information which is necessary for knowledge reasoning. Therefore, it's helpful to have a combined knowledge sources of plain text and knowledge graphs. Plain text may help alleviate the low coverage problem of the knowledge graph and improve concept grounding or entity linking.

Given the knowledge source and input, the next step is to extract related knowledge. Here we focus on the knowledge graph cases, and the task is called concept grounding or entity linking according to the node type. This part is usually not claimed as a contribution by commonsense QA papers. They usually use off-the-shelf tools (e.g. entity linker) or develop simple string matching rules to identify matched entities/concepts on the knowledge graph. After locating these "root" nodes, an extractive way to get the evidence graph is to construct a subgraph covering all "root" nodes. However, finding the minimal spanning subgraph is a NP-complete problem. Therefore, researchers develop heuristics (Lv et al., 2019; Lin et al., 2019) for graph construction or formulate the path finding problem as an optimization problem which can be efficiently solved. There isn't a comprehensive study on how the quality of the extracted graph affects the final performance. (Lin et al. (2019) find path pruning to be helpful.) Heuristic algorithms prove to work in some papers while the potential of optimization-based algorithms is still not clear. Another method is to construct the evidence graph in a generative way, which hasn't been studied to the best of our knowledge. The advantage is that it can capture the semantic meaning of edges in a more flexible way as well as avoid running selection algorithms on the huge knowledge graph.

5 Reasoning over Evidence

After we have an evidence graph, the next step is to reason over it, which corresponds to message passing in Graph Neural Networks (GNNs). Therefore, many GNN variants (Wu et al., 2019; Zhou et al., 2018) can be adopted for reasoning. Marcheggiani and Titov (2017); Zhang et al. (2018) find that Relational-GCN (Schlichtkrull et al., 2018) tends to over-parameterize the model. (Lv et al., 2019; Lin et al., 2019) both use GCNs on the undirected graph while Lin et al. (2019) propose an additional LSTM-based path encoder. The lesson here is that we should not only use symbolic knowledge from the graph, but also leverage semantic clues from the input sequence. Xiao et al. (2019) propose Dynamically Fused Graph Network to deal with constructed graph, which is similar to the setting of generative graph construction. It also shows that entity-level reasoning and token-level contexts are both important for question answering. For these graph neural networks, the attention mechanism is usually helpful for feature aggregation as well as interpreting the results and debugging.

6 Conclusion

In this paper, we review typical resources and approaches for commonsense question answering. Note that techniques for commonsense QA also closely relate to broader fields like general question answering, subgraph selection, graph reasoning and graph embedding. Those papers should also be referenced when doing research. Those papers should also be referenced when doing research.

References

Lisa Bauer, Yicheng Wang, and Mohit Bansal. 2018. Commonsense for generative multi-hop question answering tasks. *arXiv preprint arXiv:1809.06309*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Michael Chen, Mike D'Arcy, Alisa Liu, Jared Fernandez, and Doug Downey. 2019. Aqua: An adversarially authored question-answer dataset for common sense. *arXiv preprint arXiv:1904.04365*.

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*.

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

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*.

Shangwen Lv, Daya Guo, Jingjing Xu, Duyu Tang, Nan Duan, Ming Gong, Linjun Shou, Daxin Jiang, Guihong Cao, and Songlin Hu. 2019. Graph-based reasoning over heterogeneous external knowledge for commonsense question answering. *arXiv preprint arXiv:1909.05311*.

Diego Marcheggiani and Ivan Titov. 2017. Encoding sentences with graph convolutional networks for semantic role labeling. *arXiv preprint arXiv:1703.04826*.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. *arXiv preprint arXiv:1907.07355*.

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.

Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Rianne Van Den Berg, Ivan Titov, and Max Welling. 2018. Modeling relational data with graph convolutional networks. In *European Semantic Web Conference*, pages 593–607. Springer.

Robert Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Thirty-First AAAI Conference on Artificial Intelligence*.

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

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S Yu. 2019. A comprehensive survey on graph neural networks. *arXiv preprint arXiv:1901.00596*.

Yunxuan Xiao, Yanru Qu, Lin Qiu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019. Dynamically fused graph network for multi-hop reasoning. *arXiv preprint arXiv:1905.06933*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *arXiv preprint arXiv:1906.08237*.

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *arXiv preprint arXiv:1808.05326*.

Yuhao Zhang, Peng Qi, and Christopher D Manning. 2018. Graph convolution over pruned dependency trees improves relation extraction. *arXiv preprint arXiv:1809.10185*.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. Ernie: Enhanced language representation with informative entities. *arXiv preprint arXiv:1905.07129*.

Jie Zhou, Ganqu Cui, Zhengyan Zhang, Cheng Yang, Zhiyuan Liu, and Maosong Sun. 2018. Graph neural networks: A review of methods and applications. *arXiv preprint arXiv:1812.08434*.