Do Language Models Perform Generalizable Commonsense Inference?

Peifeng Wang[1,2], **Filip Ilievski**[2], **Muhao Chen**[1,2], **Xiang Ren**[1,2]
[1]Department of Computer Science, University of Southern California
[2]Information Sciences Institute, University of Southern California
{peifengw,muhaoche,xiangren}@usc.edu, ilievski@isi.edu

Abstract

Inspired by evidence that pretrained language models (LMs) encode commonsense knowledge, recent work has applied LMs to automatically populate commonsense knowledge graphs (CKGs). However, there is a lack of understanding on their generalization to multiple CKGs, unseen relations, and novel entities. This paper analyzes the ability of LMs to perform generalizable commonsense inference, in terms of *knowledge capacity*, *transferability*, and *induction*. Our experiments with these three aspects show that: (1) LMs can adapt to different schemas defined by multiple CKGs but fail to reuse the knowledge to generalize to new relations. (2) Adapted LMs generalize well to unseen subjects, but less so on novel objects. Future work should investigate how to improve the transferability and induction of commonsense mining from LMs.[1]

1 Introduction

Large-scale commonsense knowledge graphs (CKGs), like ConceptNet (Speer et al., 2017) and ATOMIC (Sap et al., 2019), store structured knowledge that can benefit various knowledge-driven applications. Given the usefulness of CKGs, but also their inability to flexibly provide information, (Paulheim, 2018), recent work has paid much attention to populating CKGs with commonsense knowledge mined from pretrained language models (LMs) (Wang et al., 2020c; Bosselut et al., 2019). Enhancing the knowledge of CKGs is essential to support reasoning on downstream tasks (Talmor et al., 2019; Wang et al., 2020b; Young et al., 2018).

The task of completing CKGs has typically been posed as *commonsense knowledge inference*, where the goal is to predict the object of a fact triplet, given its *subject* and a *relation* (*predicate*) (Petroni et al., 2019; Bosselut et al., 2019). Commonsense inference techniques, such as COMET (Bosselut et al., 2019), typically fine-tune an LM, like GPT (Radford et al., 2018), over the training set from a single CKG. While such methods are able to dynamically enhance the completeness of CKGs, their application so far has been limited to the relation set of the source (training) CKG (Da et al., 2021). In addition, the generated object concepts are found to be largely biased towards the ones in the training set (Wang et al., 2020a). It remains unclear to which extent LMs can generalize to multiple CKGs, new relations, and novel objects. To this end, we pose the question: *do language models perform generalizable commonsense inference?*

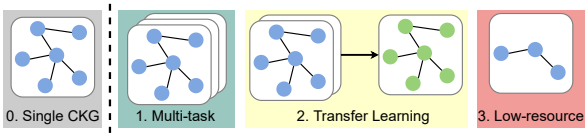

Figure 1: Unlike previous studies that adapt LM on one single CKG (0), we investigate LM's three aspects of **generlizability**: (1) *knowledge capacity* by multi-task learning, (2) *transferability* by transfer learning and (3) *induction* by controlled low-resource learning.

To answer this question, we study three aspects of the LM generalizability for commonsense inference, namely: knowledge capacity, transferability, and induction. To measure the *knowledge capacity* ability of LMs, we examine whether LMs can be adapted to multiple CKGs simultaneously, and tested on each of the CKGs. We test their *transferability* by assessing whether an initial adaptation of a LM on multiple source CKGs can reduce the effort on further adapting it to a new CKG. The *inductive* power of LMs is measured by varying the overlap between the objects in the training and test splits of a CKG. The overview of our analysis is depicted in Figure 1. Our results show that LMs are able to infer knowledge for multiple CKGs simultaneously without loss of performance on the

[1]The code is avaiable at https://github.com/wangpf3/LM-for-CommonsenseInference.

target inference task, though the transferability of knowledge across tasks is limited. In addition, we observe that the inductive power of LMs for commonsense inference relies heavily on whether an object is observed during training.

2 Analysis Setup

To shed light on the LM's generalizability for commonsense inference, we investigate: whether LMs have the capability to adapt to multiple CKGs (*Q1: capacity*), whether LMs can reuse the knowledge learned from source CKGs to efficiently adapt to a target CKG (*Q2: transferability*), and whether LMs can predict unseen objects or mainly repeat the observed ones (*Q3: induction*). In this Section, we define the task, the CKGs we consider, our experimental settings, and relate to prior studies.

2.1 Task Formulation

Following Hwang et al. (2020); Da et al. (2021), we formalize *commonsense inference* as a task of predicting the object of a triplet, given a pair of *(subject, relation)* as input. The subject s and the object o are both expressed as free-form phrases, while the relation r is a predefined relation type from the CKG. A training example from ConceptNet could have `(go to a concert, MotivatedByGoal)` as input, and `listen to music` as output. Assuming that a CKG is given, the goal is to leverage the commonsense triplets in the CKG as training examples to adapt the LM for commonsense inference.

2.2 CKG Datasets

We consider three large and popular CKGs, with different foci:(1) **ConceptNet**'s broad set of commonsense knowledge includes taxonomic (e.g., `IsA`), utility (e.g., `UsedFor`), and temporal knowledge (e.g., `HasPrerequisite`). It combines crowdsourced knowledge with that from existing sources, such as WordNet. We use its ConceptNet-100K subset, collected by Li et al. (2016). (2) **TupleKB** (Dalvi Mishra et al., 2017) focuses on scientific commonsense knowledge like `(salt, dissolve in, water)`. It is constructed through an information extraction pipeline. (3) **ATOMIC** (Sap et al., 2019) has social commonsense knowledge about causes and effects of everyday events, and mental states (e.g., `xIntent`) of their participants. It is created by crowdsourcing.

As indicated by Jastrzebski et al. (2018), a large proportion of the subjects in the test set of ConceptNet-100K overlap with its training set, while TupleKB does not provide an official split. Thus, we (re-)split these two datasets to ensure that the subjects of testing triplets do not appear in the training set. This criterion is also consistent with how the ATOMIC dataset is constructed.

2.3 Experimental Settings

Multi-task Learning To answer Q1, we adapt an LM with balanced training data from ConceptNet, TupleKB, and ATOMIC. We sample 8 triplets from each dataset to form one training batch.

Transfer Learning To provide insight into Q2, we adopt transfer learning under a leave-one-out strategy. In this setting, we adapt an LM on two of the three CKGs, and then we further adapt it on the third target CKG. Moreover, we study the data efficiency of this transfer learning by down-sampling each training set to $x = \{1, 20, 50\}\%$, in order to see whether the LM can adapt to the target CKG with less training effort. Fine-tuning on data as small as 1% training set may suffer from instability, and results may change dramatically given a new split of training data (Gao et al., 2020). To control the randomness, we re-sample the 1% training data 5 times with a fixed set of random seeds and report the average performance instead.

Controlled Low-resource Learning To answer Q3, we design a controlled experiment, where we first split the training set into two disjoint subsets depending on whether the triplets in the original training set contain objects that exist in the test set or not. We denote the subset where the objects of the triplets appear in testing data as Ω. We sample $x = \{0, 25, 50, 100\}\%$ of the training triplets in Ω for adapting the LM. During the evaluation, we also separate the test set into two disjoint subsets, according to whether the objects are seen in the original full training set. The results on these two split test sets are reported separately for each adapted LM.

Evaluation Protocol For each *(subject, relation)* pair in the test set, we treat all their objects as ground truth references for evaluating the model inference. We report scores for commonly used automatic evaluation metrics for text generation: BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), and METEOR (Banerjee and Lavie, 2005), which are shown to be consistent with human judgements (Hwang et al., 2020). During experiments, we observe a high correlation among these differ-

Adaptation method	Input	Learnable params
Zero-shot (ZS)	(s, r)	N/A
ZS+demo	$(s^{'}, r, o^{'}, s, r)$	N/A
Fine-tuning (FT)	(s, r)	Transformer (LM)
FT+demo	$(s^{'}, r, o^{'}, s, r)$	Transformer (LM)
Adapter tuning (AT)	(s, r)	Adapter

Table 1: Methods for using LMs to conduct commonsense inference. *"+demo"* means prepending a demonstration triplet $(s^{'}, r, o^{'})$ before the input tuple.

ent metrics and choose to report METEOR in the main text and other metrics in the appendix.

2.4 Connections to Prior Studies

Earlier works (Li et al., 2016; Jastrzebski et al., 2018; Davison et al., 2019) poses the CKG completion task as triplet classification, where the goal is to score the plausibility of a complete triplet. COMET (Bosselut et al., 2019) is the first to cast this task as commonsense inference with LMs. Follow-up contributions utilize COMET as a commonsense provider in various downstream tasks (Bosselut and Choi, 2021; Ammanabrolu et al., 2021; Chakrabarty et al., 2020), thus providing evidence for LM's generalization to previously unseen scenarios. Further efforts include Hwang et al. (2020), which show that the quality of the training triplets is a key factor of adapting LMs, and (Da et al., 2021), which investigates how to learn COMET in a few-shot learning setting. Meanwhile, the study by Wang et al. (2020a) indicates the limited generalization of COMET. Ma et al. (2021) also adapt LMs simultaneously on multiple CKGs, albeit their goal is to improve downstream performance rather than CKG inference. In this paper, we aim to provide a more comprehensive study of a LM's generalizability for CKG inference.

3 Method

While a set of pretrained LMs exists, we adopt a widely used generative model, GPT2 (Radford et al., 2019), as our baseline LM. The investigation of other generative LMs is orthogonal to our analysis. We experiment with its largest version, GPT2-XL, which contains 48 transformer layers (Vaswani et al., 2017), ensuring sufficient capacity for storing knowledge acquired during its pretraining. We introduce our experimental method as follows.

Commonsense Inference with LMs Given a training triplet *(s,r,o)*, we represent s and o as sequences of tokens, $\mathbf{x}_s$ and $\mathbf{x}_o$, which is trivial given that they are already expressed as phrases. As for the relation r, we convert it by using a template taken from the literature (Davison et al., 2019) into a natural-language phrase $\mathbf{x}_r$, e.g., `IsA` is converted to "is a". This has been shown to facilitate efficient adaptation of LMs (Da et al., 2021). Note that we do not explicitly provide the LMs with the information about the source CKG of the triplet as input (e.g., prepending a related special token to the triplet).

Adapting LMs with Commonense Knowledge The training objectives for adapting LMs is to maximize the probability of generating the object phrase $\mathbf{x}_o$ given the tuple $(\mathbf{x}_s, \mathbf{x}_r)$. During inference, we adopt greedy decoding to obtain the predicted object from the adapted LM.

There have been various techniques developed for adapting pretrained LMs to downstream tasks (Howard and Ruder, 2018; Chen et al., 2020). Moreover, previously only the vanilla **Fine-tuning**, i.e., updating the whole LM architecture during training, has been employed to adapt LMs for commonsense inference (Bosselut et al., 2019; Hwang et al., 2020; Da et al., 2021). To obtain comprehensive results that are not specific to one particular way of fine-tuning, here we investigate two more alternatives, each of which has their own advantage when considered in different contexts.

Fine-tuning with Demonstration (FT+demo) Combining the ideas of fine-tuning and in-context learning (Brown et al., 2020), this technique (Gao et al., 2020) adds a demonstration to each input as additional context and fine-tunes the whole LM as usual. Incorporating demonstrations is shown to boost performance when the amount of training data is extremely limited. In our case, a demonstration is a top-1 training triplet $(s^{'}, r, o^{'})$, ranked according to the cosine similarity between the embedding of the input tuple (s, r) and the embeddings of the training tuples with the same relation type r. The tuple embeddings are given by a pretrained Sentence-BERT (Reimers and Gurevych, 2019). For instance, a demonstration (`go to restaurant`, `UsedFor`, `eat out`) would be added before the input (`go to pub`, `UsedFor`). With the demonstrated triplets, the LM could learn to understand the schema of the CKG instead of simply learning the knowledge from the training data.

Adapter Tuning (AT) Unlike fine-tuning, adapter tuning (Houlsby et al., 2019) fixes the entire LM and adds one trainable adapter right before the skip connection in each transformer layer of the LM,

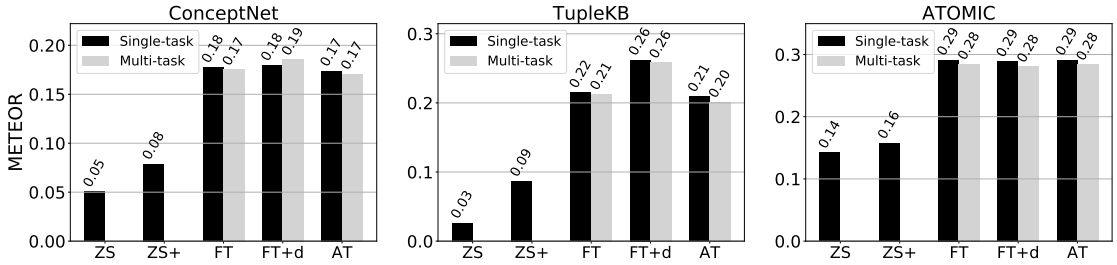

Figure 2: **Results (METEOR) for *knowledge capacity* of LMs. "FT+d" refers to FT+demo.** We find no notable performance drop for any method trained in the multi-task setting.

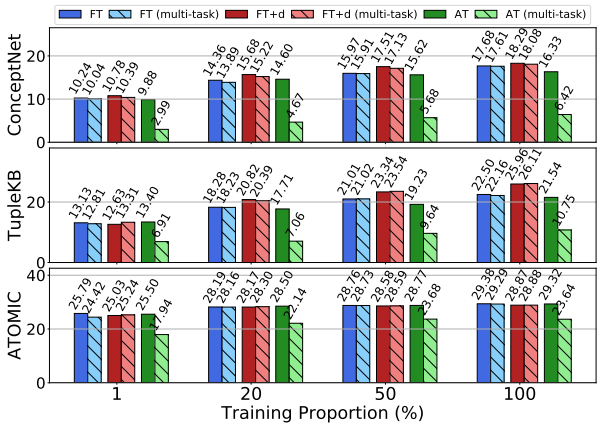

Figure 3: **Results (METEOR) for LM *transferability*.** "FT+d" refers to FT+demo. Across datasets, we do not observe that adapting to the source CKGs would enable the LMs to adapt to the target CKG better or more easily.

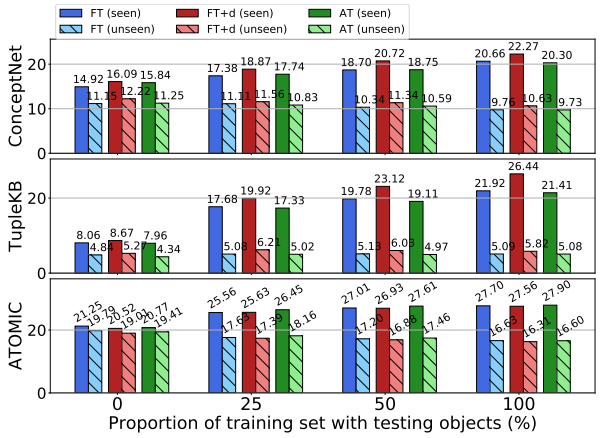

Figure 4: **Results (METEOR) for LM *induction*.** "FT+d" refers to FT+demo. All the methods perform better on predicting facts that contain seen objects, while the performance degrades when less objects are seen during training.

which is more parameter-efficient. Each adapter is a two-layer bottleneck network with a skip-connection internally. Following Houlsby et al. (2019), the parameters of the bottleneck network are initialized close to zero so that the adapter approximates an identity function from the beginning.

We compare to two additional baselines, both using GPT2-XL in a zero-shot setting: **Zero-shot (ZS)** is fed with the same input as Fine-tuning, while zero-shot with demonstrations (**ZS+demo**) combines the input plus demonstration, as in the FT+demo method. By investigating all these methods, we aim to understand the influence of different adaptation techniques on the models' performance. Table 1 summarizes the set of methods which we consider in this paper.

4 Results and Discussion

Knowledge Capacity (Q1) The results that quantify the knowledge capacity of LMs for commonsense inference over multiple CKGs with METEOR scores are shown in Figure 2. The complete results including other metrics can be found in the appendix. All adaptation methods perform considerably better than the zero-shot baselines, indicating the benefit of adaptation. There is no clear distinction between the adaptation methods, though FT+demo performs slightly better than the others across CKGs. Most importantly, we find no notable performance drop for any method in the multi-task training setup despite the challenge that there is limited overlap between these CKGs. Only 10.0% of the facts from ATOMIC can be found in ConceptNet (Hwang et al., 2020) while 8.4% of the facts from ConceptNet can be found in TupleKB (Dalvi Mishra et al., 2017) [2]. This indicates the prominent capacity of LMs to simultaneously adapt to different CKGs. Nevertheless, the results reveal that learning different CKGs jointly do not interfere with each other positively (via knowledge sharing) or negatively (due to overfitting).

Transferability (Q2) Figure 3 shows the obtained results regarding the transferability of LMs. Across different CKGs and for any training data size, we observe no indications that adapting to the source CKGs enhances the performance on the target CKG. On the contrary, adapting from source CKGs

[2] We also try to breakdown the results by relation types and do not observe correlation between the relation-wise performance and the extent of overlap.

even hurts the performance of the Adapter-tuning method, revealing that this method overfits to the source CKGs. Overall, we conclude that LMs cannot reuse the knowledge learned from the source CKGs to improve the performance on the target CKG or achieve the same performance with less training data. Thus, we call for future study on developing more effective adaptation methods.

Induction (Q3) The results in Figure 4 show that without down-sampling ($x = 100\%$), all methods perform much better on predicting facts that contain seen objects, and their performance degrades more when less object entities are seen to training. Meanwhile, the performance on facts with unseen objects stays roughly unaffected. This indicates a key limitation of the LMs: they adapt notably better on seen objects. Since the training set and test set do not share subjects, we conclude that the generalizability of the LM is largely dependent on finding the relationship between unseen subjects and observed objects. We thus posit that a novel strategy for adapting LMs while retaining the knowledge acquired during pre-training is necessary for better generalizability. Promising directions here are prefix tuning (Li and Liang, 2021) or including an additional objective during adaptation which would encourage the generation of novel objects.

5 Conclusion

This work conducted a focused study of three aspects of the generalizability of LMs for commonsense inference: knowledge capacity, transferability, and induction. We experiment with five methods of using a generative LM and three representative CKGs. Despite their capability to accommodate multiple CKGs, we have observed that LMs have limited ability to transfer knowledge across CKGs. Moreover, their adaptation relies heavily on whether the objects to predict are seen during training. These findings help our understanding of LMs' adaptation behavior on commonsense inference, and highlight the need for future work to improve their transferability and induction.

Acknowledgments

We thank the anonymous reviewers for their insightful comments. This material is based upon work sponsored by the DARPA MCS program under Contract No. N660011924033 with the United States Office Of Naval Research.

References

Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O Riedl. 2021. Automated storytelling via causal, commonsense plot ordering. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Antoine Bosselut and Yejin Choi. 2021. Dynamic knowledge graph construction for zero-shot commonsense question answering. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence (AAAI)*.

Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. COMET: Commonsense transformers for automatic knowledge graph construction. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy. Association for Computational Linguistics.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Tuhin Chakrabarty, Smaranda Muresan, and Nanyun Peng. 2020. Generating similes effortlessly like a pro: A style transfer approach for simile generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6455–6469, Online. Association for Computational Linguistics.

Sanyuan Chen, Yutai Hou, Yiming Cui, Wanxiang Che, Ting Liu, and Xiangzhan Yu. 2020. Recall and learn: Fine-tuning deep pretrained language models with less forgetting. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7870–7881, Online. Association for Computational Linguistics.

Jeff Da, Ronan Le Bras, Ximing Lu, Yejin Choi, and Antoine Bosselut. 2021. Understanding few-shot

commonsense knowledge models. *arXiv preprint arXiv:2101.00297*.

Bhavana Dalvi Mishra, Niket Tandon, and Peter Clark. 2017. Domain-targeted, high precision knowledge extraction. *Transactions of the Association for Computational Linguistics*, 5:233–246.

Joe Davison, Joshua Feldman, and Alexander Rush. 2019. Commonsense knowledge mining from pretrained models. 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 1173–1178, Hong Kong, China. Association for Computational Linguistics.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 2790–2799. PMLR.

Jeremy Howard and Sebastian Ruder. 2018. Universal language model fine-tuning for text classification. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 328–339, Melbourne, Australia. Association for Computational Linguistics.

Jena D Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2020. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. *arXiv preprint arXiv:2010.05953*.

Stanislaw Jastrzebski, Dzmitry Bahdanau, Seyedarian Hosseini, Michael Noukhovitch, Yoshua Bengio, and Jackie Cheung. 2018. Commonsense mining as knowledge base completion? a study on the impact of novelty. In *Proceedings of the Workshop on Generalization in the Age of Deep Learning*, pages 8–16, New Orleans, Louisiana. Association for Computational Linguistics.

Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. 2016. Commonsense knowledge base completion. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1445–1455, Berlin, Germany. Association for Computational Linguistics.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Kaixin Ma, Filip Ilievski, Jonathan Francis, Yonatan Bisk, Eric Nyberg, and Alessandro Oltramari. 2021. Knowledge-driven Data Construction for Zero-shot Evaluation in Commonsense Question Answering. In *35th AAAI Conference on Artificial Intelligence*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Heiko Paulheim. 2018. How much is a triple? estimating the cost of knowledge graph creation. In *Proceedings of the 17th International Semantic Web Conference*.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

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

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

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. 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 3982–3992, Hong Kong, China. Association for Computational Linguistics.

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 *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 3027–3035. AAAI Press.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*, pages 4444–4451. AAAI Press.

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: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Cunxiang Wang, Jinhang Wu, Luxin Liu, and Yue Zhang. 2020a. Commonsense knowledge graph reasoning by selection or generation? why? *arXiv preprint arXiv:2008.05925*.

Haoyu Wang, Muhao Chen, Hongming Zhang, and Dan Roth. 2020b. Joint constrained learning for event-event relation extraction. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 696–706, Online. Association for Computational Linguistics.

Peifeng Wang, Nanyun Peng, Filip Ilievski, Pedro Szekely, and Xiang Ren. 2020c. Connecting the dots: A knowledgeable path generator for commonsense question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4129–4140, Online. Association for Computational Linguistics.

Tom Young, Erik Cambria, Iti Chaturvedi, Hao Zhou, Subham Biswas, and Minlie Huang. 2018. Augmenting end-to-end dialogue systems with commonsense knowledge. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 4970–4977. AAAI Press.

		BLEU-2		ROUGE-L		METEOR	
		single-task	multi-task	single-task	multi-task	single-task	multi-task
ConceptNet	Zero-shot	0.0069	NA	0.1009	NA	0.0506	NA
	ZS+demo	0.0284	NA	0.1281	NA	0.0787	NA
	Adapter-tuning	0.1289	0.1279	0.2598	0.2560	0.1739	0.1706
	Fine-tuning	0.1325	0.1286	0.2629	0.2575	0.1775	0.1749
	FT+demo	**0.1333**	**0.1398**	**0.2678**	**0.2738**	**0.1795**	**0.1851**
TupleKB	Zero-shot	0.0017	NA	0.0999	NA	0.0263	NA
	ZS+demo	0.0099	NA	0.2748	NA	0.0869	NA
	Adapter-tuning	0.1383	0.1323	0.3785	0.3627	0.2094	0.2010
	Fine-tuning	0.1371	0.1388	0.3985	0.3812	0.2151	0.2122
	FT+demo	**0.1699**	**0.1698**	**0.4902**	**0.4714**	**0.2622**	**0.2580**
ATOMIC	Zero-shot	0.0436	NA	0.2523	NA	0.1419	NA
	ZS+demo	0.0808	NA	0.2233	NA	0.1572	NA
	Adapter-tuning	**0.2161**	0.2035	**0.4008**	0.3890	**0.2913**	0.2832
	Fine-tuning	0.2125	0.2057	0.3982	**0.3908**	0.2913	**0.2843**
	FT+demo	0.2111	**0.2070**	0.3915	0.3868	0.2887	0.2800

Table 2: Results of all the evaluation metrics for the knowledge capacity experiments.

A Appendix

A.1 Dataset Statistics

[h]

	Train	Dev	Test
ConceptNet100k	79,770	10,203	10,027
TupleKB	98,674	12,357	12,427
ATOMIC	578,002	64,902	71,127

Table 3: CKG Dataset Statistics.

A.2 Implementation Details

The GPT2-XL language model we adopted in this work has 1558M parameters in total. We train all the models on a V100 GPU. As for hyper-parameters, we adopt the commonly-used learning rate (1e-5) and batch size (16) for adapting GPT2, except that in the multi-task learning setting, the batch size is 24 (8 samples from each CKG).

A.3 Additional Results

See Table 2 for the full results of all the evaluation metrics considered in this paper.