Few-Shot Commonsense Knowledge Models

Anonymous EMNLP submission

Abstract

Providing natural language processing systems with commonsense knowledge at scale is a critical challenge for achieving grounded language understanding. Manually constructed commonsense knowledge bases are difficult to scale to the quality and situational diversity necessary for general-purpose language understanding. More recently, commonsense knowledge models (Bosselut et al., 2019) have shown promise as general purpose tools that can hypothesize commonsense knowledge on-demand. While these systems have been broadly used for a wide variety of natural language applications, they remain limited by the data investment required to train them (hundreds of thousands of annotated examples).

In this work, we define a pipeline for training commonsense knowledge models effectively with few examples per commonsense relations. Human quality ratings are within 6% of fully supervised settings, despite training on very few tuples. We perform five separate studies on training commonsense knowledge models using few examples, providing an empirical study on how to best introduce new relations to knowledge models with limited tuples.

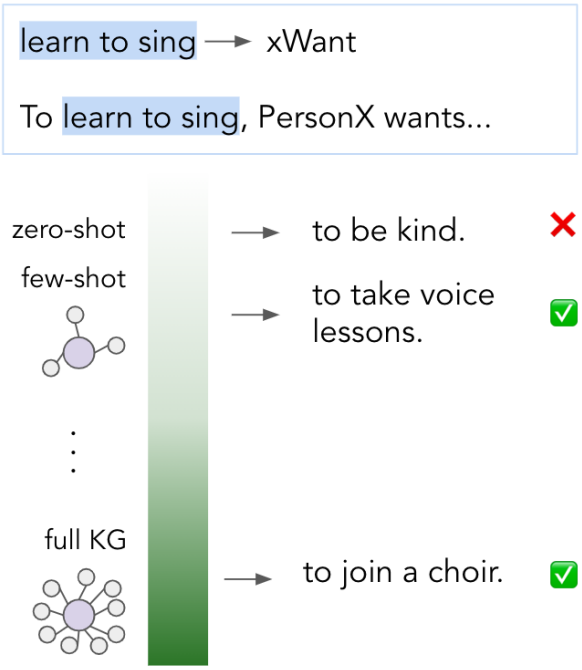

Figure 1: Humans can apply previous knowledge to a new relationship by learning from few examples, however, supervised knowledge models still need to fine-tune on many examples. We show that models *learn the interface* of relations in few examples, allowing effective leverage of pre-training knowledge for commonsense knowledge modeling.

1 Introduction

Recently, it has been shown that large-scale language systems are an effective architecture for consuming facts and rules about the world and reasoning about them (Bosselut et al., 2019; Clark et al., 2020; Khashabi et al., 2020), typically by pre-training on large corpora (Vaswani et al., 2017; Devlin et al., 2019; Liu et al., 2019; Radford, 2018) and evaluating reasoning ability by fine-tuning (Richardson and Sabharwal, 2019; Talmor et al., 2019). Despite these successes, general purpose language understanding systems grounded in world knowledge remain elusive. The desired interplay between knowledge and language representations is not reliably captured by current leading training schemes for NLP systems. Consequently, practitioners turn to commonsense knowledge graphs to ground language with explicit rules about the world.

However, these approaches are limited by the breadth and diversity of commonsense knowledge graphs. These structured, symbolic representations can not be grown with high fidelity to the scale needed to model general purpose commonsense knowledge. Manual approaches to knowl-

edge graph construction yield reliable and precise annotations of commonsense relationships, but they are cost-prohibitive: human annotation is not cheap (Speer et al., 2017; Sap et al., 2019). Automatic, extraction-based methods to commonsense knowledge graph construction (Zhang et al., 2020a) achieve much greater scale, but yield tuples of lower precision and questionable fidelity (Hwang et al., 2021). Reporting bias (Gordon and Van Durme, 2013) — the idea that obvious details go unstated in text (Grice et al., 1975) — limits the degree of useful commonsense knowledge that can be directly extracted from text.

More recently, commonsense knowledge models have emerged as a potential solution this bottleneck (Bosselut et al., 2019). These models learn to represent knowledge graphs implicitly and accessibly. They are pretrained on large text corpora (Radford et al., 2019; Lewis et al., 2020; Raffel et al., 2019) and then further fine-tuned on examples from a knowledge graph (Bosselut et al., 2019). This two stage process allows them to transfer implicit, but inaccessible, representations of knowledge learned from language (Petroni et al., 2019) to the task of hypothesizing declarative knowledge. Since their inception, they have become a popular mechanism for providing commonsense knowledge *on-demand* to downstream NLP task systems (Chakrabarty et al., 2020; Ammanabrolu et al., 2020; Kearns et al., 2020; Majumder et al., 2020).

Despite their successes, commonsense knowledge models remain fundamentally restricted by the knowledge graph used to support knowledge transfer. To learn representations of the relations in the knowledge graph, they require large numbers of training examples (in the order of hundreds of thousands; Bosselut et al., 2019). Furthermore, they can only learn the relations that make up the schema of the knowledge base. In cases where broader and more diverse notions of common sense are required for situational understanding, these systems cannot generalize from the fixed relationships present in their seed KG. To reach the level of relation coverage needed for broad applicability of NLP systems, knowledge models must be able to rapidly ingest new formulations of commonsense knowledge.

In this work, we perform the first comprehensive study of the few-shot learning potential of commonsense knowledge models. We explore five different dimensions for knowledge model transfer in the few-shot setting: learning vs. context augmentation, model scale, input representation, example selection, and learning schedule. Along these axes, we provide novel insights into the few-shot learning behavior of knowledge models. Our empirical results showcase the following main takeaways:

- We present a pipeline for training commonsense knowledge models for few-shot relation learning. Few-shot learning is more flexible than few-shot adaptation (§4.1), allowing a 16x smaller model to exceed generalization performance of GPT-3 (Brown et al., 2020).
- Under similar example budgets and training schemes, a commonsense knowledge model with more parameters generalizes more effectively to new examples of knowledge (§4.2).
- Expressive prompting that uses language descriptions of relations allows for highly efficient knowledge transfer — 10x-100x fewer examples (§4.3).
- When annotating examples for few-shot transfer, situation breadth (*i.e.*, diverse knowledge tuple heads) is more beneficial than situation depth (*i.e.*, multiple tails for the same head) across any example budget (§5.1).
- Learning from other relations improves zero-shot relation induction, but its benefits subside when as few as 30 examples of a new relation are available for transfer (§5.2).

2 Background

2.1 Commonsense Knowledge Graphs

Commonsense knowledge graphs are structured, relational representations of commonsense knowledge. They showcase a large variety of commonsense entities and the relations between them. In our study, we use the ATOMIC$^{20}_{20}$ (Hwang et al., 2021) knowledge graph, a commonsense knowledge graph with 1.33M everyday inferential knowledge tuples about entities and events. ATOMIC$^{20}_{20}$ represents a large-scale commonsense repository of textual descriptions that encode social and physical aspects of common human experiences. Across its 1.33M tuples, ATOMIC$^{20}_{20}$ captures information about 23 relationship types: 9 relations about social interactions, 7 physical-entity commonsense relations, and 7 event-centered commonsense relations.

2.2 Commonsense Knowledge Models

Commonsense knowledge models represent commonsense facts by learning to encode a commonsense knowledge graph. These models learn to hypothesize commonsense facts through language generation. on tuples from a commonsense knowledge graph automatic knowledge graph construction, producing knowledge on-demand for any head entity that can be learn expressed through language.

For example, COMET (Bosselut et al., 2019) is a commonsense transformer knowledge model that adapts pretrained language models by training them on tuples from. Bosselut et al. (2019) show that a COMET model trained on ConceptNet and ATOMIC is able to be express knowledge more precisely than naive knowledge models trained only on a knowledge graph.

2.3 Few-Shot Learning vs. Augmentation

Recently, the term "few-shot" has taken two meanings: the classical definition of training on limited examples, and a new definition linked to few-shot context augmentation (Brown et al., 2020). In few-shot learning, language models are trained directly on a limited set of examples from the knowledge graph. In few-shot augmentation (or adaptation), models are given examples as a prepended augmentation of their context and can attend to these examples to recognize the structure of the task.

While the results of few-shot adaptation have been impressive, classical few-shot learning has key advantages. First, larger example sets will exceed maximum context windows associated with transformer language models. This bottleneck limits the number of examples that can be used in few-shot adaptation methods, potentially lowering performance when additional training examples could be used for adaptation (§4.1). In addition, classical few-shot learning methods can leverage information learned from training on other relations. We find this gives a performance boost in few-shot settings (§5.2).

3 COMET$^{20}_{21}$ Pipeline Overview

3.1 Definitions

Our goal is to train knowledge models using a limited number of examples per relation. A relation describes how two or more natural language statements combine logically. For example, we define the `ObjectUse` relation type with the prompt "h is used for t". We use the relation taxonomy from ATOMIC$^{20}_{20}$, which describes 23 relations across social and physical commonsense scenarios. Each training example uses the {*head, relation*} as input, and the task is to generate a valid *tail* as output. Instead of using relation embeddings (Bosselut et al., 2019), we automatically format the input tuples into natural language using prompts. Table 3 shows examples of such prompt formatting, and we perform ablation studies on the use of these prompts during training (§4.3). We define n as a per-relation example count, e.g. $n = 5 \implies 23 * 5$ examples.

3.2 Data Generation

We source training tuples from the ATOMIC$^{20}_{20}$ training set (Hwang et al., 2021). When constructing few-shot training sets, we set a target number n of examples to randomly sample from the knowledge graph for each relation. For each relation, n examples are selected by first randomly selecting a head entity, and then selecting one tail entity (connected through that relation) for each selected head entity. This procedure ensures a diversity of head entities for training as each head can have a multiple connected tail entities in the graph. The ATOMIC$^{20}_{20}$ knowledge graph is adversarially split into training, development, and test subsets such that no head entities in one set appear in any other. This property allows models to be evaluated on their capacity to generalize new commonsense inferences, rather than memorize them.

3.3 Training

Once a training set of examples is produced, the model is trained on this subset to minimize the negative log-likelihood of the tokens of the tail entity for each tuple. We use a constant learning rate schedule of 0.001, a mini-batch size of 4, and train the model for 3 epochs. For training, we use $v3 - 8$ Google Cloud TPUs on an $n1$ CPU VM machine. Our backbone is T5-11B (Raffel et al., 2019), using the pretrained weights from $v1.0$. We checkpoint the model after each epoch and select the best one based on training loss (using the development set for picking the early stopping point would violate the few-shot setting).

3.4 Evaluation

We evaluate the final models in our experimental settings using human and automatic evaluations. For the human evaluation (Accept % in Table 1), we use the human evaluation procedure described

Methodology	Model	BLEU-2	METEOR	ROUGE-L	CIDEr	Accept %	Cost
zero-shot	GPT-2 XL	2.8	8.2	9.8	4.7	36.6	$0
few-shot ($n = 5$)	GPT-2 XL	5.7	10.2	13.8	6.6	38.3	$0.50
	GPT-3	15.3	18.2	25.5	17.5	73.0	
	COMET$^{20}_{21}$ (T5)	21.9	19.5	25.7	19.2	78.6	
large-scale crowdsourcing	COMET (GPT-2 XL)	24.8	29.2	48.5	65.3	72.5	$4,347
	COMET (BART)	28.6	33.0	**49.5**	**65.8**	84.5	
	COMET$^{20}_{21}$ (T5)	**28.6**	**33.5**	47.1	59.7	**84.6**	

Table 1: Comparison between various methods of training knowledge models. Large pretrained models perform few-shot ($n = 5$) knowledge completion well in both text-to-text (COMET$^{20}_{21}$ and COMET) and generative (GPT-3) settings, suggesting that the majority of the gains in knowledge modeling comes from the first few examples. Cost considers a per-relation cost of producing the respective models, considering collection cost ($0.10 per tuple).

Relation r	Prompt
ObjectUse	h is used for t
AtLocation	You are likely to find a h in a t
xIntent	Because of h, PersonX wanted t
xWant	After h, PersonX would want t.
xAttr	h is seen as t
isAfter	Something that happens after h is t
oWant	As a result of h, others would want t

Table 2: Examples of templates used for prompting language models. We find that use of natural language prompts significantly helps in few-shot settings, with the hypothesis of being due to ease of transfer from the pretraining objective.

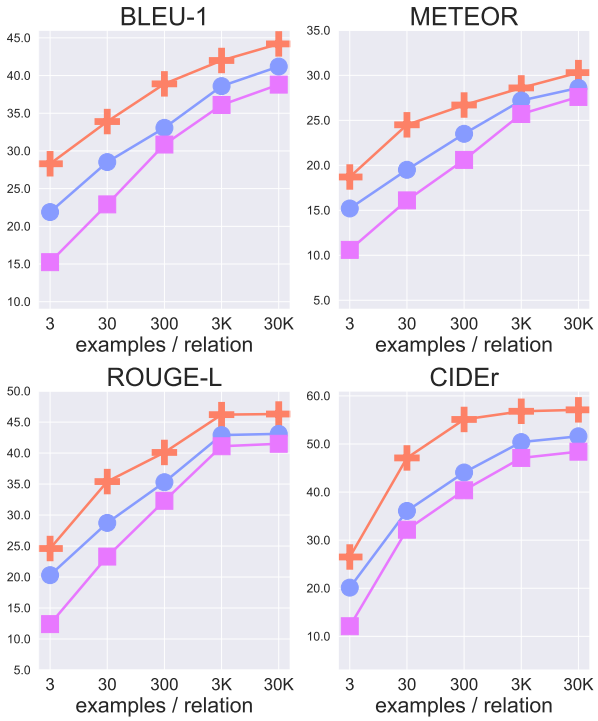

Figure 2: Effect of model size for commmonsense knowledge modeling. Small (60M) ( ), Large (770M) ( ), 11B ( ). The difference between model sizes is greatest in few-shot settings, where performance is separated by around 13 BLEU-1 points.

in Hwang et al. (2021). We ask annotators to label the quality of the {*given head, given relation, generated tail*} tuple using a 4-point Likert scale. Our scale corresponds to the following assessments of plausibility about the tuple: {2 - *always/often true,* 1 - *sometimes/likely true,* −1 - *false/untrue,* −2 - *nonsensical*}. We collect 3 annotations per relation and use the majority label as the score, with Fleiss's $\kappa = 0.46$ for the overall acceptability judgement. Positive scores are considered as accepted tuples. Due to the cost of human annotation, and the number of experiments pursued, we use automated metrics, such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), ROUGE-L (Lin, 2004), CIDEr (Vedantam et al., 2015)) to report performance in most experimental settings.

4 How do knowledge models learn?

We survey how commonsense knowledge models (e.g. via the COMET$^{20}_{21}$ pipeline) should be trained in a few-shot setting. First, we explore the capability of systems in eliciting knowledge models from pretrained architectures. The following questions explore best practices for few-shot modeling. We experiment with different model sizes across varying numbers of examples. Then, we investigate the importance of natural language prompts over symbolic representations.

4.1 Can commonsense knowledge models be trained in few-shots?

We find that the knowledge models trained via supervision in few-shots can produce high-

Head	Relation	Generated Tail (COMET^{20}_{21})	Generated Tail (GPT-3)
nail	AtLocation	construction site ✓	wall ✓
state highway	ObjectUse	statewide transportation ✓	drive a car ✗
video camera	ObjectUse	video recording ✓	record ✓
PersonX takes it to the vet	HinderedBy	PersonX doesn't have money to pay the vet ✓	PersonX gets a new pet ✓
PersonX makes PersonY very sick	HinderedBy	PersonX isn't close to PersonY. ✓	PersonY is not sick ✓
PersonX finds another job	isAfter	PersonX leaves job ✓	PersonX gets a new job ✗
PersonX falls ill	isBefore	PersonX feels sick ✓	to be happy with personx ✗
PersonX gets a call for an interview	xAttr	qualified ✓	hopeful ✓
PersonX wants to learn how to swim	xAttr	PersonX isn't confident in the water ✓	to be able to swim ✗
PersonX falls ill	xEffect	is hospitalized ✓	they are under the weather ✓
PersonX sprays by a skunk	xEffect	PersonX will be sick ✓	their smell ✓
PersonX gets milk	xIntent	to drink milk ✓	to drink milk ✓
PersonX works really hard	xIntent	to be appreciated ✓	to be rewarded ✓
PersonX misses class	xNeed	to have a valid excuse ✓	to be in class ✗
PersonX pumps PersonX's gas	xNeed	to have a gas pump ✓	to be safe ✗
PersonX notices a strange smell	xWant	to investigate the smell ✓	to get rid of it ✓
PersonX throws a frisbee	xWant	to catch the frisbee ✓	to be in the air ✗
PersonX wants to learn how to sing	xWant	to take voice lessons ✓	to learn how to sing ✗

Table 3: Comparison between few-shot ($n = 5$) generations produced by COMET^{20}_{21} (T5) and GPT-3. We find that COMET^{20}_{21} is able to produce surprisingly diverse and novel tail entities despite being supervised on only a few examples.

quality tuples in few-shot settings (Table 1).

Motivation The bulk of work on commonsense knowledge modeling has been done in fully supervised settings (Bosselut et al., 2019; Hwang et al., 2021), where knowledge models are trained on knowledge graphs of significant scale (hundreds of thousands of examples). The scale makes training knowledge models on new relations expensive (Table 1). However, for knowledge models to achieve broad applicability, they must be able to represent new relations *on-demand*, being quickly adapted to representing new conditions from seeing only a few examples.

Pretrained language models have shown aptitude for commonsense knowledge in zero-shot settings (Feldman et al., 2019). Similar to our few-shot training method, these methods use prompts to elicit knowledge from pretrained models. However, these zero-shot methods have drawbacks: the quality dips significantly, and the generations are fickle to the syntax of the prompt because the prompt is out of the distribution learned during pretraining (Shin et al., 2020).

Experiments In generative settings (GPT-2, GPT-3), we append the example templates to the beginning of the input sequence, modeling the generation task as a "complete-the-tuple" task. We prepend the templates mentioned to the beginning of the generated sequence. We split after a new line character and use the text before the next line character as our generated tail. We train COMET^{20}_{21}models as described in §3. In few-shot settings, we set the number of examples per relation to 5. For the "large-scale croudsourcing" setting, we train on all 1 million tuples in ATOMIC2020. To support our method in differing the usage of prompts, we investigate the importance of prompts in supervised settings in §4.3. For COMET (GPT2-XL) (Radford et al., 2019) and COMET (BART) (Lewis et al., 2020), we use scores from Hwang et al. (2021), which uses relation embeddings (Bosselut et al., 2019) rather than prompts.

Findings We find that both generative (GPT-3) and COMET^{20}_{21} (supervised) settings are able to produce high quality tuples, with both GPT-3 and COMET^{20}_{21} beating the fully-supervised GPT-2 XL setting with only 5 tuples. In the zero-shot setting, we find that the model performs worse that GPT-2 in the few-shot setting. We note that we are re-

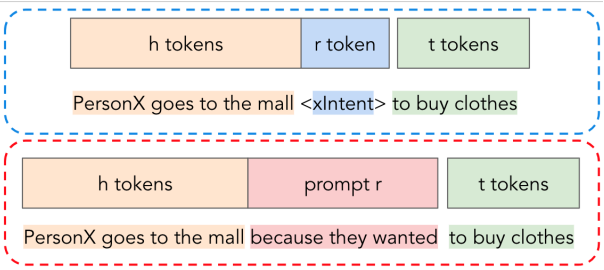

Figure 3: Illustration of knowledge model priming (Figure 4). When using priming, the knowledge model is never given the relation directly – rather, the input is only natural language.

stricted by uses for GPT-3 and thus cannot evaluate GPT-3 in the zero-shot setting until public release. The table is a culmination of the results in the rest of the paper, and furthermore, we will explore the best way to construct few-shot knowledge models.

In addition to automated metrics, we show examples of generations comparing the fully supervised and few-shot settings. Table 3 shows a set of generations across different relations types. We note that, in supervised settings, we are able to train on a few examples without significantly overfitting to those examples, e.g. only generating tuples that came from the three example relations. This allows us to generate tuples that heavily depend on the head, which is expected, and generate high-quality commonsense tuples in few-shot settings.

4.2 How does model size affect knowledge model learning?

We find that large model size is important in few-shot settings. Counterintitively, larger sizes models have a more agile data efficiency curve (Figure 2).

Motivation Recently, interest in training few-shot models has resurged, in part due to the popularity of GPT-3 (Brown et al., 2020). Brown et al. (2020) conducts a study exploring model sizes similar to ours, however, their exploration focuses on generative models (whereas our model is supervised via a sequence-to-sequence loss). Zhang et al. (2020b) and Mosbach et al. (2020) highlight that BERT (Devlin et al., 2019) few-shot fine-tuning is viable under proper training circumstances.

Method For each model size, we run expiremements varying model sizes as propsed in the COMET$^{20}_{21}$ pipeline. For model sizes, we test the smallest COMET$^{20}_{21}$ (COMET$^{20}_{21}$ T5-Small)

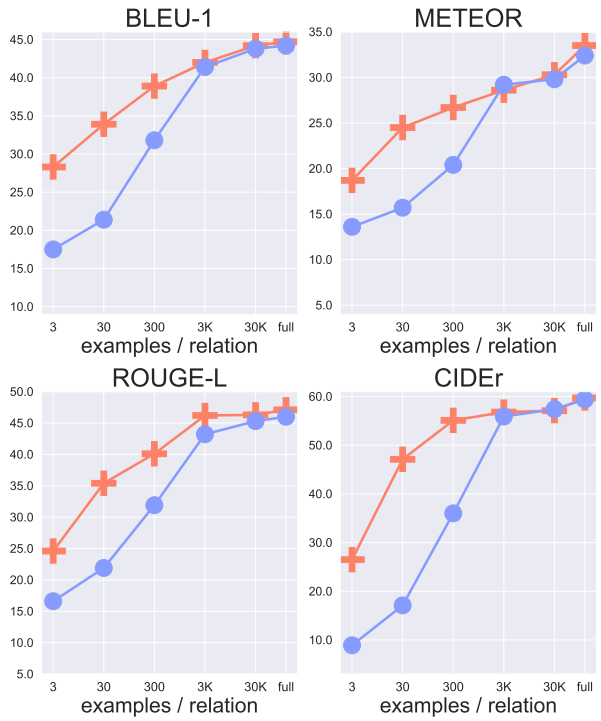

Figure 4: Comparison of training using natural language prompting (✚) versus previous methods utilizing a relation embedding (●). We show that priming methods improves the data efficiency curve of knowledge model training.

as the backbone of the knowledge model, with 60M parameters. We also test the large model (COMET$^{20}_{21}$ T5-Large) and the full (11B) pipeline. We keep hyper-parameters consistent between model sizes. Each model is trained with their respective $v1.0$ weights, provided by T5.

Findings Figure 2 shows the difference between model sizes. We find that the difference between model sizes is largest in few-shot settings, and gradually decreases in more supervised settings. Larger language models have a more agile data learning curve. Contrary to intuition, the data efficiency curve for small language models is consistently greater than that of large language models, despite the larger language models having more parameters to tune.

4.3 How do prompts affect knowledge model learning?

We find that the use of natural language prompts makes eases learning for commonsense language models in low-data settings (Figure 4).

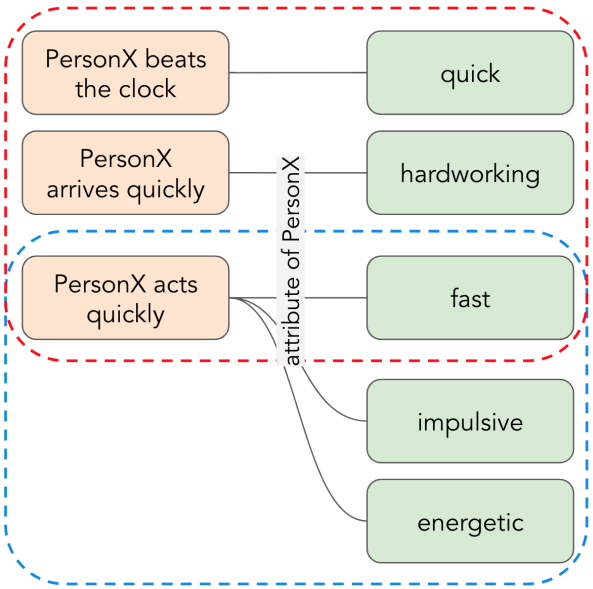

Figure 5: Illustration of a knowledge graph with $n = 3$ tuples for both the single tail (●) and diverse tail (✚) settings.

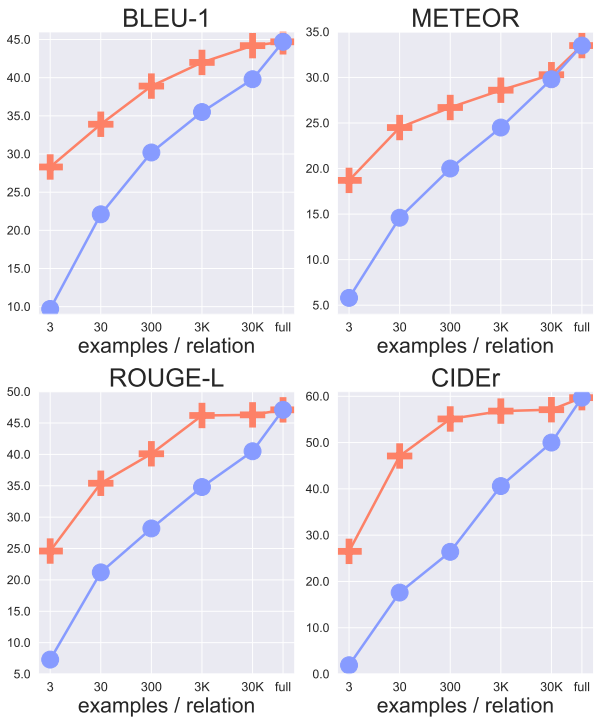

Figure 6: Comparison of training on examples with a greater diversity of heads (✚) and examples with a greater diversity of tails per head (●). Our results suggest that, in a few-shot setting, head variety is more important for training knowledge models than tail variety.

Motivation Contemporaneous work has shown that prompts can help models elicit knowledge from pre-trained representations (Shin et al., 2020; Feldman et al., 2019). However, in these zero-shot settings, the knowledge elicited is fickle to changes in prompts. Our exploration suggests few-shot settings as an alternative, which is less sensitive to minor syntax artifacts.

Method We study the use of prompts in different data settings. In summary, we explore two different settings for modeling knowledge models (Figure 3).

- Symbolic. Mimicking COMET (Bosselut et al., 2019), we initialize a symbolic embedding which is then attached to the head entity. We keep the rest of the pipeline the same as COMET^{20}_{21} for fair comparison.
- Linguistic. In few-shot settings, it may be easier for models to perform commonsense knowledge completion if the task is more similar to the pretraining objective. Our method follows the previously defined COMET^{20}_{21} procedure.

Findings Figure 4 shows our results. We find that prompts help improve the data efficiency in learning commonsense knowledge relations across all example counts. Prompts are especially important in low-data settings (where the empirical gap is greater). In addition, we show that after around 3000 heads per relation, the performance between and is similar, suggesting that it takes around 3000 examples per relation to learn the relation embedding in . We find the benefit is significant without the need for an additional coherency ranking layer (Feldman et al., 2019) or pulling prompts from the distribution of the model (Shin et al., 2020).

5 How should we annotate new knowledge relations?

We explore how to best construct few-shot knowledge graphs. To begin, we compare the importance of knowledge graph example breadth and example depth. Then, we study the benefit of additional training on a large-scale commonsense knowledge graph.

5.1 Heads or Tails: Example Breadth vs. Example Depth

We find that it is important to build commonsense KGs with significant head diversity: example breadth is a strong collator with knowledge model performance (Figure 6).

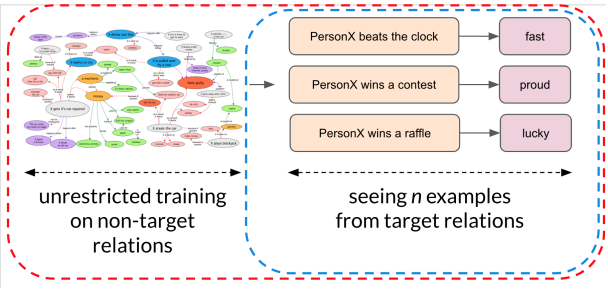

Figure 7: Illustration of training on additional relations (✚), then performing additional training on n test examples. The "no additional training" example in Fig. 8 trains on the right n test examples only (●), whereas the "additional training" model sees the majority of the ATOMIC$^{20}_{20}$ training set (17 relations), sans the test relations.

Method For this setting, we keep the number of examples constant and change if we allow multiple tails or not. To get a large diversity of heads, we allow only one tail as described in COMET$^{20}_{21}$. To collect a large diversity of tails, we select only from tails that have at least 5 tails in ATOMIC$^{20}_{20}$, and collect all tails from that head (5).

Findings Figure 6 shows our results. In summary, there is a large empirical gain (especially in few-shot settings) when increasing example breadth rather than depth. This suggests that for COMET$^{20}_{21}$, it is best to focus on collecting a wide variety of unique heads, rather than collecting many examples per head.

5.2 How many examples do commonsense knowledge models need to learn a *new* relation?

In zero and 3-shot scenarios, training commonsense knowledge models on ATOMIC$^{20}_{20}$ gives a boost in generation quality; it takes around 30 examples to learn a new relation (Figure 8).

Method To conduct the exploration, we take define a subset of 6 relations to evaluate on (2 from physical, 2 from event-centered, and 2 from social categories in ATOMIC2020). Then, we pretrain on ATOMIC2020, leaving out relations from these 6 categories. We then perform sequential training on these 6 relations according to the number of examples defined. We stop at 300, since larger data sizes require significant efforts in crowdsourcing. We compare these results to a baseline trained on only the 6 relations.

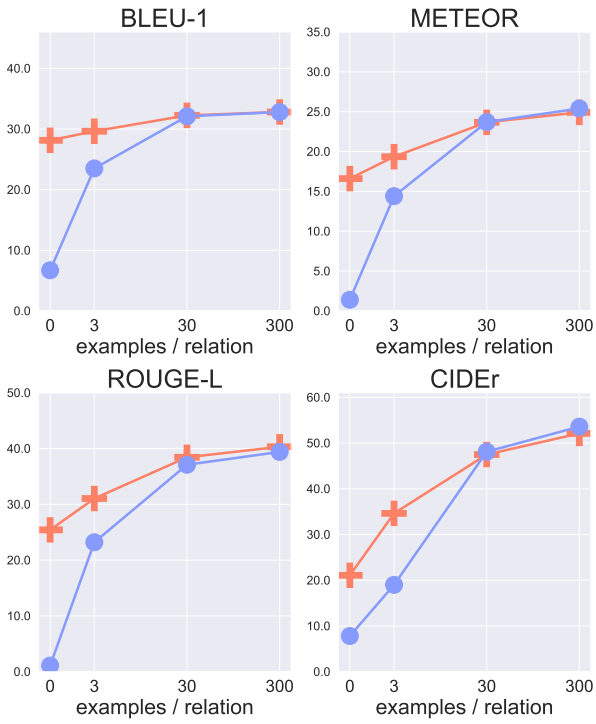

Figure 8: Comparison between models training commonsense knowledge models only on 6 selected target relations (✚) versus additional training on all examples from the ATOMIC$^{20}_{20}$ training set and then fine-tuning on the 6 selected target relations (●).

Findings We find that the zero-shot scores outperform the few-shot ($n = 3$) setting for knowledge modeling, highlighting the utility of using an additional knowledge model for pretraining. We find that it takes about 30 examples for knowledge models to learn a new relation, which is especially relevant for relations that are significantly out of the domain of a current large knowledge graph (such as ATOMIC$^{20}_{20}$), since it may not transfer as well to the new setting.

6 Related Work

Commonsense Knowledge Graphs Cyc (Lenat, 1995) is the first general knowledge base of commonsense rules and assertions. ConceptNet (Speer et al., 2017) and ATOMIC (Sap et al., 2019) model physical and social commonsense, repsectively. TransOMCS (Zhang et al., 2020a) uses retrieval-based methods to automatically construct the KG. SenticNet stores the semantics and sentics of commonsense entities.

Commonsense Knowledge Models COMET (Bosselut et al., 2019) introduces commonsense knowledge models as a method for automatic com-

momsense knowledge base construction. Bosselut et al. (2020) extends beyond static knowledge graphs by generative contextually-relevant knowledge structures on demand. VisualCOMET (Park et al., 2020) models commonsense knowledge multimodally. EMU (Da et al., 2020) produces multimodal commonsense related to social scenes. Path knowledge models (Wang et al., 2020b) models connections between head and tail entities.

Commonsense Knowledge Base Completion Previous work has shown model capability in commonsense knowledge base completion (Li et al., 2016; Saito et al., 2018). Malaviya et al. (2020) shows benefit from learning from local graph structure, and Wang et al. (2020a) shows adding edges between semantically related entities improves generalization ability.

7 Conclusion

We perform a study on few-shot knowledge models. We find that the COMET$^{20}_{21}$ pipeline helps commonsense knowledge models train in few-shot settings. We do a study on a variety of experimental settings, noting that: prompts are important for transfer between pretraining objective, head diversity in small knowledge graphs is important, and larger language models learn easier.

References

Prithviraj Ammanabrolu, W. Cheung, William Broniec, and M. Riedl. 2020. Automated storytelling via causal, commonsense plot ordering. *ArXiv*, abs/2009.00829.

S. Banerjee and A. Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *IEEvaluation@ACL*.

Antoine Bosselut, Ronan Le Bras, and Yejin Choi. 2020. Dynamic neuro-symbolic 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, A. Çelikyilmaz, and Yejin Choi. 2019. Comet: Commonsense transformers for automatic knowledge graph construction. In *ACL*.

T. Brown, B. Mann, Nick Ryder, Melanie Subbiah, J. Kaplan, P. Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, G. Krüger, Tom Henighan, R. Child, Aditya Ramesh, D. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, E. Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, J. Clark, Christopher Berner, Sam McCandlish, A. Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *ArXiv*, abs/2005.14165.

Tuhin Chakrabarty, Debanjan Ghosh, Smaranda Muresan, and Nanyun Peng. 2020. R^3: Reverse, retrieve, and rank for sarcasm generation with commonsense knowledge. In *ACL*.

P. Clark, Oyvind Tafjord, and Kyle Richardson. 2020. Transformers as soft reasoners over language. *ArXiv*, abs/2002.05867.

Jeff Da, M. Forbes, Rowan Zellers, Anthony Zheng, Jena D. Hwang, Antoine Bosselut, and Yejin Choi. 2020. Edited media understanding: Reasoning about implications of manipulated images. *ArXiv*, abs/2012.04726.

J. Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

J. Feldman, Joe Davison, and Alexander M. Rush. 2019. Commonsense knowledge mining from pretrained models. *ArXiv*, abs/1909.00505.

Jonathan Gordon and Benjamin Van Durme. 2013. Reporting bias and knowledge acquisition. In *Proceedings of the 2013 workshop on Automated knowledge base construction*, pages 25–30. ACM.

H Paul Grice, Peter Cole, Jerry Morgan, et al. 1975. Logic and conversation. *1975*, pages 41–58.

Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2021. Comet-atomic 2020: On symbolic and neural commonsense knowledge graphs. *AAAI*.

William R. Kearns, Neha Kaura, Myra Divina, Cuong Viet Vo, Dong Si, Teresa M. Ward, and Weichao Yuwen. 2020. A wizard-of-oz interface and persona-based methodology for collecting health counseling dialog. *Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems*.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, P. Clark, and Hannaneh Hajishirzi. 2020. Unifiedqa: Crossing format boundaries with a single qa system. In *EMNLP*.

Douglas B Lenat. 1995. Cyc: A large-scale investment in knowledge infrastructure. *Communications of the ACM*.

M. Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, A. Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *ArXiv*, abs/1910.13461.

Xiang Li, Aynaz Taheri, Lifu Tu, and Kevin Gimpel. 2016. Commonsense knowledge base completion. In *ACL*, volume 1, pages 1445–1455.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *ACL 2004*.

Y. Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, M. Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *ArXiv*, abs/1907.11692.

Bodhisattwa Prasad Majumder, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Julian McAuley. 2020. Like hiking? you probably enjoy nature: Persona-grounded dialog with commonsense expansions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9194–9206, Online. Association for Computational Linguistics.

Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. 2020. Commonsense knowledge base completion with structural and semantic context.

Marius Mosbach, Maksym Andriushchenko, and D. Klakow. 2020. On the stability of fine-tuning bert: Misconceptions, explanations, and strong baselines. *ArXiv*, abs/2006.04884.

Kishore Papineni, S. Roukos, T. Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *ACL*.

J. Park, Chandra Bhagavatula, R. Mottaghi, Ali Farhadi, and Yejin Choi. 2020. Visualcomet: Reasoning about the dynamic context of a still image. In *ECCV*.

F. Petroni, Tim Rocktäschel, Patrick Lewis, A. Bakhtin, Y. Wu, Alexander H. Miller, and S. Riedel. 2019. Language models as knowledge bases? *ArXiv*, abs/1909.01066.

A. Radford. 2018. Improving language understanding by generative pre-training.

A. Radford, Jeffrey Wu, R. 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, W. Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *ArXiv*, abs/1910.10683.

Kyle Richardson and Ashish Sabharwal. 2019. What does my qa model know? devising controlled probes using expert knowledge. *Transactions of the Association for Computational Linguistics*, 8:572–588.

Itsumi Saito, Kyosuke Nishida, Hisako Asano, and Junji Tomita. 2018. Commonsense knowledge base completion and generation. In *Proceedings of the 22nd Conference on Computational Natural Language Learning*, pages 141–150.

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. *ArXiv*, abs/1811.00146.

Taylor Shin, Yasaman Razeghi, IV RobertLLogan, Eric Wallace, and S. Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *ArXiv*, abs/2010.15980.

Robyn Speer, J. Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. *ArXiv*, abs/1612.03975.

Alon Talmor, Yanai Elazar, Y. Goldberg, and Jonathan Berant. 2019. olmpics-on what language model pretraining captures. *Transactions of the Association for Computational Linguistics*, 8:743–758.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*.

Ramakrishna Vedantam, C. L. Zitnick, and D. Parikh. 2015. Cider: Consensus-based image description evaluation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4566–4575.

Bin Wang, Guangtao Wang, J. Huang, Jiaxuan You, J. Leskovec, and C. J. Kuo. 2020a. Inductive learning on commonsense knowledge graph completion. *ArXiv*, abs/2009.09263.

Peifeng Wang, Nanyun Peng, Filip Ilievski, Pedro Szekely, and Xiang Ren. 2020b. 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.

Hongming Zhang, Daniel Khashabi, Y. Song, and D. Roth. 2020a. Transomcs: From linguistic graphs to commonsense knowledge. *ArXiv*, abs/2005.00206.

Tianyi Zhang, Felix Wu, Arzoo Katiyar, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Revisiting few-sample bert fine-tuning. *ArXiv*, abs/2006.05987.

8 Appendix

8.1 Accuracy in zero-shot setting

	BLEU-1	BLEU-2
T5 - Zero-shot	0.067	0.017

While little work has been doing exploring few-shot knowledge completion, recent works have investigated performance of zero-shot knowledge graphs, such as LAMA and coherency ranking (Petroni et al., 2019; Feldman et al., 2019). Thus, we investigate the ability of T5 to complete commonsense knowledge in a zero-shot setting. Different from the few-shot and supervised approaches, we do not use teacher forcing, but rather use sentinal tokens to leverage the masking objective of the language model pretraining. In addition to mask prediction, we try a couple variants. Since the mask only predicts several tokens at a time, for relations with longer responses (e.g. ATOMIC relations), we allow the model to predict up to 7 mask tokens in secession, or until the model predicts an empty string for the mask. We suggest that this is still only a workaround, and masked models are poor predictors of longer length tails.