Retrieval Enhanced Model for Commonsense Generation

Han Wang[1]*, **Yang Liu**[2], **Chenguang Zhu**[2], **Linjun Shou**[3], **Ming Gong**[3], **Yichong Xu**[2], **Michael Zeng**[2]
[1]New York University
[2]Microsoft Cognitive Services Research Group
[3]STCA NLP Group, Microsoft, Beijing, China
hw2725@nyu.edu
{yaliu10,chezhu,lisho,migon,yicxu,nzeng}@microsoft.com

Abstract

Commonsense generation is a challenging task of generating a plausible sentence describing an everyday scenario using provided concepts. Its requirement of reasoning over commonsense knowledge and compositional generalization ability even puzzles strong pre-trained language generation models. We propose a novel framework using retrieval methods to enhance both the pre-training and fine-tuning for commonsense generation. We retrieve prototype sentence candidates by concept matching and use them as auxiliary input. For fine-tuning, we further boost its performance with a trainable sentence retriever. We demonstrate experimentally on the large-scale CommonGen benchmark that our approach achieves new state-of-the-art results.

1 Introduction

The understanding of commonsense knowledge in human language has been acknowledged as a critical component for artificial intelligence systems. In recent years, many new tasks and datasets are proposed to assess NLP model's ability of commonsense reasoning (Yu et al., 2020). SWAG (Zellers et al., 2018) is a task of inferring the upcoming event based on a partial description using commonsense. CommonsenseQA (Talmor et al., 2019) is a commonsense question answering dataset built from ConceptNet. Recently, Lin et al. (2020) propose CommonGen, a new challenge for evaluating model's ability of generative commonsense reasoning.

CommonGen requires the system to construct a plausible sentence based on several concepts related to an everyday scenario. Two examples for this task is shown in Table 1. The task is challenging because the system needs to organize provided concepts into the most plausible scenario, avoid violation of commonsense, and ensure the generated sentence is grammatically correct. Existing approaches fine-tune pre-trained encoder-decoder models for description construction with concatenated concepts as input.

Concept Set #1:
dog, frisbee, catch, throw
Gold Target Sentence:
A dog leaps to catch a thrown frisbee.
The dog catches the frisbee when the boy throws it.
A man throws away his dog 's favorite frisbee expecting him to catch it in the air.
Concept Set #2:
lake, shore, canoe
Gold Target Sentence:
Canoe on a shore of lake.
Canoe on shore with rainbow across the lake.
Several canoes parked in the grass on the shore of a lake.

Table 1: Two concept sets and their gold corresponding sentences from CommonGen dataset.

Fan et al. (2020) propose a retrieve-and-generation method for commonsense generation which uses a prototype candidate sentence as auxiliary input. However, their retriever is non-trainable and only works for the fine-tuning process. In this work, we extend this idea and propose a novel framework for commonsense generation by using retrieval method for enhancing both the pre-training and fine-tuning stages. Furthermore, we design a trainable prototype sentence retriever to further boost generation performance.

We conduct experiments on CommonGen (Lin et al., 2020) benchmark dataset. It contains 35,141 concept sets and 79,051 corresponding sentences. Each concept set is mapped to multiple corresponding sentences. Our approach achieves new state-of-the-art results on CommonGen on several metrics, including BLEU, CIDEr and SPICE.

*Work done during internship at Microsoft.

2 Method

We frame CommonGen challenge as a sequence-to-sequence task and adopt T5 (Raffel et al., 2020), a powerful pre-trained encoder-decoder model, as our base model. Fan et al. (2020) find concepts-related sentences in external corpora can benefit relational reasoning for CommonGen. We extend this idea by proposing retrieval-enhanced T5 (RE-T5) which equips original T5 with a trainable retriever for selecting prototype sentences based on given concepts. Meanwhile, referring to (Zhou et al., 2021), we design a pre-training task for CommonGen which continue to pre-train RE-T5 on pseudo concept sets extracted from external corpora. We also use a retriever in this pre-training stage.

Formally, given a concept set $X = \{x_1, x_2, \ldots, x_n\}$, where x_i represents the i-th concept and n is the number of concepts, our goal is to generate a natural language output of tokens $Y = \{y_1, y_2, \ldots, y_m\}$, which describes a common scenario in our daily life, using all given concepts in X.

2.1 Retrieval

Since external corpora have lots of scenario knowledge to describe the relationship between concepts (Fan et al., 2020), we retrieve sentences related to input concepts to help the model perform better commonsense reasoning. First, given an input concept set, we extract all sentences from external corpora that contain at least two concepts in the input $\boldsymbol{x}$ as candidate set $\mathcal{Z}$. Then, We design two retrieval models, *matching retriever* and *trainable retriever*, to further retrieve k prototype sentences $Z = \{z_1, z_2, \ldots, z_k\}, Z \subseteq \mathcal{Z}$ as auxiliary input context for RE-T5.

Matching Retriever The *matching retriever* first orders candidate sentences by the number of contained concepts. Then it simply samples k sentences starting from sentences that contained the most concepts as the auxiliary input.

Trainable Retriever In order to retrieve more useful sentences from the sentence candidate set, we design a *trainable retriever*, which predicts scores to rank these candidates, and then select top-k sentences as additional context. The scorer is built based on BERT (Devlin et al., 2019), a pre-trained language model usually used for language understanding. Given a concept set X and a candidate sentence z_i, our trainable retriever first concatenate them into a text input:

$$\texttt{[CLS]}\, X\, \texttt{[SEP]}\, z_i\, \texttt{[SEP]}$$

where `[CLS]` and `[SEP]` are special symbols in BERT.

We pass this into BERT, which generates an output vector for each input token. We take the output vector corresponding to `[CLS]` which is used as the aggregated representation of the input sequence (denoted $\boldsymbol{c}$) into a linear layer with sigmoid activation to obtain the binary classification output y_c.

$$y_c = \sigma(\boldsymbol{W}_c \boldsymbol{c} + b_c) \quad (1)$$

where $\boldsymbol{W}_c$ is a projection matrix and b_c is a bias.

To train this retriever, for each concept set in CommonGen training set, we use its paired sentence as a positive example and we randomly sample another sentence, also from the training set, as a negative example. Then, we adopt cross entropy loss for this binary classification. The top-k scored sentences with the highest scores will be selected as the auxiliary input Z.

We will describe how these two retrievers are used in CommonGen pre-training and fine-tuning stages.

2.2 Pre-training

To enhance model's ability of commonsense reasoning, we design a pre-training task for RE-T5 which is similar to original CommonGen task. In more details, given a sentence from external corpora, we first use spaCy (Honnibal et al., 2020) to tag the sentences with part-of-speech and extract *Verb*, *Noun* and *Proper Nouns* as pseudo concept phrases. We then only keep phrases in ConceptNet (Speer et al., 2017) and remove concept-sets that appear in CommonGen's testset. We use the original sentence as the target sentence, and constructs a pre-training task of using RE-T5 to generate this sentence given pseudo concepts.

Due to the extraction method for pseudo concepts, when retrieving prototype sentences, for each concept set in pre-training data, we have a large candidate set $\mathcal{Z}$ with an excessive number of candidate sentences. This leads to a long inference time for using the trainable retriever. Thus, due to speed consideration and also to introduce a degree of randomness into pre-training, we use the *matching retriever* to retrieve k sentences as auxiliary input Z.

After retrieval, RE-T5 takes the concatenation of input concepts and retrieved prototype sentences as input, and the original sentence as output.

Model	BLEU-4	CIDEr	**SPICE**	SPICE(v1.0)
GPT-2 (Radford et al., 2019)	26.833	12.187	23.567	25.90
BERT-Gen (Bao et al., 2020)	23.468	12.606	24.822	27.30
UniLM (Dong et al., 2019)	30.616	14.889	27.429	30.20
BART (Lewis et al., 2020)	31.827	13.976	27.995	30.60
T5-Base (Raffel et al., 2020)	18.546	9.399	19.871	22.00
T5-large (Raffel et al., 2020)	31.962	15.128	28.855	31.60
EKI-BART (Fan et al., 2020)	35.945	16.999	29.583	32.40
KG-BART (Liu et al., 2021)	33.867	16.927	29.634	32.70
CALM (Zhou et al., 2021)	-	-	-	33.00
RE-T5 (ours)	**40.863**	**17.663**	**31.079**	**34.30**

Table 2: Test results on CommonGen benchmark. All results are based on the latest human references. v1.0 indicates evaluation with old evaluation protocol.[1]

2.3 Fine-tuning

At fine-tuning stage, we use *trainable retriever* to score sentences from candidate set $\mathcal{Z}$ and select top k sentence as additional context Z. Similar to pre-training, RE-T5 takes the concatenation of input concepts and retrieved prototype sentences as input, and the original sentence as output.

3 Experiments

3.1 Experiments Settings

Dataset CommonGen is a benchmark dataset designed to diagnose whether a model has the ability of generative commonsense reasoning (Lin et al., 2020). This dataset contains 32,651/993/1,497 concept sets for training/development/test, and the numbers of corresponding sentences are 67,389/4,018/7,644. We use BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), METEOR (Banerjee and Lavie, 2005), CIDEr (Vedantam et al., 2015) and SPICE (Anderson et al., 2016) as evaluation metrics. Because SPICE correlates the most with human evaluation (Lin et al., 2020), we take SPICE as the primary metric.

External Corpora We use VATEX (Wang et al., 2019), Activity (Krishna et al., 2017), SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018) as external corpora. We sample 500k sentences from these corpora to constructing our pre-training dataset. Meanwhile, these datasets are also used as our sentence pool for the retrieval module. For both the pre-training and fine-tuning, all sentences that appear in the target are not used as retrieval sentences candidates.

[1]https://inklab.usc.edu/CommonGen/leaderboard.html

Baselines We compare RE-T5 with several baseline systems. GPT-2, BERT-Gen, UniLM, BART, and T5 are pre-trained language models tested in (Lin et al., 2020). They are all fine-tuned on CommonGen training set with concatenated concepts as input and description sentence as output. EKI-BART (Fan et al., 2020) is a retrieve-and-generate framework for CommonGen, where they use a simple retriever to enhance pre-trained BART (Lewis et al., 2020). KG-BART (Liu et al., 2021) augment BART with Knowledge Graph on both the encoder and decoder side and continue to pre-train BART with a masked concept token generation task. CALM (Zhou et al., 2021) designs several self-supervised strategies encouraging model to focus on concept-centric information.

Implementation Details We adopt the T5-base as the generation model and BERT-base as the trainable retriever in fine-tuning. We use the Huggingface Transformer (Wolf et al., 2020) for model implementation. For pre-training phase, we use the AdamW optimizer (Loshchilov and Hutter, 2019) with an initial learning rate of 2e-6, and the model is pre-trained for 3 epochs. For fine-tuning, the models are optimized using AdamW with an initial learning rate of 5e-5 and trained for 20 epochs. For the number of the retrieved sentences k, we experimentally choose 3. More implementation details are in Appendix.

3.2 Results

Table 2 shows results of different approaches on the CommonGen testset. RE-T5 outperforms all previous approaches by a large margin in all metrics and sets a new state of the art. RE-T5 combines the generation flexibility of pre-trained language model and the interpretability and modularity of

Concept Set: trailer shirt side sit road
T5: A man sits on the side of a trailer and a shirt.
Matching Retriever: (1)Two guys in red shirts are sitting on chairs, by the side of the road, behind that open trailer. (2)Two men, one wearing a straw cone hat, blue shirt, talking with a guy in a tan sunhat, red plaid shirt, both with baskets in front of them, sitting on the side of a dirt road. (3)An older guy with a tan shirt and hat sitting on the side of a road with bricks all around him and a small green bowl on the side. **RE-T5(matching retriever)**: a man in a tan shirt sits on the side of a road.
Trainable Retriever: (1)Two guys in red shirts are sitting on chairs, by the side of the road, behind that open trailer. (2)Teenagers in matching shirts stand at the side of the road holding trash bags. (3)A man in a white shirt and black pants standing at the side or the road. **RE-T5(trainable retriever)**: a man in a white shirt and black pants sits on the side of a trailer on the road.

Table 3: An example of sentences retrieved by different retrievers and sentences generated based on them.

Model	**SPICE**
T5	30.80
T5 + *MR*	33.60
T5 + *MR* + pretrain	33.90
RE-T5 (T5 + *TR* + pretrain)	**34.30**

Table 4: Ablation results on the test set of CommonGen. Note that *MR* denotes *Matching Retriever* and *TR* denotes *Trainable Retriever*.

a trainable retrieval-based approach. Unlike EKI-BART (Fan et al., 2020) and KG-BART (Liu et al., 2021), RE-T5 enjoys strong results without model architecture modification. It is worth noting that although T5-base baseline does not perform as well as BART (Lewis et al., 2020) baseline, our method still outperforms the two improved BART-based methods mentioned above. RE-T5 demonstrates that for state-of-the-art performance, neither model modification nor complex fusion of knowledge graphs is necessary, only a simple and effective trainable retriever is needed.

Ablation Study We conduct ablation experiments as shown in Table 4. First, we can see that RE-T5 model outperforms the backbone T5 model by a large margin in all metrics, with 3.5 improvement in the main metric SPICE. The second line of Table 4 shows that, although large-scale pre-trained language models have been shown to learn and store a substantial amount of the world knowledge implicitly from the massive text corpora (Petroni et al., 2019), the retrieved sentences from external corpora can still explicitly expose lots of scenario knowledge to describe the relationship between concepts. The third line indicates that further pre-training with data augmentation is helpful to improve the performance of the model. In addition, the last line demonstrates that a trainable scorer can capture more helpful knowledge for the model for commonsense generation.

Example Analysis Through the example in Table 3, we can observe that the baseline model T5 generates a sentence without concept *"road"*, and the juxtaposition between *"trailer"* and *"shirt"* in this sentence is not in line with common sense. For both *matching retriever* and *trainable retriever*, the retrieved sentences remind the model not to forget the concept *"road"*, in addition to providing the relationship between *shirt* and *person*. Since *matching retriever* randomly retrieves sentences based on the number of concepts they contain, it tends to retrieve longer sentences to contain as many concepts as possible, which may confuse the model and thus ignore some concepts, for example, the sentence generated by RE-T5 (matching retriever) in this example is missing the concept *"trailer"*. RE-T5 (trainable retriever) can solve the above problems and generate a sentence that is fluent and in the line with common sense.

4 Conclusions

In this paper, we empirically investigated RE-T5, which utilizes a trainable retriever to retrieve sentences from external corpora to enhance the generative commonsense reasoning capability of pre-trained language model, such as T5. The state-of-the-art result achieved by RE-T5 on CommonGen benchmark demonstrates that a simple yet effective trainable retriever can be a useful addition to

the pre-trained language model for commonsense generation.

References

Peter Anderson, Basura Fernando, Mark Johnson, and Stephen Gould. 2016. Spice: Semantic propositional image caption evaluation. In *Computer Vision – ECCV 2016*, pages 382–398, Cham. Springer International Publishing.

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.

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Jianfeng Gao, Songhao Piao, Ming Zhou, and Hsiao-Wuen Hon. 2020. UniLMv2: Pseudo-masked language models for unified language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 642–652. PMLR.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

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: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, volume 32, pages 13063–13075. Curran Associates, Inc.

Zhihao Fan, Yeyun Gong, Zhongyu Wei, Siyuan Wang, Yameng Huang, Jian Jiao, Xuanjing Huang, Nan Duan, and Ruofei Zhang. 2020. An enhanced knowledge injection model for commonsense generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2014–2025, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Ranjay Krishna, Kenji Hata, Frederic Ren, Li Fei-Fei, and Juan Carlos Niebles. 2017. Dense-captioning events in videos. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. 2020. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1823–1840, Online. Association for Computational Linguistics.

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.

Ye Liu, Yao Wan, Lifang He, Hao Peng, and Philip S. Yu. 2021. Kg-bart: Knowledge graph-augmented bart for generative commonsense reasoning.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

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.

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, Jeff 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. 2020. Exploring

the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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*, AAAI'17, page 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.

Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. 2019. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

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

Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. SWAG: A large-scale adversarial dataset for grounded commonsense inference. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 93–104, Brussels, Belgium. Association for Computational Linguistics.

Wangchunshu Zhou, Dong-Ho Lee, Ravi Kiran Selvam, Seyeon Lee, and Xiang Ren. 2021. Pre-training text-to-text transformers for concept-centric common sense. In *International Conference on Learning Representations*.

A Implementation Details

During pre-training, we use the AdamW optimizer (Loshchilov and Hutter, 2019) with weight decay 0.01, adam epsilon 1e-6, and a warmup fraction of 0.01. We use a batch size of 16, and gradient accumulation of 4 batches. For fine-tuning, the models are optimised using AdamW optimizer with initial learning rate 5e-5, batch size 64, gradient accumulation 3 and warmup fraction 0.01, and the model is trained for 20 epochs.

Meanwhile, the BERT-base scorer is optimised using AdamW optimizer with an initial learning rate 2e-5, batch size 64, and the model is trained for 3 epochs. For the number of the retrieved sentences k, we experimentally choose 3. All experiments are conducted using 4 V100 with 32 GB memory.