Fine-tuning and Prompt-learning on Commonsense Causal Reasoning

Machine Learning Project 2 - ML4Science

Yiyang Feng[§], Naisong Zhou[§], Yuheng Lu[§]
Project 2, CS433 Machine Learning, EPFL

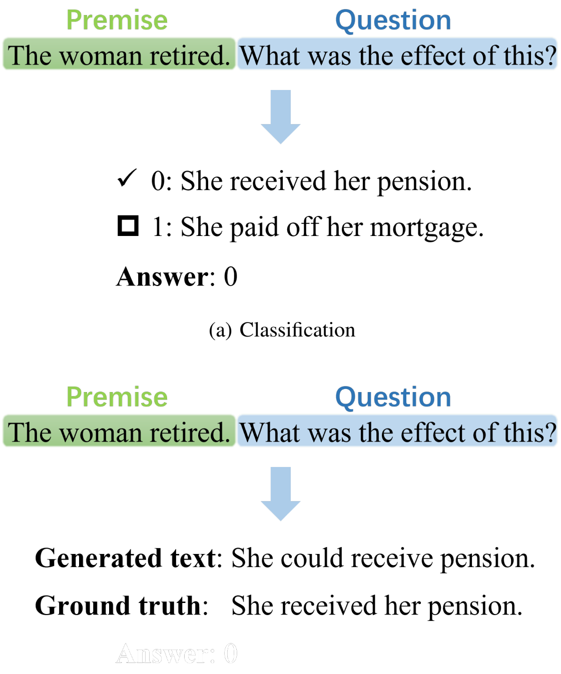
***Abstract*—Commonsense Causal Reasoning (CCR) aims to understand and reason about the cause-and-effect relationships in the world. The COPA dataset is widely used to evaluate the performance of systems in CCR tasks. In this paper, we define the COPA CCR task into two sub-tasks: the original classification task and the cause/effect generation task. We then implement fine-tuning models and the prompt learning model GPT-3 on these sub-tasks. Finally, we compare the performance between these models, and the results have shown that these models learn some commonsense causal relationship. The performance of GPT-3 with prompt learning is significantly better on both tasks. We analyze the superior performance of GPT-3 may be due to more of its large number of model parameters and massive pre-trained dataset than prompt learning itself. However, the ability of few-shot learning is still important for its efficiency in downstream adaptation.**

I. Introduction

Commonsense Causal Reasoning (CCR) is the ability to understand and reason about the cause and effect in the common world. Causal reasoning is the process of inferring the relationship between cause and effect, i.e., to make judgments about why things happen and to predict the consequences of actions, while commonsense involves basic knowledge about how the world works. CCR has become a popular area of research in NLP, as creating systems that can reason with commonsense knowledge is a major challenge in the field.

We divide the CCR task into two sub-tasks: text classification and generation. In the next classification task, which is already widely experimented with, the model receives a premise, a question field indicating cause or effect, and two choices. The model is then required to choose a reasonable answer. In the text generation task, which is even more challenging, we will no longer provide choices and let the model directly output a reasonable answer with the input of a premise and the question. Therefore, we remove the fake choice and the label, only leaving the correct one as the ground truth for the generation task. The two tasks are illustrated in Figure 1.

COPA[1], short for Choice Of Plausible Alternatives, is a widely used dataset in evaluating performance in CCR tasks. COPA provides 1000 questions in total, which are divided into a train set (400), a validation set (100), and a test set (500). Each question consists of a premise, a question, two candidate choices, and a label. The premise is a given statement, while the question on the premise is either one of two tasks: effect (forward causal reasoning) and cause (backward causal reasoning). The candidate choices include a correct choice that is more causally plausible under the question. The label records which choice is the correct answer (0 for choice 1 and 1 for choice 2). Labels are randomly distributed throughout the dataset, so a random guesser has an expected performance of 50% accuracy. Here are a few COPA examples in Table I.


Premise
Question
The woman retired.
What was the effect of this?
✓ 0: She received her pension.
□ 1: She paid off her mortgage.
Answer: 0
Premise
Question
The woman retired.
What was the effect of this?
Generated text: She could receive pension.
Ground truth: She received her pension.
Answer: 0

(a) Classification

(b) Generation

Figure 1. The illustration of the CCR classification (a) and generation (b) task.

We compare fine-tuning and prompt-learning models on the two CCR tasks. Fine-tuning is a common method for utilizing pre-trained language models to complete downstream tasks. It can be a useful technique for adapting a model to a new task or improving its performance on a specific task. However, one potential advantage is that it involves

[§] Equal contribution

example	element	value
Example1	premise	The woman retired.
	choice1	She received her pension.
	choice2	She paid off her mortgage.
	question	effect
	label	0
Example2	premise	My body cast a shadow over the grass.
	choice1	The sun was rising.
	choice2	The grass was cut.
	question	cause
	label	0

Table I
COPA EXAMPLES

adjusting and storing all the parameters of the language model, which is prohibitively expensive with the increase in model size. So another prevalent approach is proposed to solve this problem, known as prompt learning, represented by GPT-3[2]. The model is frozen in the prompt learning setting, and users just prepend a natural language instruction, some examples with answers, and a prompt masking the answer to the input. The model will automatically provide the completion.

Our work is summarized as follows:

1) We derive a new cause/effect generation task from the original COPA dataset and divide our CCR task into two sub-tasks.
2) We conduct experiments on the two CCR tasks on the COPA dataset using fine-tuning and prompt learning models.
3) We compare the performance between two sets of models and analyze the result.

II. Text Classification

We fine-tune several BERT-based models on COPA for our classification task. We also conduct prompt learning experiments on GPT-3 using OpenAI APIs. The detailed settings and results are as follows.

A. Settings

1) Models and Parameters: We fine-tuned the BERT[3], RoBERTa[4], XLM-RoBERTa[5], and ALBERT[6] models. All these models are modified from Transformer encoders. The ALBERT models include both the base and large versions, and other models are base models. We used the Hugging Face[7] Multiple Choice model, which takes in an input sentence and two candidates, and determines which candidate is better connected to the input sentence. For the COPA dataset, we designed the input sentence in the form of "premise + 'What was the cause/effect of this?'" and used choice1 and choice2 as the two candidates. We trained on 4 NVIDIA TITAN X cards, with 50 epochs, a batch size of 32, and a learning rate of 5×10^{-5}. Other parameters were set to the default values in the Hugging Face model.

In the prompt-learning process, we experiment with the text-davinci-003 model. The hyperparameters are shown in Table II:

key	value
Model	text-davinci-003
temperature	0.7
max tokens	256

Table II
HYPERPARAMETERS OF GPT-3 FOR CLASSIFICATION

Temperature ranges from 0 to 1 and denotes the sampling temperature to use. Higher values mean the model will take more risks, and 0 means the model will choose an answer with less uncertainty. Here we use a temperature of 0.7, which allows the model to behave more actively and risky to "guess" causal relations.

Max tokens is the maximum number of tokens to generate in the completion. To align, the token count of prompt plus *max tokens* should be less than the model's context length (for text-davinci-003, we chose it is 4096). Here we set *max tokens* as 256 to allow for longer completions if needed. As hints are given to the model, here we design the prompts as shown in Algorithm 1:

```
Input  : A validation datapoint $D$
Prompt: (Instruction) Identify the correct response
        from two sentences.
        Premise: D[premise]
        Choice1: D[choice1]
        Choice2: D[choice2]
        Question: cause or effect
Output: choice1 or choice2
```

Algorithm 1: Prompt paradigm for classification

2) Metrics: We compare the performance of these models on classification tasks with 4 metrics: accuracy, macro precision, macro recall, and macro F1-score.

B. Results and Analysis

Results are seen in Table III. We run each setting for five times and report the mean and standard error of the four metrics. We find that the accuracy of all models is higher than that of random guesses (50%), indicating these models learn some commonsense causal relationship. The performance of the BERT-base and ALBERT-large models is better, and the performance of these two models is similar. However, the performance of fine-tuning models and GPT-3 with prompt learning has a large degree of difference, and the accuracy of the latter model reaches more than 90%.

We believe that the outstanding performance of GPT-3 is due to its large number of model parameters. The text-davinci-003 model we used has 175B parameters. The maximum number of parameters of the fine-tuned model we chose is only 125M parameters in the RoBERTa and

XLM-RoBERTa models. Moreover, GPT-3 chose datasets from Common Crawl and WebText2, which are 45TB web text corpora that may already contain part of the text from the COPA dataset, resulting in dataset leakage. Therefore, the good performance of GPT-3 trained with prompt learning is more likely due to the size of its model and pre-trained dataset rather than prompt learning itself.

	Accuracy (%)	Precision (%)	Recall (%)	F1-score (%)
Random	49.8±4.0	49.4±3.9	49.4±3.9	49.3±3.9
BERT -base	74.6±1.2	74.3±1.2	74.4±1.3	74.4±1.2
RoBERTa -base	68.4±1.2	68.4±1.3	68.5±1.3	68.3±1.6
XLM-RoBERTa -base	64.2±1.6	66.1±2.3	65.4±1.1	64.1±1.5
ALBERT -base-v2	69.6±0.8	69.6±1.0	69.8±1.0	69.5±0.9
ALBERT -large-v2	74.6±1.2	74.4±1.2	74.3±1.2	74.3±1.2
GPT-3 -175B	**92.0±0.6**	**92.4±0.7**	**91.6±0.6**	**91.8±0.6**

Table III
COMPARISON BETWEEN FINE-TUNING AND PROMPT LEARNING ON THE CCR CLASSIFICATION TASK

III. TEXT GENERATION

Text generation, which is another aspect of NLP, can also be implemented via prompt learning through both fine-tuning and prompt learning. In this part, we still use COPA as the dataset. First, we design an appropriate prompt with the training set in COPA and then train them through fine-tuning and prompt learning. After that, we use the COPA validation set and calculate the relevant semantic metrics between the correct answer and what the model generates to test the model's performance.

A. Settings

1) Models and Parameters: We first fine-tuned a series of BERT models to generate cause or effect. We selected BERT, RoBERTa, XLM-RoBERTa, and BART[8] models for our experiment. All models are base models. The three models, BERT, RoBERTa, and XLM-RoBERTa are all variants of Transformer encoders. Therefore we only use the causal language model for fine-tuning instead of the encoder-decoder structure. A causal language model is a type of machine learning model that is used to predict the next word in a sequence, given the context of the previous words. The key characteristic of a causal language model is that it considers the order in which the words appear and uses this information to make more accurate predictions, so the labels for training are the token IDs in the sentence itself.

During training, we put together the premise sentence, the question sentence "What was the cause/effect of this?", and the correct choice sentence. In the inference time, we only give a premise sentence concatenated with a question sentence and let the model generate the following tokens. Then we decode these tokens to obtain the text of cause or effect.

BART models use a standard sequence-to-sequence architecture with a bidirectional encoder and an auto-regressive (left-to-right) decoder. It also includes a de-noising task that randomly shuffles the sentences and an algorithm to reconstruct the texts. Therefore, we use the Hugging Face conditional generation model for our generation task. We send the premise sentence and question sentence to the encoder and generate the cause or effect from the decoder. Then the model computes the loss by comparing the token ids between the correct choice and the prediction and updates its parameters. After that, we only input the sequence into the model and generate the cause or effect without giving the ground truth.

We use the same parameters as the classification task for fine-tuning, and we limit the number of max new tokens generated by the model to 100 at the time of inference.

In the prompt-learning process, we test with the text-davinci-003 model. The hyper-parameters are shown in Table IV:

key	value
Model	text-davinci-003
temperature	0.7
max tokens	100

Table IV
HYPERPARAMETERS OF GPT-3 FOR GENERATION

The prompt design are shown in Algorithm 2:

```
Input  : A validation datapoint $D$
Prompt: (Instruction) Answer the Question of
         Premise.
         Then there are 20
Premise-Question-Answer examples.
         Premise: D[premise]
         Question: What is the cause or effect of
         Premise?
         Answer:
Output: Text Generated
```

Algorithm 2: Prompt paradigm for generation

2) Metrics: In the generation task, we mainly use the four different metrics, i.e., BLEU, ROUGE-L, METEOR, and CIDEr, to evaluate the performance of the models. In the BLEU part, we calculate BLEU scores corresponding to 4 different n-grams (n = 1, 2, 3, 4).

BLEU score is used in this experiment to evaluate the similarity of the text generated by our model. Our model is implemented in the COPA validation set.

ROUGE is a set of metrics and a software package used in NLP. The metrics compare an automatically generated text against a reference or a set of references.

Meteor evaluates a generated text by computing a score based on explicit word-to-word matches between the generated text and a given reference.

B. *Results and Analysis*

All the metrics are shown in Table V. We can see that the BART model has a large gain over other fine-tuning models. The result indicates that the encoder-decoder-based model like BART is more advantageous for the generation task than standard causal language modeling. In addition, all the fine-tuning models fall short of the prompt learning model GPT-3. As with the section II-B, we believe that the performance of GPT-3 benefits more from its vast model size and massive training data than from the prompt learning approach itself.

However, the ability of language models to be few-shot learners is still essential because it allows them to generalize well to the new text they have not seen before. Prompt learning enables the models to be used in a wide range of NLP tasks, such as machine translation, text summarization, and question-answering.

	BERT -base -uncased	RoBERTa -base	XLM -RoBERTa -base	BART -base	GPT-3 -175B
BLEU-1	2.0%	26.7%	0	31.2%	**39.2%**
BLEU-2	0.5%	5.9%	0	11.8%	**23.3%**
BLEU-3	0	0	0	6.2%	**14.3%**
BLEU-4	0	0	0	3.5%	**14.8%**
METEOR	8.9%	16.5%	0.8%	23.3%	**24.0%**
ROUGE-L	4.3%	15.0%	0.8%	**22.8%**	15.1%
CIDEr	0.001	0.109	0.003	0.399	**0.892**

Table V
COMPARISON BETWEEN FINE-TUNING AND PROMPT LEARNING ON THE CCR GENERATION TASK

IV. DO THE MODELS LEARN COMMONSENSE CAUSALITY?

Last, we want to discuss whether our models learn commonsense causality.

A. *Classification Task*

We have observed all models outperform the random guess in Table III, which shows that the models learn something. But does that mean the models learn commonsense causality?

Recent works suggest that some NLP benchmarks may have annotation artefacts[9], [10]. For example, models could achieve high accuracy in classifying inference types even without seeing the premise of the SNLI task. In our task, the models may simply learn the pattern of choices and choose the correct one, regardless of the premise and question. Therefore, we may need some analysis of the classification performance without premises and questions in future work.

B. *Generation Task*

In the generation task, we only use some word/token level evaluation metrics. But simply word-level evaluation is not adequate. For example, given the sentence ”My eyes became red and puffy. What was the cause of this?”, one may answer ”I was crying”, while the ground truth is ”I was sobbing”. The BLEU-1 and BLEU-2 scores are 0.75 and 0.33, respectively. Another answer ”I was laughing” receives the same score as ”I was crying”, but the latter one is more reasonable.

We could use some semantic measures such as BERT-score[11], which uses the pre-trained embeddings from BERT and matches words in prediction and ground truth sentences by cosine similarity. We can obtain a 0.97 BERT precision score for the above example. However, semantic measures are also not enough to prove causality, as there are multiple possible different causes or effects given the premise. For example, when asking about the effect of the fact that my computer crashed, one can answer ”restart the computer” or ”lose all the data”. Both of the answers are reasonable, but they have different semantic meanings. Thus we need some metrics that actually measure causality.

V. CONCLUSION

In this paper, we compare the performance between fine-tuning models and the prompt learning model GPT-3 on two COPA CCR tasks: classification and generation. We find that all models learn the features of commonsense causality, and the GPT-3 prompt learning model outperforms the other models on the two tasks. We think GPT-3 performs well because of its vast model size and massive pre-trained data instead of the prompt learning itself. But the huge prompt learning models are still important for their ability and efficiency to task generalization.

ACKNOWLEDGEMENTS

The authors would like to thank Shaobo Cui for his instructions and suggestions for this project and his careful reading and advice for this paper. Also, thanks to the Artificial Intelligence Laboratory LIA in EPFL for providing the research opportunity for the project ML4Science.

References

[1] A. Gordon, Z. Kozareva, and M. Roemmele, “SemEval-2012 task 7: Choice of plausible alternatives: An evaluation of commonsense causal reasoning,” in **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*. Montréal, Canada: Association for Computational Linguistics, 7-8 Jun. 2012, pp. 394–398. [Online]. Available: https://aclanthology.org/S12-1052

[2] T. B. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. M. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, “Language models are few-shot learners,” 2020. [Online]. Available: https://arxiv.org/abs/2005.14165

[3] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Pre-training of deep bidirectional transformers for language understanding,” 2018. [Online]. Available: https://arxiv.org/abs/1810.04805

[4] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “Roberta: A robustly optimized bert pretraining approach,” 2019. [Online]. Available: https://arxiv.org/abs/1907.11692

[5] A. Conneau, K. Khandelwal, N. Goyal, V. Chaudhary, G. Wenzek, F. Guzmán, E. Grave, M. Ott, L. Zettlemoyer, and V. Stoyanov, “Unsupervised cross-lingual representation learning at scale,” 2019. [Online]. Available: https://arxiv.org/abs/1911.02116

[6] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, “Albert: A lite bert for self-supervised learning of language representations,” 2019. [Online]. Available: https://arxiv.org/abs/1909.11942

[7] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, “Transformers: State-of-the-art natural language processing,” in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-demos.6

[8] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, “Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension,” 2019. [Online]. Available: https://arxiv.org/abs/1910.13461

[9] S. Gururangan, S. Swayamdipta, O. Levy, R. Schwartz, S. Bowman, and N. A. Smith, “Annotation artifacts in natural language inference data,” in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, Jun. 2018, pp. 107–112. [Online]. Available: https://aclanthology.org/N18-2017

[10] M. Geva, Y. Goldberg, and J. Berant, “Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets,” 2019. [Online]. Available: https://arxiv.org/abs/1908.07898

[11] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi, “Bertscore: Evaluating text generation with bert,” 2019. [Online]. Available: https://arxiv.org/abs/1904.09675