MVA Speech and Natural Language Processing Course
Project 5: It's Not Just Size That Matters: Small Language Models Are Also Few-Shot Learners [1]

BONNIN Maxime, CHAUVIN Paul, LENOBLE Colin

March 2022

1 Introduction

Pretraining large-scale language models (LMs) on vast corpora has been a significant breakthrough in natural language processing (NLP), leading to impressive performance improvements on a wide range of downstream tasks ([2]; [3]; [4]; [5], among others). Typically, a standard approach for utilizing pre-trained LMs for a specific task is to replace the model's output layer with a task-specific head and fine-tune the entire model on a labeled dataset. However, it has been observed that language modeling can also serve as an effective pre-training objective. Many tasks can be reformulated as cloze questions, enabling pre-trained LMs to solve them with little or no labeled examples ([6]; [7]). Recently, [8] introduced GPT-3, an LLM with an unprecedented 175 billion parameters, and demonstrated remarkable few-shot abilities by reformulating tasks as LM problems. However, this method requires massive computational resources and cannot scale to more examples beyond a few hundred tokens.

An alternative approach to priming is Pattern-Exploiting Training (PET) [7], which combines cloze questions with traditional gradient-based fine-tuning. PET requires unlabeled data, which is much easier to obtain than labeled examples in many real-world applications, and it works well when the LM can predict a single token for the answer. However, this approach has a significant limitation, as many tasks cannot be formulated as predicting a single token. In this work, we reproduce the work done in this paper [1] which addresses the previous limitation by adapting PET to tasks that require predicting multiple tokens. The proposed method called iterative PET (iPET), is shown to outperform GPT-3 on the SuperGLUE [9] benchmark with only 0.1% of its parameters while requiring only a few hours of training on a single GPU. We also demonstrate that this approach achieves similar performance without unlabeled data, making it a more efficient and environmentally friendly alternative to GPT-3.

We provide a detailed analysis of the factors contributing to PET's strong performance, including its ability to combine multiple task formulations, its robustness to challenging wording, its usage of labeled data, and the characteristics of the underlying LM. Our proposed method has implications for a wide range of NLP tasks that require predicting multiple tokens and provides a more efficient and eco-friendly alternative to large LMs like GPT-3.

2 Pattern-Exploiting Training

Our work is inspired by the paper [1] that introduces a novel approach for using masked language models (MLMs) in pattern-exploiting training (PET) to map inputs to outputs. The proposed method employs pattern-verbalizer pairs (PVPs) to determine the probability of the correct output for a given input from the correct token at the masked position in the pattern.

We fine-tune an MLM on each PVP and annotate unlabeled examples with soft labels based on the ensemble of finetuned MLMs. To further enhance the performance, we implement the iterative variant of PET called iPET.

In iPET, we train several generations of models on datasets of increasing size that are labeled by previous generations. This allows MLMs trained on different patterns to learn from one another and improve their performance over time.

This approach is memory-efficient, as each model's predictions can be computed sequentially. We expect the proposed method to demonstrate promising results and apply to a range of tasks that require mapping inputs to outputs, particularly in situations where a large development set is unavailable.

3 Experiments

We decide to reproduce partially the paper [1] for a question of time and computing resources. Hence, we will focus on BoolQ [10], WiC [11] and CB [12], three of the eight tasks in SuperGLUE [9] extending the original GLUE [13] benchmark.

3.1 Selected tasks

3.1.1 BoolQ

BoolQ [10] is a QA task where each example consists of a passage p and a yes/no question q. We use the following patterns:

• p. Question: q? Answer: ...
• p. Based on the previous passage, q? ...
• Based on the following passage, q? ... . p

3.1.2 WiC

For WiC [11], given a word w and two sentences s_1 and s_2 in which it occurs, the task is to decide if w is used with the same sense in both sentences. We use:
• "s_1" / "s_2". Similar sense of "w"? ...
• s_1 s_2. Does w have the same meaning in both sentences? ...
• w. Sense (1) (a) "s_1" (...) "s_2"
For the first two patterns, we use yes as verbalization for words used in the same sense and no for other words; for the third pattern, we use b and 2.

3.1.3 CB

CB [12] is a textual entailment task like MNLI, so we use PVPs similar to [7]. For a premise p and hypothesis h, we use h? — ... , p , "h"? — ... , "p" , h? — .... p , "h"? — .... "p" and a verbalizer that maps entailment to yes, disagreement to no, and neutral to maybe.

3.2 Setup

We opted to utilize the Albert-base-v2 [14] as the underlying language model for PET as this model performed well on SuperGLUE when trained on the standard, full-sized training sets. To build our final classifier, we utilized the same model with the addition of a sequence classification head. We ran PET on the FewGLUE training sets for the three SuperGLUE tasks, without utilizing any development sets to optimize hyperparameters. We adopted the exact same setup and hyperparameters as [7]. We trained iPET on the three tasks, as these tasks' unlabeled sets contained less than 1,000 examples.

The evaluation framework is interesting when we compare approaches from few shot learning between them or with large language models. In the paper [1], they basically used three models: iPET, PET and GPT3 [8]. Due to the large number of parameters in GPT3, we were forced to use GPT2 [6] instead. So, we expect to have lower performances than those presented in the paper [1].

4 Results

4.1 Experiments

For our experimentations, we chose to train two models, Albert-v2-base and GPT2-base with respectively 11M and 117M parameters. The goal is to compare a smaller version of the paper's model Albert-v2-XXL and another model bigger with another architecture. All models were trained using NVIDIA RTX 3090 and A5000 with 24Go of RAM each. The duration per epoch of our training was 8 hours for the Albert-v2-base and 12 hours for the GPT2 model. Each model was fine-tuned during 3 epochs while using some pre-trained models found on hugging-face.

	AlbertV2-Base (11M)		
	PET	iPET	Sequence-Classifier
BoolQ (acc)	78.2	74.3	75.6
WiC (acc)	51.	-	67.1
CB (acc/F1)	73.2 / 59.5	74.4 / 63.6	66.1 / 55.2

Table 1: Performances (accuracy of F1-score) on BoolQ, WiC, and CB tasks using a pre-trained Albert-V2-Base (11M) as backbone

	GPT2-base (117M)		
	PET	iPET	Sequence-Classifier
CB (acc/F1)	63.5 / 56.3	67. / 58.3	51.2 / 40.9

Table 2: Performances (accuracy or F1-score) on CB tasks using a pre-trained GPT2-Base (117M) as backbone

	Roberta-XXL (223M)		
	PET	iPET	SotA
BoolQ (acc)	79.1	81.2	91.2
WiC (acc)	50.7	49.3	76.9
CB (acc/F1)	87.2 / 60.2	88.8 / 79.9	93.9 / 96.8

Table 3: Paper's performances (accuracy or F1-score) on BoolQ, WiC and CB tasks with an Albert-V2-XXL (223M) as backbone

4.2 Comments on our results

4.2.1 Performances on BoolQ

The experiments conducted on the SUPEGLUE tasks, BoolQ, yielded results that were generally similar to the ones in the article. The fact that they are slightly lower is certainly due to the duration of the training (only 3 epochs). Nevertheless, we observe that the accuracy of iPET is lower than that of PET which is not the case in the paper. Maybe this iterative version needs more training to reach similar results before outperforming the classical method.

4.2.2 Performances on WiC

Unlike BoolQ, this task was more complicated to run with our setup due to the limited memory. Indeed, our models (Albert-base-v2 and GPT2) have a limited input length of 1024 which is too small for a lot of samples. As a result, the too-long sentences are just truncated, losing a lot of meaningful information. In the same way as for MultiRC (and that's why we chose to change our task) some patterns were too long to be processed. Thus, in comparison with the model used in the paper, like GPT3 which uses 2048

tokens at most as input, we cannot show similar results. In addition, only the results on PET and the sequence classifier could be obtained, consistent with those of the article.

4.2.3 Performances on CB

From our experimental results, we can point out that the model performs better while using PET or iPET models than a sequence classifier showing the benefits of the method introduced by the paper [1]. Indeed, for some tasks, the classifier has almost a gap of 10% for the accuracy and the F1 score. Even if our results are still far from the paper's ones.

Other comments could be made for the GPT2 model because it had lower performances than the Albert-v2 one. One major explanation is the training time, we assessed that full potential was not used and our model could still learn a lot from more epochs. However, the efficiency of the PET and iPET in comparison with the basic classifier shows first that it performs better and learns faster.

In comparison with the previous tasks, we identified that the performances on CB are always lower than BoolQ. We do not consider WiC for our experiment because of the token length limitation. But while looking at the paper results, this is also the case for more than just BoolQ. There can be several reasons why models tend to have lower performance on the CB task compared to tasks like BoolQ in the SuperGLUE benchmark. Here are a few possible factors:

1. Complexity of Commitment Understanding: The CB task requires a deeper understanding of the commitment expressed in a given statement. It involves comprehending the nuanced levels of commitment, such as strong commitment, strong negation, or no commitment. This task demands a more sophisticated understanding of context, implicatures, and subtle linguistic cues, making it inherently more challenging for models.

2. Lack of Large-Scale Training Data: The CB dataset may not have as large a training data size as other tasks like WiC or BoolQ. Insufficient training data can limit the model's ability to learn complex patterns and generalize well to unseen examples, resulting in lower performance.

3. Annotation Difficulties: Annotating commitment levels in the text is subjective and can be challenging even for human annotators. The task requires making judgments about the strength of commitment, which can introduce inherent ambiguities. Disagreements among annotators or inconsistencies in the annotation process can impact the quality and reliability of the CB dataset, making it more difficult for models to learn effectively.

4. Linguistic Variation and Complexity: The CB task involves diverse language patterns, idiomatic expressions, and subtle contextual cues that indicate commitment levels. Capturing and comprehending these variations is crucial for accurate predictions. The complexity of natural language and the wide range of possible linguistic constructions in the CB dataset can pose challenges for models, leading to lower performance.

5 Conclusion

In this work, we reproduced partially the studied paper [1]. Because we lacked computation resources, we were not able to use the MLM having the most parameters, and hence couldn't reach the paper's results. However, we demonstrated the pertinence of iPET over PET (and thus also the sequence classifier) for the SuperGLUE [9] tasks we worked on.

References

[1] Timo Schick and Hinrich Schütze. It's not just size that matters: Small language models are also few-shot learners, 2020.

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

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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, June 2019. Association for Computational Linguistics.

[4] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach, 2019.

[5] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer, 2019.

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

[7] Timo Schick and Hinrich Schütze. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269, Online, April 2021. Association for Computational Linguistics.

[8] 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. Language models are few-shot learners. 2020.

[9] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. SuperGLUE: A Stickier Benchmark for General-Purpose Language Understanding Systems. 2019.

[10] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. 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 2924–2936, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[11] Mohammad Taher Pilehvar and Jose Camacho-Collados. WiC: the word-in-context dataset for evaluating context-sensitive meaning representations. 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 1267–1273, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[12] Marie-Catherine De Marneffe, Mandy Simons, and Judith Tonhauser. The CommitmentBank: Investigating projection in naturally occurring discourse. 2019.

[13] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. 2019.

[14] Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. ALBERT: A lite BERT for self-supervised learning of language representations. *CoRR*, abs/1909.11942, 2019.