Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/43/20230858

Large language model's capsule: A research analysis of In-context Learning (ICL) and Parameter-efficient Fine-tuning (PEFT) methods

Haokun Wu

Department of Statistics & Data Science, Carnegie Mellon University,5533 Forbes Ave Apt 2, Pittsburgh, PA 15217, United States

haokunwu@andrew.cmu.edu

Abstract. In the context of natural language processing (NLP), this paper addresses the growing need for efficient adaptation techniques for pre-trained language models. It begins by summarizing the current landscape of NLP, highlighting the challenges associated with fine-tuning large language models like BERT and Transformer. The paper then introduces and analyzes three categories of parameter-efficient fine-tuning (PEFT) approaches, namely, In-Context Learning (ICL)-inspired Fine-Tuning, Low-Rank Adaptation PEFTs (LoRA), and Activation-based PEFTs. Within these categories, it explores techniques such as prefix-tuning, prompt tuning, (IA)3, and LoRA, shedding light on their advantages and applications. Through a comprehensive examination, this paper concludes by emphasizing the interplay between performance, parameter efficiency, and adaptability in the context of NLP models. It also provides insights into the future prospects of these techniques in advancing the field of NLP. To summarize, this paper offers a detailed analysis of PEFT methods and their potential to democratize access to cutting-edge NLP capabilities, paving the way for more efficient model adaptation in various applications.

Keywords: Natural language processing, large language models, parameter-efficient fine-tuning, In-context learning, low-rank adaptation.

1. Introduction

Over the past few years, there has been an unprecedented surge in innovation and progress in the realm of natural language processing (NLP), largely attributed to the appearance of powerful large language models (LLM), for example, BERT and Transformer. These models have not only raised the bar in terms of linguistic understanding but have also achieved remarkable performance across an extensive spectrum of applications, ranging from machine translation to text generation and sentiment analysis. The profound impact of Transformers in NLP is undeniable, but their widespread utilization comes with its set of challenges, notably the intricate task of adapting these pre-trained models for specific downstream applications without resorting to extensive fine-tuning, which is often laborious, time-consuming, and resource-intensive.

In response to these challenges, a captivating solution has emerged in the form of Parameter-Efficient Fine-Tuning (PEFT). This ingenious concept represents a paradigm shift in how we leverage pre-trained language models, allowing us to harness their full potential with minimal modifications. PEFT

DOI: 10.54254/2755-2721/43/20230858

empowers researchers and practitioners to apply these language models efficiently and effectively across a multitude of tasks, thus democratizing access to cutting-edge NLP capabilities.

In this paper, we embark on a comprehensive exploration of the fascinating world of PEFT. We introduce and delve into three distinct flavors of PEFTs, each offering its unique advantages and insights into the world of efficient model adaptation. The first kind of PEFT draws inspiration from In-Context Learning, a technique that enables models to learn task-specific information from contextual examples, thereby reducing the reliance on massive amounts of task-specific data. The second variant explores the realm of low-rank adaptation structures, unlocking novel avenues for fine-tuning that are both computationally efficient and highly effective. Lastly, we explore PEFTs that operate on model activations, shedding light on innovative ways to fine-tune models by manipulating their internal representations. We aim to provide a comprehensive overview, analysis, and evaluation of these techniques, shedding light on their potential, limitations, and future prospects in the landscape of natural language processing.

2. ICL-inspired Fine-Tuning

The earliest methods aimed at efficiently fine-tuning language models drew inspiration from the concept of in-context learning. A study by [1] introduced the notion of in-context learning, wherein a language model is furnished with a prompt or context to guide its generation of relevant responses. This approach obviates the need for extensive fine-tuning by capitalizing on the model's pre-trained knowledge and supplying task-specific instructions in the form of prompts. This idea inspired the development of methods such as prefix-tuning [2], prompt tuning [3], P-tuning [4], among others. Below, we delve into the first two fine-tuning approaches.

Prefix-tuning [2] stands as a lightweight technique for adapting Transformers to conditional generation tasks (Figure 1). Unlike traditional fine-tuning, which entails retraining the entire model on task-specific data, prefix-tuning entails freezing the parameters of the pre-trained model and appending a brief vector sequence, which is especially oriented to the task, to the input. This appended sequence is optimized to facilitate the generation of desired outputs.

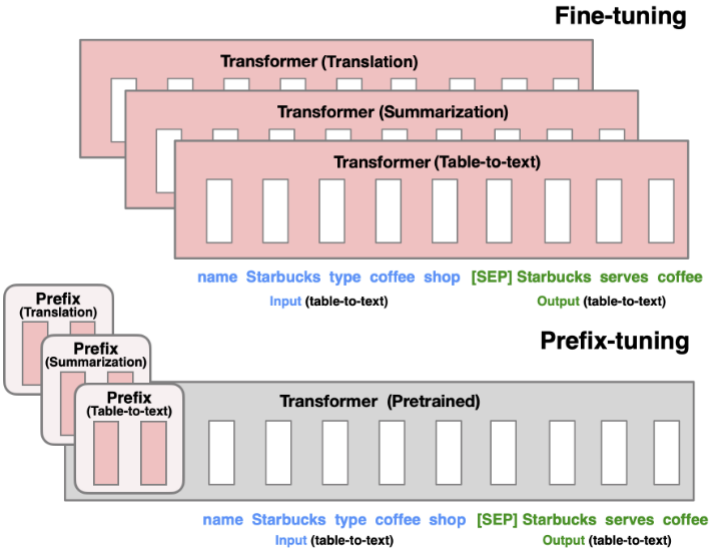

Figure 1. Prefix Tuning.

The significance of this method lies in its capacity to maintain performance levels similar to traditional fine-tuning, all while employing significantly fewer parameters. This proves particularly advantageous in scenarios with limited data, as prefix-tuning often outperforms full fine-tuning under

DOI: 10.54254/2755-2721/43/20230858

such constraints. The concept of prefix-tuning offers a promising avenue for task adaptation, simplifying the adaptation process and enabling swift deployment of language models across a variety of tasks.

Prompt tuning, introduced by [3], presents another approach to adapting language models for specific tasks, drawing inspiration from the principles of in-context learning. This method can be viewed as a streamlined version of prefix-tuning, focusing on the addition of a small number of tunable tokens to the input sequence. These tokens serve as prompts that guide the language model's generation process. Similar to prefix-tuning, prompt tuning retains the pre-trained parameters of the model while permitting task-specific adjustments through the incorporation of prompts.

One noteworthy distinction between prefix-tuning and prompt tuning lies in the way prompts are integrated into the model architecture. In prefix-tuning, a sequence of prefixes is added at each transformer layer, allowing for multiple levels of context. In contrast, prompt tuning involves adding a solitary prompt representation at the beginning of the embedded input. This approach empowers the transformer to revise task-related representations in inner layers of the language model, taking into account the context provided by the input examples.

The merits of prompt tuning become especially apparent as the scale of the language model grows. With increasingly complex and sophisticated models, prompt tuning can match or even surpass the performance of the traditional model tuning baseline and employ significantly fewer parameters. This not only increases the efficiency of task adaptation but also contributes to the overall versatility of the model across a range of tasks.

3. Low-Rank Adaptation

Although both prefix-tuning and prompt tuning have introduced novel approaches to adapt language models for downstream tasks, there's another method called Low-Rank Adaptation, or LoRA [5], which offers a parameter-efficient solution for task adaptation. In LoRA, the weights from the pre-trained model are preserve (Figure 2). What makes it novel is that it incorporates trainable rank decomposition matrices which have small ranks into the language model, parallel with the fixed weights. Via this clever technique, much fewer trainable parameters need to be trained during fine-tuning for specific downstream tasks.

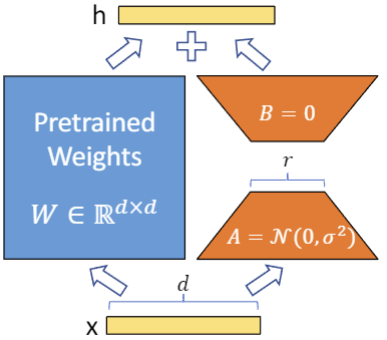

Figure 2. LoRA.

In LoRA, a low-rank decomposition is used to represent the update of a pre-trained weight matrix. Specifically, for a weight matrix W0 $\in \mathbb{R}^{d \times k}$ from pretrained model, its update $\Delta W \in \mathbb{R}^{d \times k}$ is expressed as the product of matrices B $\in \mathbb{R}^{d \times r}$ and A $\in \mathbb{R}^{r \times k}$, where r << min(d, k):

$$W = W_0 + \Delta W = W0 + BA \quad (1)$$

The key advantage of LoRA lies in its ability to efficiently adapt to various tasks without the need for additional adapter layers. Instead, the trainable matrices are seamlessly integrated with the pre-trained weights, resulting in no inference latency. Moreover, as only the injected low-rank matrices are

DOI: 10.54254/2755-2721/43/20230858

optimized, LoRA accelerates the training process and reduces the hardware requirements, making it an appealing alternative to full fine-tuning.

In the realm of LoRA, the rank r for each update matrix is typically uniform. However, the relative importance of pre-trained weight matrices varies within large language models. Assigning parameters to less significant matrices can lead to inefficiencies and suboptimal outcomes. Addressing this challenge, [6] introduced Adaptive Budget Allocation for LoRA (AdaLoRA). The parameter budget are distributed dynamically according to the significance of weight matrices, thereby improving the efficiency of the adaptation process. In AdaLoRA, the update ΔW is parameterized as PΛQ, with $P \in \mathbb{R}^{d \times r}$, $Q \in \mathbb{R}^{r \times k}$, and $\Lambda \in \mathbb{R}^{r \times r}$:

$$W = W_0 + \Delta W = W_0 + P\Lambda Q \tag{2}$$

During training, after each iteration, Λ is pruned to adjust the rank, aligning the budget with the target allocation. This adaptive approach ensures that more parameters are allocated to matrices of greater significance, resulting in improved model performance and efficiency across a range of NLP tasks.

4. Activation-based PEFTs

ICL and PEFT are two prominent approaches that makes the large language model performs better on downstream tasks. However, the choice between them depends on various factors. Liu et al. [7] have identified certain drawbacks associated with ICL. These include high computational costs, subpar performance when compared to fine-tuning, and unpredictability in its impact on task performance. On the other hand, PEFT generally exhibits better task performance and requires lower computational resources. Nevertheless, prior studies have seldom explored PEFT's performance when dealing with very limited labeled data.

To address this gap, the authors introduce T-Few, which is essentially a recipe for adapting language models to novel tasks without the need for additional fine-tuning. Here, "recipe" refers to a specific model and hyperparameter configuration that delivers robust task performance.

Furthermore, the paper presents a novel PEFT method known as “Infused Adapter by Inhibiting and Amplifying Inner Activations,” or (IA)3. This method is designed for handling mixed-task batches with minimal computational overhead while maintaining high accuracy. Taking the activations of the model, (IA)3 employs element-wise multiplication to a learned vector. For instance, consider the operation v ⊙ x, where v represents a vector with d dimensions, x is a sequence of activations with length T, and ⊙ denotes element-wise multiplication. The paper recommends the incorporation of vector scaling in self-attention and encoder-decoder architectures, as well as within the activation step between position-wise feed-forward networks. This results in the creation of a new attention mechanism as follows:

$$softmax\left(\frac{Q(v_k \odot K^T)}{\sqrt{d_k}}\right)(l_v \odot V) \tag{3}$$

Additionally, the position-wise feed-forward network undergoes a transformation as $\left(v_{ff} \odot \gamma\left(W_1 x\right)\right) W_2$. This method enables the cost-effective handling of mixed-task batches since each activation sequence within the batch are able to be individually timed by its corresponding task vector.

T-few utilizes parameters initialized from the T0 model as presented by Sanh et al. in 2021 [8]. These parameters are obtained from the pre-training process and are further adapted using (IA)3 on the same multitask mixture that was used for T0. Additionally, T-Few incorporates two supplementary loss terms designed to achieve specific objectives. The first loss term encourages the model to generate smaller probabilities for wrong choices, enhancing the model's ability to make accurate predictions. The second loss term takes into consideration the varying lengths of different answer choices, further refining the model's performance in handling diverse responses. The study's findings suggest that T-Few outperforms few-shot ICLs in terms of accuracy and computational efficiency [9,10].

DOI: 10.54254/2755-2721/43/20230858

5. Conclusion

To conclude, ICL methods, including prefix-tuning and prompt tuning, emphasize the importance of providing contextual cues or prompts to guide model responses. These techniques incorporate task-specific information in a controlled manner, reducing the need for extensive fine-tuning while optimizing performance and maintaining parameter efficiency. Notably, ICL methods achieve comparable or even superior performance to traditional fine-tuning methods while utilizing fewer parameters.

On the other hand, PEFT strategies, such as Low-Rank Adaptation (LoRA) and Activation-based approaches, offer parameter-efficient alternatives to full fine-tuning. LoRA integrates trainable rank decomposition matrices with pre-trained weights to effectively reduce the overall number of trainable parameters while preserving performance. Similarly, the Activation-based approach introduces IA3, a method that leverages activation-level adjustments to achieve accurate predictions with minimal computational overhead. Additionally, the concept of T-Few bridges the gap between ICL and PEFT by providing a methodology for achieving strong task performance even with limited labeled data.

The exploration of these methodologies reveals an intriguing interplay between performance, parameter efficiency, and adaptability. While ICL strategies like prefix-tuning and prompt tuning offer simplicity and efficiency in adapting models, PEFT methods such as LoRA and Activation-based approaches provide innovative solutions for balancing accuracy and computational resources.

References

[1] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J. D., Dhariwal, P., & Amodei, D. (2020). Language models are few-shot learners. Advances in neural information processing systems, 33, 1877-1901.

[2] Li, X. L., & Liang, P. (2021). Prefix-tuning: Optimizing continuous prompts for generation. arXiv preprint arXiv:2101.00190.

[3] Lester, B., Al-Rfou, R., & Constant, N. (2021). The power of scale for parameter-efficient prompt tuning. arXiv preprint arXiv:2104.08691.

[4] Liu, X., Zheng, Y., Du, Z., Ding, M., Qian, Y., Yang, Z., & Tang, J. (2021). GPT Understands, Too. arXiv e-prints, arXiv-2103.

[5] Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., ... & Chen, W. (2021). Lora: Low-rank adaptation of large language models. arXiv preprint arXiv:2106.09685.

[6] Zhang, Q., Chen, M., Bukharin, A., He, P., Cheng, Y., Chen, W., & Zhao, T. (2023). Adaptive budget allocation for parameter-efficient fine-tuning. arXiv preprint arXiv:2303.10512.

[7] Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M., & Raffel, C. A. (2022). Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. Advances in Neural Information Processing Systems, 35, 1950-1965.

[8] Sanh, V., Webson, A., Raffel, C., Bach, S. H., Sutawika, L., Alyafeai, Z., & Rush, A. M. (2021). Multitask prompted training enables zero-shot task generalization. arXiv preprint arXiv:2110.08207.

[9] Liu, H. , Tam, D. , Muqeeth, M. , Mohta, J. , Huang, T. , & Bansal, M. , et al. (2022). Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning.10.48550/arXiv.2205.05638

[10] Liu, J. , & Rajati, M. R. . (2020). Transfer Learning with Shapeshift Adapter: A Parameter-Efficient Module for Deep Learning Model. 2020 International Conference on Machine Learning and Cybernetics,1-12.