Literature Review: LoRA: Low-Rank Adaptation of Large Language Models

Mika Senghaas | 377332 | mika.senghaas@epfl.ch
MLP

1 Summary

Over the past years, fine-tuning has become the de facto standard in deep learning for achieving state-of-the-art performance on specialised tasks. Traditionally, it requires training *all* pre-trained weights of a large model on a smaller task-specific dataset. However, with the growing size of pre-trained models, such as the 150B parameter GPT-3 (Brown et al., 2020), full fine-tuning becomes prohibitively expensive in terms of the hardware required for training and deployment.

This development motivated a group of Microsoft researchers to develop Low-Rank Adaptation (**LoRA**) (Hu et al., 2021), a novel parameter-efficient fine-tuning method.

It is based on the simple idea that any pre-trained dense layer W_0, used during a forward-pass $h = W_0 x$, can be adapted by augmenting the forward pass through a trainable adaptation matrix ∇W, such that $h = W_0 x + \nabla W x$. Motivated by prior work showing that overparameterised models reside on low intrinsic dimensions (Aghajanyan et al., 2020), the authors propose using a pair of low-rank decomposition matrices, B and A, as their adaptation matrix. The adapted forward pass becomes

$$h = W_0 x + \nabla W x = W_0 x + BAx,$$

where $W_0 \in R^{d \times k}$ is the pre-trained weight matrix, and $B \in R^{d \times r}$ and $A \in R^{r \times d}$ are the adaptation matrices with rank $r \ll \min(d, k)$. During training, the original weight matrix W_0 is frozen, and only the adaptation matrices B and A are trained. During inference, the task-specific LoRA weights are merged with the pre-trained weights $W' = W_0 + BA$ to adapt the forward pass. The method's advantages can be summarised as follows:

Fewer Trainable Parameters. The low-rank dimension significantly reduces the number of trainable parameters - in the case of GPT-3, by up to 10,000x. This decreases the storage required for saving checkpoints from 350GB to 35MB for GPT-3 as only the LoRA matrices need to be stored.

High Performance. LoRA fine-tuning performs on par or better when compared to full-finetuning and other parameter-efficient fine-tuning methods discussed in the literature (Houlsby et al., 2019; Li and Liang, 2021).

No Additional Inference Latency. Unlike other parameter-efficient fine-tuning methods, LoRA does not introduce any additional latency during inference due to weight merging.

Task Switching A set of LoRA weights for multiple tasks can easily be "hot-swapped" by simple addition and subtraction.

I believe LoRA is a significant contribution to the field of parameter-efficient fine-tuning of large pre-trained models, marking a milestone towards democratising the use of state-of-the-art language models. Throughout, the paper argues convincingly for the method's effectiveness and provides extensive empirical and theoretical evidence to support their claims.

2 Strengths

The paper proposes a conceptually simple, yet powerful, idea for parameter-efficient fine-tuning of large pre-trained models.

Clarity and Conciseness. Throughout the entire report, the authors provide a clear and concise outline of the method, its motivation, and its expected advantages. Background information is provided where necessary, making the paper easy to follow.

Solves a Real Problem. The authors address some of the most pressing challenges researchers and practitioners face when fine-tuning enormous pre-trained models, they reduce the time and memory footprint during training and the storage requirements for model checkpoints during experimentation and deployment.

Grounded in Research. The method is well-motivated by previous literature - it extends empirical findings that overparameterised models reside on low intrinsic dimensions (Aghajanyan et al., 2020). Further, the authors make sure to clearly differentiate their approach from previous methods, such as adapter tuning (Houlsby et al., 2019) and prefix tuning (Li and Liang, 2021), by highlighting the limitations of these methods and how LoRA overcomes them. This helps the reader to understand the novelty and importance of the method.

Extensive Empirical Analysis. The authors provide extensive empirical evidence to support their claims. They evaluate LLMs of varying sizes, like RoBERTa (Liu et al., 2019), DeBERTa (He et al., 2021), GPT-2 (Radford et al., 2018), and GPT-3 (Brown et al., 2020) on a range of Natural Language Understanding (NLU) and Natural Language Generation (NLG) benchmarks, such as GLUE (Wang et al., 2019). Overall, the experimental results convince the reader of the method's effectiveness.

Theoretical Analysis. To further strengthen their claims the authors conduct an ablation study on the hyper-parameters of the method and provide a theoretical analysis of the adapted matrices. First, they find that very low-rank dimensions of $r = 1$ or $r = 2$ already perform competitively and that performance plateaus for higher rank dimensions. Second, they show that the direction corresponding to the top singular vector overlap significantly between learned adaptation matrices of various sizes. Third, they show the adaptation matrix amplifies directions that are not emphasised in the original pre-trained matrix, hence highlighting the important features for a specific downstream task that were learned but not emphasised in the pre-trained matrix. All of these findings provide a deeper understanding of the method and its inner workings.

Research Contributions. The authors' release of code, model checkpoints, as well as the creation of an open-source library (Hu et al., 2024), is a clear commitment to the research community and promotes the adoption and extension of the method. Today, roughly three years after the release of the pre-print, the method is a standard fine-tuning technique in the deep learning community and is integrated into major deep learning libraries, such as Huggingface (HF, 2024). Moreover, the method has sparked further research in the field of parameter-efficient fine-tuning and has been extended in various ways, with a popular example being its quantised counter-part, QLoRA (Dettmers et al., 2023).

3 Weaknesses

However, the paper is not without its shortcomings.

Reliability of Empirical Results. In many instances the benchmark results are only marginally better than existing methods - often falling within the ranges of the standard deviations. The authors fail to acknowledge the marginal differences in their discussion and make strong statements. This might mislead an uncautious reader into wrongly thinking that alternative approaches are obsolete.

Novelty. Existing methods have already explored the idea of injecting low-rank adaptation matrices for parameter-efficient fine-tuning. The only functional difference is that LoRA allows for easy weight merging which removes the latency penalty incurred by sequential adapter layers. A critical reader could question the novelty of the method.

Inference Latency. The authors state that LoRA does not trivially support batching of inputs for different tasks, as LoRA weights are task-specific. They mention a possible solution is to not merge the LoRA weights but dynamically swap them during inference based on the task. It would be interesting to see the added latency incurred by not merging model weights. This could guide practitioners to make an informed decision on possibly choosing another parameter-efficient finetuning method over LoRA.

References

Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. 2020. Intrinsic Dimensionality Explains the Effectiveness of Language Model Fine-Tuning.

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. 2020. Language Models are Few-Shot Learners.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. QLoRA: Efficient Finetuning of Quantized LLMs.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: Decoding-enhanced bert with disentangled attention.

HF. 2024. HuggingFace Documentation: LoRA. https://huggingface.co/docs/diffusers/en/training/lora.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-Efficient Transfer Learning for NLP.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. LoRA: Low-Rank Adaptation of Large Language Models.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2024. LoRA: Low-Rank Adaptation of Large Language Models. https://github.com/microsoft/LoRA.

Xiang Lisa Li and Percy Liang. 2021. Prefix-Tuning: Optimizing Continuous Prompts for Generation.

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

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2018. Language Models are Unsupervised Multitask Learners.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding.