LoRA Training in the NTK Regime has No Spurious Local Minima

Uijeong Jang
Seoul National University, Math

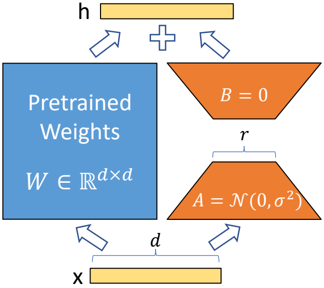
Jason D. Lee
Princeton, ECE

Ernest K. Ryu
UCLA, Math

International Conference on Machine Learning
Oral Presentation
July 25, 2024

LoRA background

Low-Rank Adaptation (LoRA) fine-tunes large pre-trained language models by introducing low-rank updates to the attention layers.[1]



Given a linear layer mapping

$$x \mapsto Wx$$

LoRA introduces the rank-r update

$$x \mapsto (W + BA)x$$

The W weights are frozen (not trained) while $A \in \mathbb{R}^{r\times d}$ and $B \in \mathbb{R}^{d\times r}$ are trained.

LoRA reduces memory cost. To fine-tune LLMs on academic GPU hardware (bottlenecked by GPU memory) LoRA is mandatory.[2]

[1] E. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, LoRA: Low-rank adaptation of large language models, *ICLR*, 2022

[2] T. Dettmers, A. Pagnoni, A. Holtzman, and L. Zettlemoyer, QLoRA: Efficient finetuning of quantized LLMs, *NeurIPS*, 2023.

Prior work on LoRA

Empirical research on LoRA:

Enormous body of work! 2022 LoRA paper has over 5300 as of today.
Prevalence of LoRA warrants theory research.

Theoretical research on LoRA:

Only a handful of papers. [3] [4] [5] [6]

---

[3] Y. Zeng and K. Lee, The expressive power of low-rank adaptation, *ICLR*, 2024.

[4] S. Lotfi, M. A. Finzi, Y. Kuang, T. G. J. Rudner, M. Goldblum, and A. G. Wilson, Non-vacuous generalization bounds for large language models, *ICML*, 2024.

[5] C. Yaras, P. Wang, L. Balzano, and Q. Qu, Compressible dynamics in deep overparameterized low-rank learning & adaptation, *ICML*, 2024.

[6] J. Y.-C. Hu, M. Su, E.-J. Kuo, Z. Song, and H. Liu, Computational limits of low-rank adaptation (LoRA) for transformer-based models, *arXiv*, June 2024.

Problem setup

- Transformer network: $f_\Theta \colon \mathcal{X} \to \mathbb{R}$.
- Subset of weights (dense layers in QKV-attention) that we fine-tune: $\mathbf{W} = (W^{(1)}, \dots, W^{(T)}) \subset \Theta$.
- In this talk, set $T = 1$ for notational simplicity.
- Pre-trained weights: $\mathbf{W}_0 \subset \Theta_0$.
- Fine-tuning data: $\{(X_i, Y_i)\}_{i=1}^N$. (Think of $N < 1000$.)
- Fine-tuning update: $\boldsymbol{\delta} \subset \Theta$, i.e., $f_{\mathbf{W}_0 + \boldsymbol{\delta}}$ is fine-tuned model.
- Let ℓ be MSE or cross-entropy loss.

Full fine-tuning:

$$\underset{\boldsymbol{\delta}}{\text{minimize}} \quad \hat{\mathcal{L}}(\boldsymbol{\delta}) = \frac{1}{N} \sum_{i=1}^N \ell(f_{\mathbf{W}_0 + \boldsymbol{\delta}}(X_i), Y_i).$$

LoRA fine-tuning:

$$\underset{\mathbf{u}, \mathbf{v}}{\text{minimize}} \quad \hat{\mathcal{L}}(\mathbf{u}\mathbf{v}^\intercal) = \frac{1}{N} \sum_{i=1}^N \ell(f_{\mathbf{W}_0 + \mathbf{u}\mathbf{v}^\intercal}(X_i), Y_i).$$

Weight decay on LoRA is nuclear norm regularization

LoRA training often uses weight decay. Can be interpreted as solving

$$\underset{\mathbf{u},\,\mathbf{v}}{\text{minimize}} \quad \hat{\mathcal{L}}(\mathbf{u}\mathbf{v}^\intercal) + \frac{\lambda}{2}\|\mathbf{u}\|_F^2 + \frac{\lambda}{2}\|\mathbf{v}\|_F^2,$$

with regularization parameter $\lambda \geq 0$. By [7], this is equivalent to

$$\underset{\boldsymbol{\delta},\,\mathrm{rank}\boldsymbol{\delta}\leq r}{\text{minimize}} \quad \hat{\mathcal{L}}_\lambda(\boldsymbol{\delta}) \triangleq \hat{\mathcal{L}}(\boldsymbol{\delta}) + \lambda\|\boldsymbol{\delta}\|_*,$$

where $\boldsymbol{\delta} = \mathbf{u}\mathbf{v}^\intercal$ and $\|\cdot\|_*$ is the nuclear norm (sum of singular values).

Insight: Weight decay induces nuclear norm regularization, which, in turn, induces low-rank updates.

[7] B. Recht, M. Fazel, and P. A. Parrilo, Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization, *SIAM Review*, 2010.

The NTK assumption

If the first-order Taylor approximation holds throughout training

$$f_{\mathbf{W}_0+\boldsymbol{\delta}}(X) \approx f_{\mathbf{W}_0}(X) + \langle \nabla f_{\mathbf{W_0}}(X), \boldsymbol{\delta} \rangle$$

we say training stays within the NTK regime. This approximation is justified empirically[8] when prompt-based fine-tuning is used.

Consider the loss with the linearized neural network

$$\hat{L}(\boldsymbol{\delta}) = \frac{1}{N} \sum_{i=1}^{N} \ell \left(f_{\mathbf{W}_0}(X_i) + \langle \nabla f_{\mathbf{W}_0}(X_i), \boldsymbol{\delta} \rangle, Y_i \right)$$

instead of the actual loss $\hat{\mathcal{L}}$.

$$\hat{\mathcal{L}}(\boldsymbol{\delta}) = \frac{1}{N} \sum_{i=1}^{N} \ell(f_{\mathbf{W}_0+\boldsymbol{\delta}}(X_i), Y_i).$$

In the following theorems, we assume

$$\hat{L}(\boldsymbol{\delta}) \approx \hat{\mathcal{L}}(\boldsymbol{\delta})$$

and analyze $\hat{L}(\boldsymbol{\delta})$ instead of $\hat{\mathcal{L}}(\boldsymbol{\delta})$.

[8] S. Malladi, A. Wettig, D. Yu, D. Chen, and S. Arora, A kernel-based view of language model fine-tuning, *ICML*, 2023.

Theorem 1: Existence

Theorem 1

Let $\lambda \geq 0$. Assume $\hat{L}_\lambda(\boldsymbol{\delta})$ has a global minimizer. In the full fine-tuning setup, there is a rank-r solution such that $\frac{r(r+1)}{2} \leq N$. (So $r \lesssim \sqrt{N}$.)

Great! A low-rank solution exists, so using LoRA makes sense.

So, then, can we find the low-rank solution with SGD?

Background: Strict saddles vs. SOSP

U is a (first-order) *stationary* point if

$$\nabla \hat{L}(U) = \mathbf{0}.$$

U is a *second-order stationary point* (SOSP) if

$$\nabla \hat{L}(U) = \mathbf{0}, \qquad \nabla^2 \hat{L}(U)[V, V] \geq 0,$$

for any direction $V \in \mathbb{R}^{m \times n}$. (Hessian has no negative eigenvalues.)

U is *strict saddle* if it is a first- but not second-order stationary point.

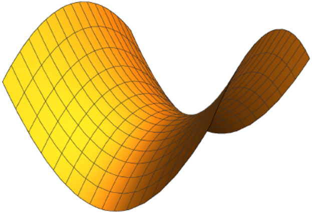


Figure: A strict saddle

Background: SGD avoids strict saddles

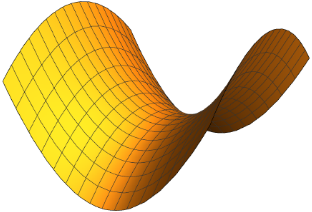


Figure: A strict saddle

Stochastic gradient descent (SGD) does not converge strict saddle points.[9] [10] SGD only converges to SOSP.

In general, however, an SOSP can be non-global local minima (spurious local minima). In our setup, all SOSPs are global minima, so SGD converges to global minima.

[9] R. Ge, F. Huang, C. Jin, and Y. Yuan, Escaping From Saddle Points — Online Stochastic Gradient for Tensor Decomposition, *COLT*, 2015.

[10] J. D. Lee, M. Simchowitz, M. I. Jordan, and Benjamin Recht, Gradient descent only converges to minimizers, *COLT*, 2016.

Theorem 2: Trainability

Theorem 2

Let $\lambda \geq 0$*. Assume* $\hat{L}_\lambda(\boldsymbol{\delta})$ *has a global minimizer and* $\frac{r(r+1)}{2} > N$*. Consider the perturbed loss function*

$$\hat{L}_{\lambda,P}(\mathbf{u},\mathbf{v}) \triangleq \hat{L}(\mathbf{u}\mathbf{v}^\intercal) + \frac{\lambda}{2}\|\mathbf{u}\|_F^2 + \frac{\lambda}{2}\|\mathbf{v}\|_F^2 + \underbrace{\begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix}^\intercal P \begin{bmatrix}\mathbf{u}\\ \mathbf{v}\end{bmatrix}}_{\text{small perturbation}} .$$

If $P \in \mathbb{S}_+^{(m+n)}$ *is a small random perturbation, all SOSPs of* $\hat{L}_{\lambda,P}$ *are global minimizers with probability* 1*.*

Generically, LoRA training has no spurious local minima!

$\hat{L}_{\lambda,P}$ has saddle points, but SGD won't converge to them.
SGD converges to an SOSP, which is a global minimum.

Theorem 3: Generalization

LoRA with weight decay is nuclear-norm regularized training.
So, standard Rademacher arguments yield generalization guarantees.

Theorem 3

Assume the population risk L has a minimizer $\boldsymbol{\delta}^\star_{\mathrm{true}}$. We randomly sample P. Let $(\hat{\mathbf{u}}, \hat{\mathbf{v}}) \in \operatorname{argmin} \hat{L}_{\lambda,P}(\hat{\mathbf{u}}\hat{\mathbf{v}}^\intercal)$. Under certain conditions,

$$L(\hat{\mathbf{u}}\hat{\mathbf{v}}^\intercal) - L(\boldsymbol{\delta}^\star_{\mathrm{true}}) < \tilde{\mathcal{O}}\Big(\frac{\|\boldsymbol{\delta}^\star_{\mathrm{true}}\|_*}{\sqrt{N}}\Big)$$

with high probability.

(The omitted conditions are what you would expect from a Rademacher complexity argument.)

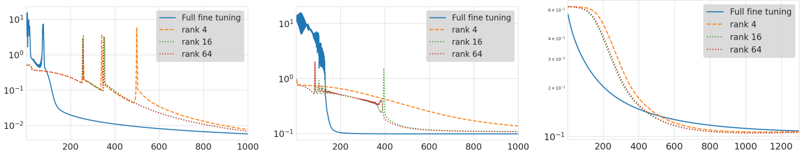
Experiments

Observation: Rank r (if $r \gtrsim \sqrt{N}$) doesn't affect where we converge to, but higher rank (or full fine-tuning) leads to faster convergence.

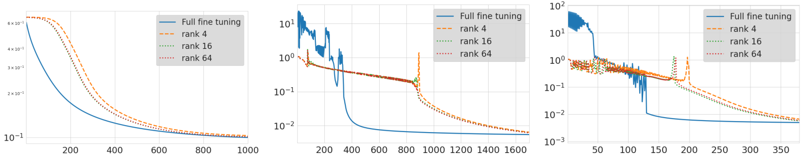
(Theorem 2 implies convergence. Says nothing about convergence *speed*.)

Trade-off: Smaller r uses less memory but requires more training epochs.

Experiments: NLP tasks



(a) SST-2 (b) QNLI (c) MR



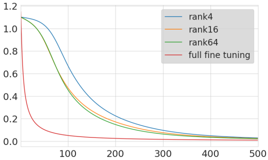
(d) CR (e) QQP (f) Subj

Fine-tuning RoBERTa-base[11] different NLP tasks with dataset size $N = 32$ using cross-entropy loss.

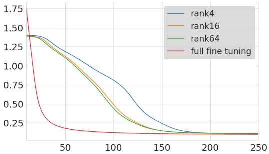
---

[11] 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.

Experiments: Image and speech classification tasks



(a) Image classification



(b) Speech classification

Fine-tuning vision transformer[12] and wav2vec[13].

[12] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, An image is worth 16x16 words: Transformers for image recognition at scale, *ICLR*, 2021.

[13] A. Baevski and Y. Zhou and A. Mohamed and M. Auli, wav2vec 2.0: A framework for self-supervised learning of speech representations, *NeurIPS*, 2020.

Conclusion

Provides a trainability and generalization analysis of LoRA fine-tuning.
Future directions:

- In practice, $r = 4$ is successfully used. Not explainable by our theory.
- When NTK assumption is violated, our theory doesn't apply.
- Theory on convergence speed of LoRA training is needed.
- Many more interesting questions!

LoRA Training in the NTK Regime has No Spurious Local Minima,
Uijeong Jang, Jason D. Lee, and **Ernest K. Ryu**, *ICML*, 2024.

SAMSUNG

We thank Samsung for sponsoring the Global Research Symposium, where the initial idea of this work was conceived.

UCLA

I just moved to UCLA, and I am recruiting! If you want to work on optimization and/or deep learning theory, feel free to contact me.