SuperLoRA: Parameter-Efficient Unified Adaptation for Large Vision Models

Xiangyu Chen[1,2], Jing Liu[2], Ye Wang[2], Pu (Perry) Wang[2],
Matthew Brand[2], Guanghui Wang[3], Toshiaki Koike-Akino[2]

[1] University of Kansas, Lawrence, KS 66045, USA

[2] Mitsubishi Electric Research Laboratories (MERL), Cambridge, MA 02139, USA

[3] Toronto Metropolitan University, Toronto, ON M5B 2K3, Canada

xychen@ku.edu, {xiachen, jiliu, yewang, pwang, brand, koike}@merl.com, wangcs@torontomu.ca

Abstract

Low-rank adaptation (LoRA) and its variants are widely employed in fine-tuning large models, including large language models for natural language processing and diffusion models for computer vision. This paper proposes a generalized framework called SuperLoRA that unifies and extends different LoRA variants, which can be realized under different hyper-parameter settings. Introducing new options with grouping, folding, shuffling, projection, and tensor decomposition, SuperLoRA offers high flexibility and demonstrates superior performance, with up to 10-fold gain in parameter efficiency for transfer learning tasks.

1. Introduction

Large neural network models are dominating machine learning recently with the emergence of exceptional models, such as large vision models (LVMs) including Vision Transformer (ViT) [10], ConvNeXt [33] and Stable Diffusion [19] for vision tasks, and large language models (LLMs) including GPT [1], PALM2 [4], Gemini [3] and LLaMA2 [39] for natural language processing (NLP). However, the increased resource consumption and data requirement along with model size limits its generalization on downstream tasks. To solve this, Parameter-Efficient Fine-Tuning (PEFT) has been widely explored to fine-tune less parameters while retaining high performance. Among this, adapter-based technique like LoRA (**Lo**w-**R**ank **A**daptation) [21] demonstrates advantages and flexible convenience.

LoRA [21] approximates the weight updates of the base model by approximating the change ΔW of each weight matrix as the product of two low-rank matrices. This decreases the required parameters from d^2 to $2rd$ when $r \ll d$, where d and r are weight size and the rank, respectively. Most LoRA variants work on solving the inherent *low-rank constraint* of matrix factorization, including LoHA (**Lo**w-rank **Ha**damard) [42], LoKr (**Lo**w-rank **Kr**onecker) [42], and LoTR (**Lo**w **T**ensor **R**ank) [5]. We discuss more related work in Appendix A. However, we find these variants can be nicely unified within our framework—SuperLoRA—with different hyper-parameters as shown in Table 1. Our proposed SuperLoRA framework is depicted in Figure 1, which also yields to some new variants: LoNKr (**Lo**w-rank **N**-split **Kr**onecker) and LoRTA (**Lo**w-**R**ank **T**ensor **A**daptation). Additionally, we introduce three extended options: 1) reshaping ΔW to any arbitrary multi-dimensional tensor arrays before applying LoRA variants; 2) splitting all ΔW into an arbitrary number of groups, which breaks the boundaries for ΔW across different weights; and 3) projecting fewer number of trainable parameters into larger weights through a projection layer $\mathcal{F}(\cdot)$ with fixed parameters. Accordingly, SuperLoRA provides more flexibility and extended functionality, controlled by a set of hyper-parameters listed in Table 2. Our contributions include:

- We propose a new PEFT framework SuperLoRA which gracefully unifies and extends most LoRA variants.
- With projected tensor rank decomposition, SuperLoRA can adapt all weights across layers jointly with a wide range of adjustable parameter amount.
- We investigate the effect of tensor reshaping, grouping, random projection, and shuffling.
- We demonstrate high parameter efficiency for large ViT and diffusion models in two transfer learning tasks: image classification and image generation.
- Significant parameter reduction by up to 10 folds can be achieved.

2. SuperLoRA

Figure 1 shows the overview of SuperLoRA, which is a generalization of LoRA variants to allow high flexibility in the

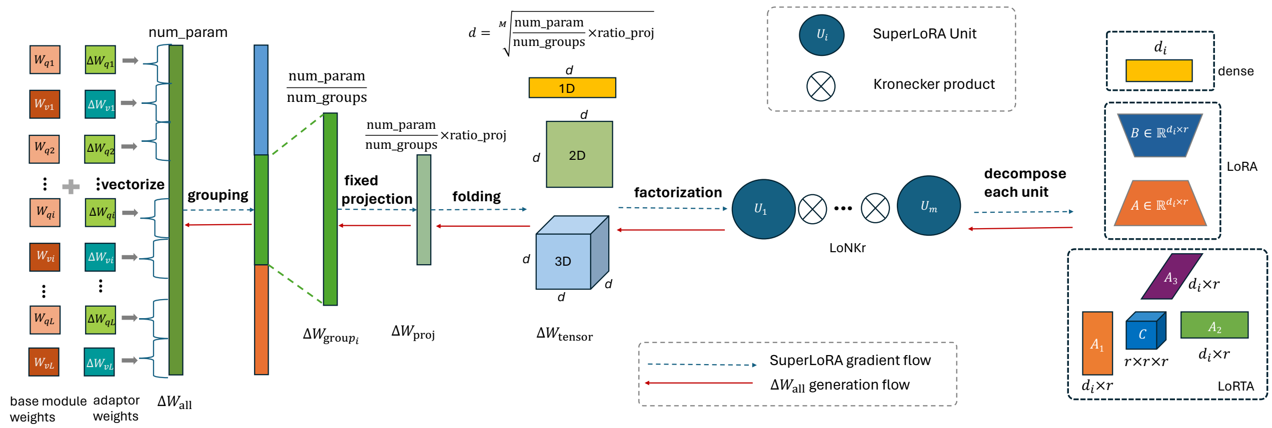

Figure 1. Schematic of SuperLoRA to fine-tune multi-layer attention modules at once with grouping, projection, folding, and factorization.

Table 1. Hyper-parameter settings in SuperLoRA and the resultant LoRA variant

hyper-parameters settings	method
$\mathcal{F} = I$, weight-wise, $K = 1$, $C_{g1} = I$, $M = 1$	dense FT
$\mathcal{F} = I$, weight-wise, $K = 1$, $C_{g1} = I$, $M = 2$	LoRA [21]
$\mathcal{F} = I$, weight-wise, $K = 2$, $C_{gk} = I$, $M = 2$	LoKr [42]
$\mathcal{F} = I$, group-wise, $G = 1$, $M > 2$	LoTR [5]
$\mathcal{F} = I$, group-wise, $K > 2$, $C_{gk} = I$, $M = 2$	LoNKr
$\mathcal{F} = I$, group-wise, $K = 1$, $M > 2$	LoRTA

Table 2. Hyperparameters and notation.

notation	description
r	rank of factorization
$\mathcal{F}$	mapping function
ρ	compression ratio
G	number of groups
M	order of tensor modes
K	number of splits

weight update ΔW. SuperLoRA can be formulated as:

$$\Delta W_{\mathrm{group}_g} = \mathcal{F}\Big(\bigotimes_{k=1}^{K} \big(C_{gk} \times_1 A_{gk1} \times_2 \cdots \times_M A_{gkM}\big)\Big),$$

where $\mathcal{F}(\cdot)$ is a simple projection function applied on the results of SuperLoRA modules. We denote $\times_m$ as mode-m tensor product, and $\otimes$ as Kronecker product. Here, M represents the order of the reshaped weight tensor modes, and high-order Tucker decomposition [41] is employed to formulate this high-order tensor, where $C_{gk} \in \mathbb{R}^{r_1 \times r_2 \times \cdots \times r_M}$ is M-D core tensor and $A_{gkm} \in \mathbb{R}^{d_m \times r_m}$ are 2D plane factors. SuperLoRA units in Figure 1 are combined with Kronecker product across K splits in a proper shape. Depending on reshaping, each split has multiple choices including a combination of dense fine-tuning (FT: 1D), LoRA (2D), and high-order Tucker decomposition (3D, 4D, etc.).

For SuperLoRA as depicted in Figure 1, we first concatenate all $\Delta W \in \mathbb{R}^{d_i \times d_i}$ across multiple layers to get the total correction of $\Delta W_{\mathrm{all}} \in \mathbb{R}^{\sum_i d_i^2}$. Then, ΔW_{all} is divided into g groups: $\{\Delta W_{\mathrm{group}_g}\}$ for $g \in \{1, 2, \ldots, G\}$. Each LoRA module will then produce $\Delta W_{\mathrm{group}_g}$. Finally, stretch $\Delta W_{\mathrm{group}_g}$ to one dimension, fetch corresponding size of ΔW from those $\Delta W_{\mathrm{group}_g}$ and add it to candidate weight matrix, *e.g.*, query and value projection weights for attention modules across layers. Figure 2 shows the grouping mechanism which provides various options, including weight-wise, layer-wise, and general grouping. Reshaping in Figure 2(c) can solve unbalanced fan-in/fan-out issue in Figure 2(b) when stacking multiple weights.

SuperLoRA can further modify the tensor arrays through a simple mapping $\mathcal{F}(\cdot)$: *e.g.*, we can project much smaller $\Delta W_{\mathrm{lora}_g}$ into larger final $\Delta W_{\mathrm{group}_g}$ to improve the parameter efficiency. We use the fastfood projection [2, 28] as shown in Figure 3, which is written as follows:

$$\begin{aligned} \Delta W_{\mathrm{group}_g} &= \mathcal{F}(\Delta W_{\mathrm{lora}_g}) \\ &= \mathsf{vec}[\Delta W_{\mathrm{lora}_g}]\, \mathcal{H}'\, \mathsf{diag}[\mathcal{G}]\, \Pi\, \mathcal{H}\, \mathsf{diag}[\mathcal{B}], \end{aligned}$$

where $\mathsf{vec}[\cdot]$ is a vectorization operator, $\mathsf{diag}[\cdot]$ denotes a diagonalization operator, $\mathcal{H}$ is left-truncated Walsh–Hadamard matrix, $\mathcal{H}'$ is its right-truncated version, $\mathcal{G}$ is a random vector drawn from normal distribution, Π is a random permutation matrix, and $\mathcal{B}$ is a random vector drawn from Rademacher distribution. The compression ratio for the projection $\mathcal{F}(\cdot)$ is $\rho = |\Delta W_{\mathrm{lora}_g}|/|\Delta W_{\mathrm{group}_g}|$, where $|\cdot|$ denotes the total number of elements of the tensor. It is a fast Johnson–Lindenstrauss transform with log-linear

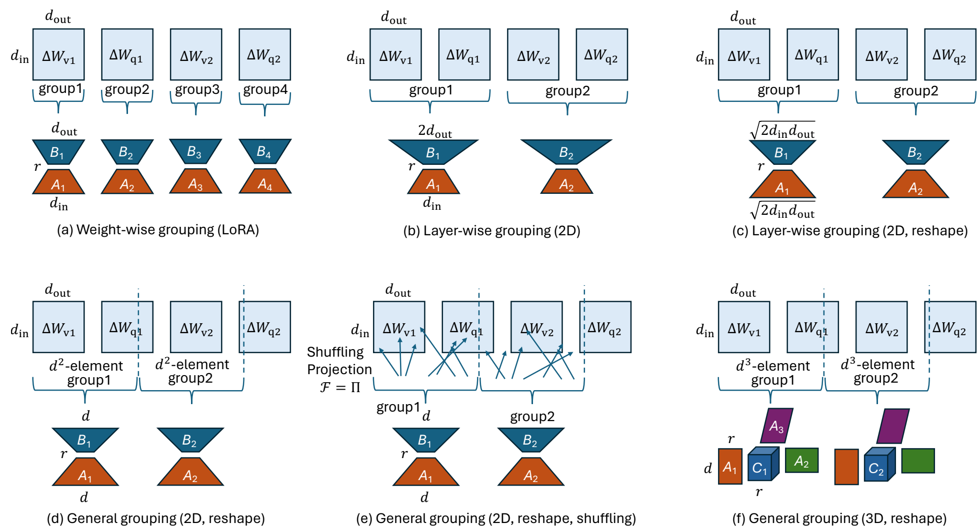

Figure 2. Examples of grouping mechanism.

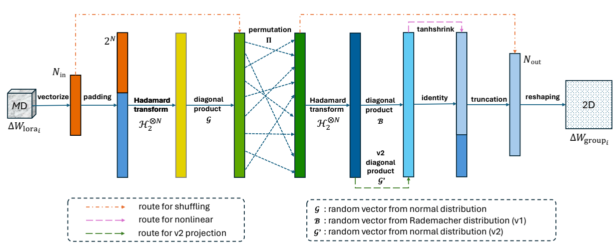

Figure 3. Illustration of fastfood projection and its variants.

complexity due to the fast Walsh–Hadamard transform, and no additional parameters are required when the random seed is predetermined. The projection also includes a shuffling variant as in Figure 2(e). More details of SuperLoRA framework are found in Appendix A.2, and its different variants are discussed in Appendix A.10.

3. Transfer Learning Experiments

Transfer learning for image classification is conducted between ImageNet21k [9] and CIFAR100 [26] based on a ViT-base [10] model. More details of the ViT model are described in Appendix A.3. The query and value projection layers in the attention modules are fine-tuned with SuperLoRA. The model is trained for 5,000 steps with the stochastic gradient descent (SGD) optimizer, with a batch size of 128 and a learning rate of 0.05. The OneCycleLR [38] scheduler is used.

We evaluated SuperLoRA with grouping with/without reshaping to square-like for 2D $\Delta W_{\text{group}_g}$, reshaping version for higher-order $\Delta W_{\text{group}_g}$ including 3D, 4D and 5D. The fixed projection layers are inserted to SuperLoRA with reshaping (2D version) and also dense. Original weight-wise LoRA is also examined for comparison by setting the number of groups to the number of query and value weights (24 for 12-layer ViT-base) as all projection weights for ViT-base are equal size. Each correction weight is of size 768×768 as the projection weight for query/value, resulting in 14M parameters. Except for most cases, more ranks are needed to span the parameter axis well, including larger ranks from 34 to 128 and smaller ranks below 8 for LoRTA. Projection compression ratio is from $\rho \in \{0.5, 0.25, 0.1, 0.01\}$, and the fixed projection parameters are shared across all groups in our experiments.

Classification results versus the number of parameters are shown in Figure 4 with Pareto frontier lines. Comparing group-wise SuperLoRA (2D with/without reshape) with weight-wise LoRA, we can find that SuperLoRA versions show better performance in terms of the trade-off between classification accuracy and the number of parameters. Noticeably, we observe three to four times advantage in terms of parameter efficiency for the same accuracy. As the largest number of groups is set to 24 (*i.e.* LoRA), it indicates smaller number of groups are superior. This may be because ViT model is excessively large for the CIFAR100 dataset, with much more redundant weights. Grouping weights and layers together can reduce noise brought by the redundancy. With reshaping $\Delta W_{\text{group}_g}$ to a square matrix, classification accuracy further increases in the lower parameter regime and the range of parameters the model can cover becomes wider as higher rank can be used while maintaining a smaller number of parameters.

To examine the effect of higher-order tensor folding, the order M is set to be 3, 4 and 5 for SuperLoRA (*i.e.* LoRTA) as well as 2. For $M = 2$ cases with 2D tensor, we use

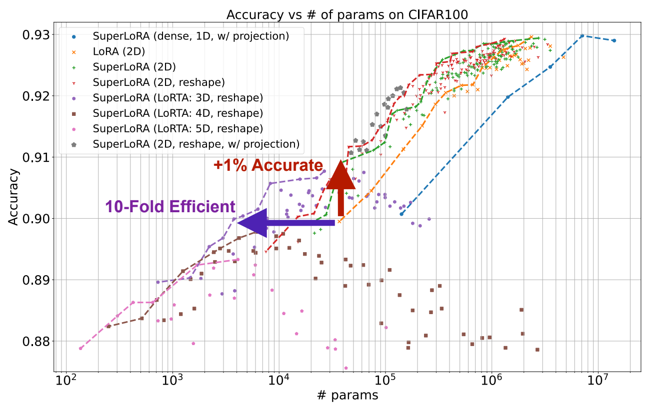

Figure 4. Transfer learning from ImageNet21K to CIFAR100, parameters in classifier head excluded.

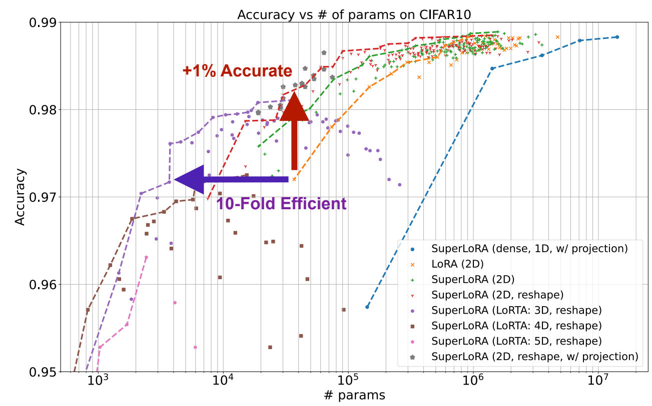

Figure 5. Transfer learning from ImageNet1K to CIFAR10, with frozen classifier head after manual label matching.

identity core tensor like typical LoRA. With the increase of order from 2 to 5, higher order takes place lower-order at fewer-parameter regimes. Moreover, data points for high-order LoRTA show a hill-like trend with the increase of parameters. This may be caused by the inefficient core tensor, which increases parameters rapidly without benefiting the accuracy. When comparing the lowest rank LoRA (which achieves around 0.9 accuracy with about 4×10^4 parameters), our LoRTA (3D) significantly improves the accuracy by about 1% at the comparable number of parameters, and more significantly reduces the number of parameters by 10 folds to keep the comparable accuracy of 0.9.

Finally, we address the impact of the projection layer $\mathcal{F}(\cdot)$. Fixed fastfood projection as in Figure 3 is applied on SuperLoRA. For 1D dense, the plot for a projection ratio of $\{1, 0.5, 0.25, 0.1, 0.01\}$ is placed from right to left in Figure 4. The classification accuracy dropped less than 1% from projection ratio 1 to 0.1 (*i.e.* 90% less parameters), but it is worse than LoRA. To get some results of projection for the number of parameters around 10^4 and 10^5, we select a few settings for SuperLoRA (2D, reshape) with $G = 1$ as shown in the figure with a marker of dark stars. Most projection results demonstrate better accuracy compared with other SuperLoRA settings without projection in the same number of parameters level. This result shows a smaller adapter with fixed projection layer is a strong functionality to improve the parameter efficiency of SuperLoRA.

We also examined another transfer learning task from ImageNet1k to CIFAR10. Most settings are same as Figure 4 for transfer learning from ImageNet21k to CIFAR100. The classifier head is frozen after selecting most relevant labels in ImageNet1k. Details are found in Appendix A.3.2. Classification results can be found in Figure 5. Even though only attention modules are adapted, overall performance is excellent, reaching an accuracy close to 0.99. Besides, SuperLoRA significantly outperforms original LoRA in terms of both classification accuracy and the parameter range it covers as the transfer learning. SuperLoRA (2D, reshape) shows at least 3-fold reduction in the required number of parameters compared to LoRA. Noticeably, when comparing the lowest-rank LoRA with around 0.97 accuracy, SuperLoRA (2D, reshape, w/ projection) improves the accuracy by about 1%, and moreover the required number of parameters can be greatly reduced by 10 folds with SuperLoRA (LoRTA: 3D, reshape) to maintain the comparable accuracy.

We confirmed the remarkable gain of our SuperLoRA on a transfer learning task for image classification with ViT models. In Appendix A.6, we further discussed the geometric analysis of SuperLoRA, and grouping impacts in Appendix A.7. In addition, We evaluated the advantage in another transfer learning task for image generation with diffusion models in Appendix A.8, Appendix A.9, Appendix A.11, and Appendix A.12.

4. Conclusion

We proposed a new unified framework called SuperLoRA, which generalizes and extends LoRA variants including LoKr and LoTR. SuperLoRA provides some extended variants, which we refer to as LoNKr and LoRTA. It offers a rich and flexible set of hyper-parameters, including the rank of factorization, the choice of projection function, projection ratio, the number of groups, the order of tensor, and the number of Kronecker splits. Through transfer learning experiments, we demonstrated that SuperLoRA achieves promising results in parameter efficiency for finetuning at low-parameter regimes. We could reduce the required number of parameters by 3 to 10 folds compared to LoRA. Future work includes studying the projection functions to further improve the efficiency in extremely-low-parameter regimes, and applications to various transfer learning tasks along with different large models such as LLMs.

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. GPT-4 technical report, 2023. 1

[2] Armen Aghajanyan, Luke Zettlemoyer, and Sonal Gupta. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. *arXiv preprint arXiv:2012.13255*, 2020. 2, 1

[3] Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023. 1

[4] Rohan Anil, Andrew M. Dai, and Orhan Firat et al. PaLM 2 technical report, 2023. 1

[5] Daniel Bershatsky, Daria Cherniuk, Talgat Daulbaev, and Ivan Oseledets. LoTR: Low tensor rank weight adaptation. *arXiv preprint arXiv:2402.01376*, 2024. 1, 2

[6] Mikołaj Bińkowski, Danica J Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. *arXiv preprint arXiv:1801.01401*, 2018. 6

[7] Shoufa Chen, Chongjian Ge, Zhan Tong, Jiangliu Wang, Yibing Song, Jue Wang, and Ping Luo. AdaptFormer: Adapting vision transformers for scalable visual recognition. *Advances in Neural Information Processing Systems*, 35:16664–16678, 2022. 1

[8] Wei Chen, Zichen Miao, and Qiang Qiu. Parameter-efficient tuning of large convolutional models. *arXiv preprint arXiv:2403.00269*, 2024. 1

[9] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 3

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020. 1, 3

[11] Ali Edalati, Marzieh Tahaei, Ivan Kobyzev, Vahid Partovi Nia, James J Clark, and Mehdi Rezagholizadeh. Krona: Parameter efficient tuning with Kronecker adapter. *arXiv preprint arXiv:2212.10650*, 2022. 1

[12] Demi Guo, Alexander M Rush, and Yoon Kim. Parameter-efficient transfer learning with diff pruning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4884–4896, 2021. 1

[13] Tianxiang Hao, Hui Chen, Yuchen Guo, and Guiguang Ding. Consolidator: Mergable adapter with group connections for visual adaptation. In *The Eleventh International Conference on Learning Representations*, 2022. 1

[14] Soufiane Hayou, Nikhil Ghosh, and Bin Yu. LoRA+: Efficient low rank adaptation of large models. *arXiv preprint arXiv:2402.12354*, 2024. 1

[15] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations*, 2021. 1

[16] Xuehai He, Chunyuan Li, Pengchuan Zhang, Jianwei Yang, and Xin Eric Wang. Parameter-efficient model adaptation for vision transformers. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 817–825, 2023. 1

[17] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. GANs trained by a two time-scale update rule converge to a local Nash equilibrium. *Advances in neural information processing systems*, 30, 2017. 6

[18] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. In *NeurIPS 2021 Workshop on Deep Generative Models and Downstream Applications*, 2021. 3, 6

[19] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[20] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *International Conference on Machine Learning*, pages 2790–2799. PMLR, 2019. 1

[21] Edward J Hu, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2021. 1, 2

[22] Shibo Jie and Zhi-Hong Deng. Fact: Factor-tuning for lightweight adaptation on vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 1060–1068, 2023. 1

[23] Shibo Jie, Haoqing Wang, and Zhi-Hong Deng. Revisiting the parameter efficiency of adapters from the perspective of precision redundancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17217–17226, 2023. 1

[24] Rabeeh Karimi Mahabadi, James Henderson, and Sebastian Ruder. Compacter: Efficient low-rank hypercomplex adapter layers. *Advances in Neural Information Processing Systems*, 34:1022–1035, 2021. 1

[25] Oscar Key, Jean Kaddour, and Pasquale Minervini. Local LoRA: Memory-efficient fine-tuning of large language models. In *Workshop on Advancing Neural Network Training: Computational Efficiency, Scalability, and Resource Optimization (WANT@ NeurIPS 2023)*, 2023. 1

[26] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 3

[27] Tuomas Kynkäänniemi, Tero Karras, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Improved precision and recall metric for assessing generative models. *Advances in Neural Information Processing Systems*, 32, 2019. 6

[28] Quoc Le, Tamás Sarlós, Alex Smola, et al. Fastfood-approximating kernel expansions in loglinear time. In *Proceedings of the international conference on machine learning*, page 8, 2013. 2, 1

[29] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86 (11):2278–2324, 1998. 6

[30] Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, 2021. 1

[31] Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4582–4597, 2021. 1

[32] Jing Liu, Toshiaki Koike-Akino, Pu Wang, Matthew Brand, Ye Wang, and Kieran Parsons. LoDA: Low-dimensional adaptation of large language models. *NeurIPS'23 Workshop on on Efficient Natural Language and Speech Processing*, 2023. 1

[33] Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A ConvNet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11976–11986, 2022. 1

[34] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, number 5, page 7. Granada, Spain, 2011. 6

[35] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *16th Conference of the European Chapter of the Associationfor Computational Linguistics, EACL 2021*, pages 487–503. Association for Computational Linguistics (ACL), 2021. 1

[36] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pages 234–241. Springer, 2015. 6

[37] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training GANs. *Advances in neural information processing systems*, 29, 2016. 6

[38] Leslie N Smith and Nicholay Topin. Super-convergence: Very fast training of neural networks using large learning rates. In *Artificial intelligence and machine learning for multi-domain operations applications*, pages 369–386. SPIE, 2019. 3

[39] Hugo Touvron, Louis Martin, and Kevin Stone et al. Llama 2: Open foundation and fine-tuned chat models, 2023. 1

[40] Anton Tsitsulin, Marina Munkhoeva, Davide Mottin, Panagiotis Karras, Alex Bronstein, Ivan Oseledets, and Emmanuel Mueller. The shape of data: Intrinsic distance for data distributions. In *International Conference on Learning Representations*, 2019. 6

[41] Ledyard R Tucker. Some mathematical notes on three-mode factor analysis. *Psychometrika*, 31(3):279–311, 1966. 2

[42] Shin-Ying Yeh, Yu-Guan Hsieh, Zhidong Gao, Bernard B W Yang, Giyeong Oh, and Yanmin Gong. Navigating text-to-image customization: From lyCORIS fine-tuning to model evaluation. In *The Twelfth International Conference on Learning Representations*, 2024. 1, 2

[43] Jiacheng Zhu, Kristjan Greenewald, Kimia Nadjahi, Haitz Sáez de Ocáriz Borde, Rickard Brüel Gabrielsson, Leshem Choshen, Marzyeh Ghassemi, Mikhail Yurochkin, and Justin Solomon. Asymmetry in low-rank adapters of foundation models. *arXiv preprint arXiv:2402.16842*, 2024. 1