RANDLORA: FULL RANK PARAMETER-EFFICIENT FINE-TUNING OF LARGE MODELS

Anonymous authors
Paper under double-blind review

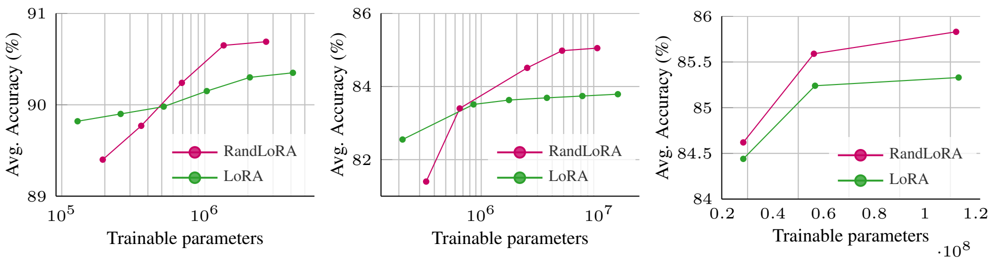
ABSTRACT

Low-Rank Adaptation (LoRA) and its variants have shown impressive results in reducing the number of trainable parameters and memory requirements of large transformer networks while maintaining fine-tuning performance. However, the low-rank nature of the weight update inherently limits the representation power of the fine-tuned model, potentially compromising performance on complex tasks. This raises a critical question: when a performance gap between LoRA and standard fine-tuning is observed, is it due to the reduced number of trainable parameters or the rank deficiency? This paper aims to answer this question by introducing RandLoRA, a parameter-efficient method that performs full-rank updates using a learned linear combinations of low-rank, non-trainable random matrices. Our method limits the number of trainable parameters by restricting optimization to diagonal scaling matrices applied to the fixed random matrices. This allows us to effectively overcome low-rank limitations while maintaining low parameter count and memory usage during training. Through extensive experimentation across vision, language, and vision-language benchmarks, we systematically evaluate the limitations of LoRA and existing random basis methods. Our findings reveal that full-rank updates are beneficial across vision and language tasks separately, but especially so for vision-language tasks, where RandLoRA significantly reduces—and sometimes eliminates—the performance gap between standard fine-tuning and LoRA, demonstrating its efficacy.

1 INTRODUCTION

The emergence of large pre-trained models has significantly enhanced the generalization capabilities of neural networks, demonstrating remarkable versatility across a broad range of tasks. However, a higher parameter count also leads to a significant increase in the computational resources required for fine-tuning on downstream tasks. To mitigate this issue, parameter-efficient fine-tuning (PEFT) approaches such as low-rank adaptation (LoRA) (Hu et al., 2022), draw inspiration from the low intrinsic dimensionality of pre-trained models (Li et al., 2018; Aghajanyan et al., 2021) and characterize the weight update as the product of two low-rank matrices, substantially reducing the trainable parameter count and memory requirements during training. This formulation allows for an adjustable number of trainable parameters by modifying the rank of the matrices, providing great flexibility under various resource constraints.

In spite of the strong performance of LoRAs in parameter-efficient settings, our investigation uncovers an accuracy plateau, where increases in trainable parameters by increased ranks fail to bridge the accuracy gap with standard fine-tuning. These undesirable scaling properties (Kopiczko et al., 2024) raise questions about the inherent limitations imposed by the low-rank structure of LoRA, particularly when tackling complex tasks that necessitate larger parameter budgets. This issue would ideally be addressed by introducing full-rank updates while maintaining the parameter-efficiency. To this end, we propose RandLoRA, a PEFT method that leverages multiple, linearly-independent random bases in the form of non-trainable low-rank matrices. By solely learning scaling coefficients for the linear combination of the random low-rank bases, our method achieves full-rank updates, while maintaining low memory usage. As a result, RandLoRA strikes a balance between parameter efficiency and full-rank updates, allowing for more flexible and effective fine-tuning.



(a) DinoV2 (b) CLIP (c) LLama3-8B

Figure 1: LoRA becomes limited by the rank of its update. We train DinoV2 and CLIP to classify 21 image datasets and LLama3-8B to solve 8 commonsense reasoning tasks.

Through extensive experimentation, we empirically demonstrate the low-rank limitations of LoRA, particularly on vision-language tasks, and demonstrate how RandLoRA can improve performance without parameter increases. Figure 1 summarizes our findings across pure vision (DinoV2), vision-language (CLIP) and commonsense reasoning (LLama3-8B) where increasing LoRA's parameter count has highly diminishing returns. We find that RandLoRA outperforms LoRA as the parameter budget expands, while remaining parameter efficient thanks to its full-rank update strategy. We conclude our investigation with an insightful discussion on the distinctive characteristics of RandLoRA where our analysis reveals that, in contrast to LoRA, RandLoRA yields activation patterns in deeper layers that closely align with those obtained through full fine-tuning. Furthermore, our examination of the loss landscape reports that the local minima reached by RandLoRA is often well connected with the local minima reached by standard fine-tuning, and that it always results in a lower loss minima than LoRA for an equal parameter count. Additionally, we explore the integration of sparse random bases, where initial findings highlight that sparse bases preserves the performance of RandLoRA. This suggests promising avenues to further reduce memory and computational requirements when training large transformer architectures, without compromising model performance.

Our contributions are summarized as:

1. We investigate the interplay between rank and number of trainable parameters when fine-tuning large pre-trained models, highlighting the limitations of LoRA in improving performance when larger ranks are required.
2. We propose RandLoRA, a novel parameter-efficient fine-tuning (PEFT) strategy based on random basis combinations, enabling full-rank updates without memory overhead over LoRA.
3. We rigorously assess RandLoRA across diverse pre-trained architectures and tasks, spanning pure vision and vision-language image classification to commonsense reasoning, demonstrating its versatility and effectiveness.

2 Related work

2.1 Low Rank Adaptation of Large Models

Low Rank Adaptation (LoRA) of large language models has revolutionized the fine-tuning paradigm, enabling memory-constrained adaptation to specialist tasks and democratizing access to larger models. Initially introduced by (Hu et al., 2022), LoRA leverages the observation that weight updates during fine-tuning can converge to suitable performances without necessitating full rank updates. By factorizing weight updates into the product of two low rank matrices, LoRA achieves a memory-efficient solution for adapting large models. Moreover, once the low rank matrices are merged into the original weight matrix size, no latency is present during inference. Several improvements have been proposed to build upon LoRA's success. Weight-decomposed LoRAs (DoRA) (Liu et al., 2024) proposes to improve convergence by decomposing LoRA updates into magnitude and direction components. AdaLoRA (Zhang et al., 2023) and AutoLoRA (Zhang et al., 2024c), utilize specialized metrics or meta-learning to propose rank-adapted LoRA formulations that dynamically adjust the rank to suit every layer's need. Other improvements include initialization strategies for the low rank matrices using the truncated SVD of the pretrained weights and where the whole decomposition is finetuned as in Pissa (Meng et al., 2024) or where only the singular value matrix is as in SVFT (Lingam et al., 2024) or LoRA-XS (Bałazy et al., 2024). Further improvements are proposed in HydraLoRA (Tian et al., 2024) where the scaling-up matrix of the low rank decomposition is split into multiple ones with a routing layer added to select the contribution of each head. This formulation enhances multi-task learning at the cost of losing the merging capabilities of LoRA in the pretrained weight at test-time. These advancements collectively enhance the efficiency of LoRA, solidifying its position as a cornerstone of large language model fine-tuning.

2.2 Parameter-Efficient fine-tuning (PEFT) through Random Bases

Recent research has focused on further reducing the trainable parameter count of LoRA, a crucial aspect for low-shot applications where minimizing trainable parameters can prevent overfitting and enhance generalization. A promising direction involves utilizing random bases combinations, where randomly generated matrices are combined using a limited number of trainable parameters to estimate a weight update.

PRANC (Nooralinejad et al., 2023) pioneered the random base strategy by learning a weighted averaged of random matrices through back-propagation. PRANC's solution averages multiple full size weight matrices for each layer, leading to high memory consumption. To address this, the authors generate random bases on the fly during forward and backward passes using a fixed seed random number generator, reducing memory usage to that of the largest trained layer in the network at the cost of training latency.

Building upon PRANC, NOLA (Koohpayegani et al., 2024) introduces an improved algorithm where random bases are estimated as the product of two low-rank random matrices, each weighed using a learnable scalar and summed before matrix multiplication. This approach effectively approximates a rank 1 LoRA with significantly fewer trainable parameters and largely reduces memory consumption during training over PRANC.

Concurrently, VeRA (Kopiczko et al., 2024) proposed an alternative strategy utilizing a single high-rank random matrix (typically 256 or 1024), instead of summing multiple rank 1 matrices as in NoLA. VeRA also employs a scaling strategy of random bases distinct from NoLA, detailed in section 4, which relates to our approach. Both NOLA and VeRA achieve comparable performance to LoRA in few-shot fine-tuning scenarios while training substantially fewer parameters.

2.3 Alternative strategies for parameter-efficient fine-tuning

We report orthogonal alternatives to weight tuning for parameter-efficient adaptation of large transformer models, with one direction specifically targeting prompt tuning. Context Optimization (CoOP) proposes to append learnable representations to learn context surrounding the textual class name embedding of CLIP (Radford et al., 2021). These learnable prompts where later generalized in Conditional Context Optimization (CoCoOP) (Zhou et al., 2022) to be instance specific by adding a lightweight meta-network in charge of predicting image-specific context. More recently, prompt tuning approaches have focused on preserving the initial knowledge in the foundation model to improve performance. Decoupled Prompt Tuning (DePT) (Zhang et al., 2024b) identifies and isolates shared subspaces during prompt optimization to promote the retention of shared knowledge and avoid catastrophic forgetting and Prompting with Self-regulating Constraints (PromptSRC) (Khattak et al., 2023b) regularize the learned prompt to remain near the initial embeddings. Although highly parameter-efficient, prompt tuning algorithms can struggle to generalize past few-shot settings (Han et al., 2024) and LoRA has previously been shown to be a stronger alternative as the number of shots increases (Zanella & Ben Ayed, 2024). We suggest here that prompt tuning should be considered as an orthogonal optimization to parameter-efficient weight-tuning and leave them out of the comparison with RandLoRA.

3 MOTIVATIONS

Our literature review reveals that research on improving LoRA is focused on reducing the number of trainable parameters further, either through adaptable ranks or by using fixed or shared low rank projection matrices. When looking at moderate to larger parameter budgets however LoRA remains highly competitive.

We identify that early research has convincingly demonstrated the promise of random basis combinations as a parameter-efficient strategy for large models, particularly in few-shot scenarios. Two approaches have emerged, each representing a distinct paradigm. VeRA advocates for a unique random base with large rank, while NoLA proposes to average a large number of random bases with small ranks. Both approaches report performance comparable to LoRA in few-shot scenarios while converging on a significantly reduced number of trainable parameters. However, as we will demonstrate, this reduction comes at the cost of limited performance when venturing beyond few-shot learning, limiting the scalability of these algorithms.

Finally, we report that LoRA is predicated on the assumption that low-rank updates suffice for fine-tuning large models. We aim in this paper to question the universality of this hypothesis, exploring scenarios where full rank alternatives may be necessary. The fundamental question follows: is parameter efficiency achieved through low-rank approximation limited by (1) the low-rank nature of the update or (2) by the low parameter count. Can parameter-efficient full rank updates provide a more accurate solution ? This paper aims to address these questions, exploring the balance between parameter efficiency and low-rank fine-tuning of large transformer models, and shedding light on the limitations of existing approaches.

4 RANDLORA—PARAMETER-EFFICIENT FINE-TUNING WITH FULL RANK

4.1 WEIGHT UPDATES AS A SUM OF LOW-RANK MATRICES

Let $W_0 \in \mathbb{R}^{D\times d}$ be a weight matrix of a large pre-trained model. Fine-tuning aims to find an appropriate $\Delta W \in \mathbb{R}^{D\times d}$, such that the fine-tuned weights $W_0 + \Delta W$ lead to an adapted model, tailored to a specific downstream task. Without loss of generality, let us assume $d < D$. The motivation behind RandLoRA stems from the singular value decomposition (SVD) of ΔW, i.e., $\Delta W = U\Sigma V^\mathsf{T}$, where $U \in \mathbb{R}^{D\times d}$, $\Sigma \in \mathbb{R}^{d\times d}$, $V \in \mathbb{R}^{d\times d}$. This decomposition can be written as the sum of the product of rank-one matrices, as follows

$$\Delta W = \sum_{i=1}^{d} \mathbf{u}_i \sigma_i \mathbf{v}_i^\mathsf{T}, \tag{1}$$

where $\mathbf{u}_i$ and $\mathbf{v}_i$ denote the columns of U and V, respectively. We suggest that in this context, low-rank updates such as LoRAs can be characterized as an approximation of the few largest singular values while the rest of the information in ΔW being discarded. To better illustrate this point, let us denote the rank of LoRA by r and for brevity of exposition, assume d is divisible by r. We rewrite equation 1 as a sum of the product of rank-r matrices, as follows

$$\Delta W = \sum_{j=1}^{n} U_j \Sigma_j V_j^\mathsf{T}, \tag{2}$$

where $U_j \Sigma_j V_j^\mathsf{T} = \sum_{i=rj}^{r(j+1)} \mathbf{u}_i \sigma_i \mathbf{v}_i^\mathsf{T}$ and where $n = d/r$. This formulation reveals how LoRA models the approximates the first low-rank partition $U_1 \Sigma_1 V_1^\mathsf{T}$, and implicitly assumes $\sum_{j=2}^{n} U_j \Sigma_j V_j^\mathsf{T} \approx 0$. We however argue that the remaining $n-1$ terms can play a crucial role when capturing more complex task-specific variations that require larger deviations from the pre-trained weight W_0.

4.2 PARAMETER-EFFICIENT APPROXIMATION OF LOW-RANK MATRICES

Approximating more terms in the decomposition of ΔW using LoRA's formulation quickly becomes parameter inefficient, culminating to $Dd + d^2$ parameters for a full rank d in place of the original Dd parameters of ΔW. To perform full-rank updates while maintaining parameter-efficiency, we propose instead to approximate each term of ΔW in equation 2 using low-rank random bases where only scaling coefficients are learned,

$$\Delta W = \sum_{j=1}^{n} B_j \Lambda_j A_j \Gamma_j, \tag{3}$$

where $B_j \in \mathbb{R}^{D\times r}$ and $A_j \in \mathbb{R}^{r\times d}$ are non-trainable, random matrices. The two learnable diagonal scaling matrices, $\Lambda_j \in \mathbb{R}^{r\times r}$ and $\Gamma_j \in \mathbb{R}^{d\times d}$ are unique to each of the n terms and fulfill complementary roles to improve the approximation. We aim for $A_j\Gamma_j$ transform the input features into an low-dimensional space (rank-r), Λ_j to scale the compressed features which are then transformed back into the desired output space by B_j.[1] Since Γ_j operates on the column space of A_j and is unique to each A_j, we use a unique shared matrix $A \in \mathbb{R}^{r\times d}$ across all n terms without loss of expressivity but reducing memory consumption. With a shared A, we formulate the update as

$$\Delta W = \sum_{j=1}^{n} B_j \Lambda_j A \Gamma_j. \tag{4}$$

To achieve a full-rank update, we set $n = d/r$, leading to $\frac{d}{r}(d+r) = d^2/r + d$ learnable parameters. Note that unlike LoRA, the number of learnable parameters is inversely proportional to the rank of the random bases in RandLoRA, as increasing the rank of the bases leads to a reduction in trainable parameters while maintaining full rank. In summary, RandLoRA trades-off approximation accuracy for scope, sacrificing a more precise representation of the individual SVD elements of ΔW to capture a larger portion of its singular value decomposition.

4.3 Convergence analysis

In this section, we present a theorem showing that weight updates using RandLoRA is an accurate approximation of general matrices under certain theoretical conditions.

Theorem 4.1. *Let W be a fixed $D \times d$ matrix, with $D > d$ and rank$(W) = d$. Fix $1 \leq n \leq d$, such that $d = nr$. The matrix W can be factorized using SVD as*

$$W = \sum_{j}^{n} U_j \Sigma_j V_j^{\mathsf{T}}, \tag{5}$$

where $U_j \in \mathbb{R}^{D\times r}$, $V_j \in \mathbb{R}^{r\times d}$ are partitions of the left and right singular vectors, and $\Sigma_j \in \mathbb{R}^{r\times r}$ contains r singular values. For each $1 \leq j \leq n$, let B_j denote a random $D \times r$ matrix whose entries are drawn i.i.d from either a Gaussian or uniform distribution, A_j denotes an $r \times d$ matrix whose entries are drawn similarly, Λ_j is a diagonal $r \times r$ matrix and Γ_j is a diagonal $d \times d$ matrix drawn similarly. Assume

$$\|U_j \Sigma_j V_j^{\mathsf{T}} - B_j \Lambda_j A_j \Gamma_j\|_F \leq \epsilon \tag{6}$$

for each $1 \leq j \leq n$ for some $0 < \epsilon$. Then we have that with probability 1 that each $B_j\Lambda_j A_j\Gamma_j$ has full rank and

$$\|W - \sum_{j=1}^{n} B_j \Lambda_j A_j \Gamma_j\|_F \leq n \cdot \epsilon. \tag{7}$$

For details on the proof of theorem 4.1 please refer to appendix D.1.

Theorem 4.1 is premised on $B_j\Lambda_j A_j\Gamma_j$ being a good approximation for the r-truncated singular value of ΔW, which is shown to be true empirically in VeRA (Kopiczko et al., 2024) for example. We show in this case that ΔW can be accurately approximated as $\sum_{j=1}^{n} B_j\Lambda_j A_j\Gamma_j$, motivating RandLoRA's formulation. In contrast, since the best approximation a rank-r LoRA can achieve is the r-truncated SVD of W, then by Eckart-Young-Mirsky theorem, the Frobenius norm of the difference between W and low-rank adaptation BA is lower bounded as follows

[1] The formulation of our method is similar to that of VeRA (Kopiczko et al., 2024), which will be discussed in detail in section 6.5.

$$\|W - BA\|_F \geq \left\| W - \sum_{i=1}^{r} \mathbf{u}_i \sigma_i \mathbf{v}_i^\mathsf{T} \right\|_F = \sum_{i=r+1}^{d} \sigma_i^2. \tag{8}$$

We conclude that while LoRA's rank r approximation is limited by the sum of the last $d - r - 1$ squared singular values of W, RandLoRA does not present this low bound and is only limited by how close (ϵ) can $B_j \Lambda_j A_j \Gamma_j$ approximate length-r segments of the SVD of W.

5 Experiments

5.1 Experimental Settings

We conduct a comprehensive comparison with three state-of-the-art approaches: LoRA (Hu et al., 2022), NoLA (Koohpayegani et al., 2024), and VeRA (Kopiczko et al., 2024). We perform a hyperparameter search to identify optimal settings for LoRA, NoLA, VeRA, and RandLoRA to ensure a fair comparison. More details about the experimental settings can be found in appendix C. Additional experiments on the General Language Understanding Evaluation (GLUE) (Wang et al., 2019) and End-to-end (E2E) Novikova et al. (2017) natural language generation benchmarks as well as further comparison with prompt-tuning algorithms are available in appendix B.

5.2 Vision: DinoV2 and CLIP's vision backbone

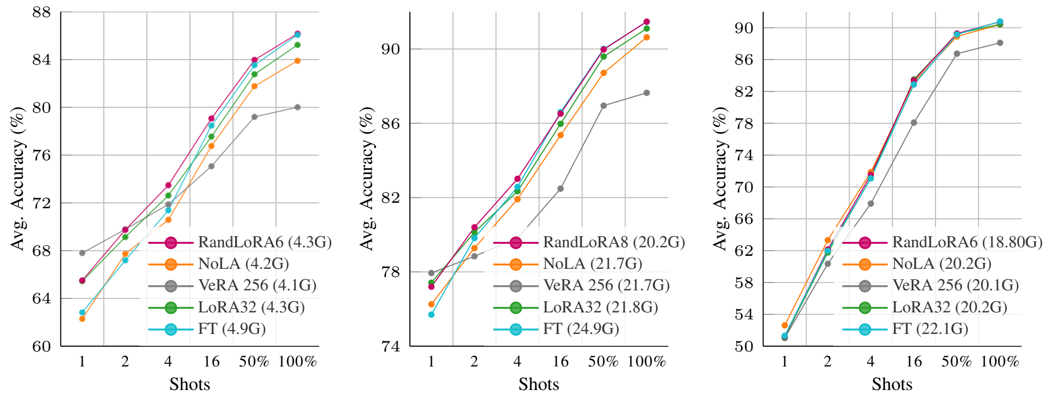
We first study fine-tuning vision backbones to perform image classification. We utilize the pretrained ViT-B/14 DinoV2 (Oquab et al., 2023) and ViT-B/32, ViT-L/14 CLIP (Radford et al., 2021) vision backbones as a strong self-supervised baselines. We fine-tune on 21 vision datasets (see Table 7 in Appendix C.1 for details) and evaluate performance on $\{1, 2, 4, 16\}$-shot learning tasks, as well as with 50% and 100% of the training data. We compare with LoRA with rank 32 as a strong parameter-efficient baseline in addition to VeRA and NoLA. For RandLoRA, we choose the rank of the random bases denoted as RandLoRA-r to ensure the closest amount trainable parameters to LoRA. We train the parameter efficient algorithms on the feature extractor and concurrently learn a linear classifier for DinoV2 or use frozen language embeddings with CLIP.

We observe that although a gap exists between LoRA and standard fine-tuning with CLIP, this gap is much smaller for DinoV2. In any case we observe that given an equal amount of trainable parameters, RandLoRA improves over LoRA's performance to bridge the gap with standard fine-tuning (FT). We believe that the success of LoRA for the DinoV2 backbone is explained with the training objective and discuss this matter further in section 6.1. A table with detailed results can be found in appendix E.2. By improving over LoRA with an equal amount of trainable parameters, RandLoRA show that LoRA can indeed be limited by its rank and that full-rank updates improve results to equate fine-tuning performance. We also report that VeRA and NoLA, although very efficient in few-shot scenarios quickly become limited by the low amount of parameter trained when the amount of data increases. VeRA in particular struggles to scale which indicates that a strong low-rank update as in NoLA is preferable to an approximated larger rank one in our image classification scenario.

5.3 Vision-Language: CLIP

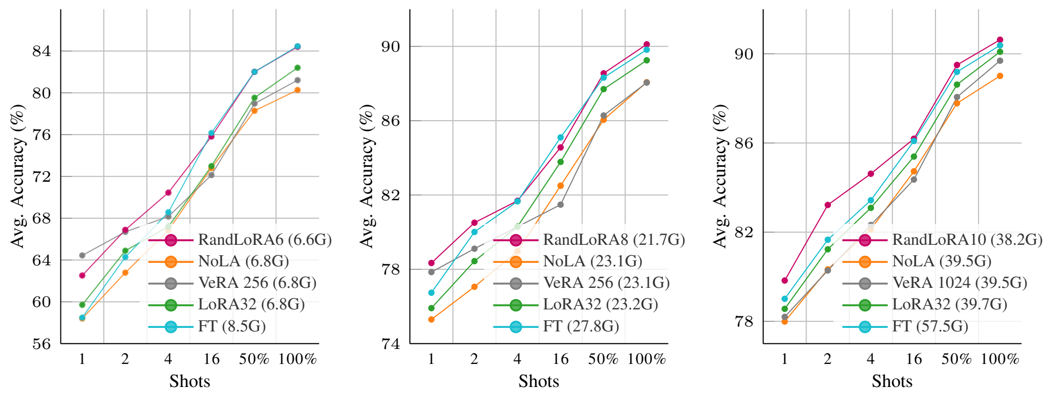
We extend in this section our experimental setting to fine-tuning CLIP-like transformer architectures on classification datasets where contrary to section 5.2 both the language and vision encoders of CLIP are trained. We add ImageNet (Krizhevsky et al., 2012) to the dataset pool to scale up to 22 classification datasets. To assess the effectiveness of RandLoRA compared to LoRA on models of varying sizes, we consider three variants of pre-trained CLIPs from the open-clip repository (Cherti et al., 2023): ViT-B/32 (151M parameters), ViT-L/14 (428M parameters) and ViT-H/14 (1B parameters). We scale the rank of the random bases in RandLoRA in the same way as section 5.2 to maintain a number of parameters comparable to a rank 32 LoRA: RandLoRA-$\{6,8,10\}$ for ViT-$\{$B/32,L/14,H/14$\}$ respectively.

A summary of results is available in Figure 3 with detailed results being available in appendix E.1. Because fine-tuning vision-language architectures such as CLIP is a harder optimization problem,



(a) ViT-B/32 (b) ViT-L/14 (c) DinoV2

Figure 2: Tuning CLIP of DinoV2 vision encoder for image classification. Accuracy averaged over 21 datasets. We additionally report max GPU VRAM usage during training.



(a) ViT-B/32 (b) ViT-L/14 (c) ViT-H/14

Figure 3: Tuning CLIP's vision and language encoders for image classification. Accuracy averaged over 22 datasets. We additionally report max GPU VRAM usage during training.

we observe the existence of a larger performance gap between full fine-tuning and LoRA than for pure vision, which we confirm is not bridged by increasing the rank of LoRA (see Figure 1). This suggests that increasing parameter count is not enough, pointing towards the rank of the update as the possible limit to the performance of LoRA. When running RandLoRA with the same amount of trainable parameters, we observe that the gap with fine-tuning is bridged. When compared with NoLA and VeRA we come to the same conclusions as section 5.2 although VeRA is this time much more competitive for larger data budgets, hinting towards the importance of high ranks for finetuning CLIP-like vision language architectures. We also report that our base sharing strategy allows RandLoRA to decrease VRAM usage over LoRA which can be relevant for large architectures such as ViT-H/14.

5.4 Commonsense reasoning

We finally explore fine-tuning large language models for 8 commonsense reasoning tasks, see appendix C.4 for details. We use open-source pretrained architectures including the 0.5B parameter variant of Qwen2 (Yang et al., 2024), 3B variant of Phi3 (Abdin et al., 2024), and 8B configuration of LLama3 (Dubey et al., 2024). To assess data efficiency, we investigate two training scenarios: utilizing the full 170k training samples and a 15k subset, as introduced by Hu et al. (2023). Our evaluation, presented in Table 1, compares the performance of LoRA, VeRA, NoLA, and RandLoRA. We study varying ranks of LoRA to test rank limits and scale RandLoRA fairly where variants of

Table 1: Parameter-efficient fine-tuning of Large Language Models (LLMs). Results averaged over 8 commonsense reasoning tasks. RandLoRA is abbreviated as RL. We bold the best accuracy between parameter-equivalent RandLoRA and LoRA couples.

Network	Size	ZeroShot	NoLA	VeRA	LoRA-16	RL-10	LoRA-32	RL-6	–	–
Qwen2-0.5b	15k	5.2	42.6	48.1	53.2	**53.5**	52.3	**52.9**	–	–
	170k	5.2	47.4	51.8	57.4	**57.7**	57.3	**57.9**	–	–
						RL-40		RL-20		RL-10
Phi3-3b	15k	65.4	80.4	78.6	**81.8**	81.7	80.3	**82.3**	81.4	**82.4**
	170k	65.4	82.3	81.4	84.6	**84.7**	85.0	**85.2**	84.2	**85.0**
						RL-60		RL-30		RL-15
LLama3-8b	15k	27.0	76.9	77.1	**82.7**	81.0	**83.1**	81.3	81.2	**82.0**
	170k	27.0	81.2	81.7	84.4	**84.6**	85.2	**85.6**	85.3	**85.8**

Table 2: Further comparison with related methods on LLama3-8b. Results averaged over 8 commonsense reasoning tasks. We bold the best accuracy.

Method	15k	170k	Method	15k	170k
LoRA-16	82.7	84.4	LoRA-32	**83.1**	85.2
DoRA-16	**82.8**	84.3	DoRA-32	82.5	85.2
RandLoRA-60	81.0	**84.6**	RandLoRA-30	81.3	**85.6**

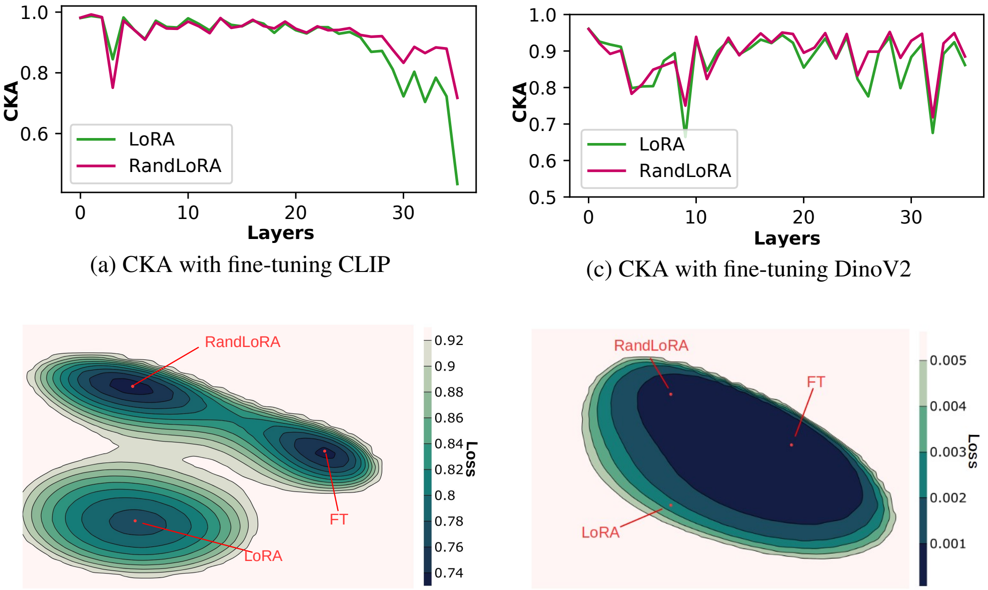
RandLoRA end up with the closest amount of trainable parameters to LoRA-{16,32,64} respectively. We do not run LoRA-64 for Qwen2 due to the smaller size of the model. We report that RandLoRA compares favorably to LoRA, outperforming it in some cases. We observe Phi3's zero-shot model exhibits good commonsense reasoning capabilities, enabling VeRA and NoLA to achieve good results despite the limited number of trainable parameters. Conversely, Qwen2 and LLama3 require more complex adaptations from their zero-shot weights, posing challenges for VeRA and NoLA to bridge the performance gap with LoRA. We notice that the 15k training setting can lead to overfitting for larger LoRA ranks and for RandLoRA, resulting in reduced performance. This is in spite of added dropout in each adapted layer and early stopping. For the larger 170k training samples subset, RandLoRA systematically improves over LoRA with the absolute accuracy gains of RandLoRA over LoRA generally increasing with larger parameter count. This result solidifies that when finetuning LLMs, LoRA can become limited by the rank of the update as the parameters budget increases whereas RandLoRA can make use of this extra budget to train more accurate models. We further propose to compare RandLoRA with DoRA (Liu et al., 2024), a recent alternative to LoRA using a decomposed magnitude/direction strategy built on top of LoRA. Results are presented in Table 2 where we observe that DoRA performs better for small parameter budgets but RandLoRA outperforms both LoRA and DoRA for larger ones. We conclude that RandLoRA presents a good alternative to LoRA and DoRA for fine-tuning LLMs of all sizes, offering competitive performance and definite advantages for larger parameter budgets.

6 Discussion

6.1 Similarities with fine-tuning: activations

We conduct an evaluation of activation similarities to assess the efficacy of LoRA and RandLoRA in replicating the activation patterns of a fully fine-tuned model. To this end, we employ the Centered Kernel Alignment (CKA) (Kornblith et al., 2019) metric to quantify the similarity between activations produced by each method and those of the fine-tuned model. This experimental design aims to assess the ability of LoRA and RandLoRA to capture dataset-specific complexities encoded in activation patterns. Figure 4a presents the CKA scores for successive self-attention and MLP layers in the CLIP and DinoV2 vision backbones, averaged over 5 datasets where RandLoRA provides the best improvements. Our results reveal that for the CLIP backbones, LoRA's CKA decreases in deeper layers, struggling to maintain alignment with fine-tuned activations. In contrast, RandLoRA

Figure 4: How close do RandLoRA and LoRA get to standard fine-tuning ? We compare CKA scores of RandLoRA and LoRA with finetuned activations (top) and the mode connectivity in the loss landscape of UCF101 (bottom)



(a) CKA with fine-tuning CLIP

(c) CKA with fine-tuning DinoV2

(b) Loss landscape CLIP

(d) Loss landscape DinoV2

with equal numbers of parameters matches the alignment of LoRA for earlier layers then improves it for the deeper layers. This phenomenon is however absent for the DinoV2 results where we observe no significant CKA decrease for LoRA in the latter layers, explaining the almost non-existant accuracy gap with standard fine-tuning for this pre-trained backbone. This disparity likely stems from fundamental differences in CLIP and DinoV2's training objectives. We suggest that DinoV2's purely visual objective yields features inherently optimized for classification, necessitating minimal adjustments in weight space direction and thus the low rank of LoRA suffices. In contrast, CLIP's multimodal vision-language objective demands higher ranks to effectively adapt to pure vision tasks.

6.2 Similarities with fine-tuning: loss landscape

We investigate the mode connectivity of models trained using standard fine-tuning, LoRA, and RandLoRA. To visualize the loss landscape, we construct a 2D plane with LoRA, RandLoRA, and fine-tuning models positioned at (0,0), (1,0), and (0.5,1), respectively. By solving for the interpolation coefficients $\alpha_1, \alpha_2, \alpha_3$ at each point (x, y) under the constraint $\sum_{i=1}^{3} \alpha_i = 1$, we evaluate the weight-space interpolated model on a 5% subset of the training set to compute the average loss. Figure 4b reveals that when fine-tuning CLIP and despite training the same amount of parameters, RandLoRA produces a deeper minima than LoRA, often presenting a low-loss bridge with the standard fine-tuning optimum. In the case of DinoV2, all optimums leave in the same low loss basin with LoRA already being very close to standard fine-tuning which illustrates the small performance gap between LoRA and standard fine-tuning. These insights further highlight that LoRA's rank limits for complex tasks. In all cases, RandLoRA with an equal amount of trainable parameters but full rank updates achieves a deeper minima than LoRA. 3D visualizations for 2 more datasets are available in appendix A.

6.3 Further studies on full vs low rank fine-tuning of CLIP

A further point of interest we briefly address here is whether RandLoRA performs better than LoRA on CLIP because it is a better approximation of the truncated SVD of ΔW than LoRA is or if it is indeed the full rank capabilities that allow for improved generalization. We perform an ablation

Table 3: Ablation on the rank of the updates. The same amount of trainable parameters is used in all methods.

Method	Rank	Accuracy
LoRA	32	83.74
RandLoRA-a	32	83.62
RandLoRA-b	384	85.32
RandLoRA-6	768	85.98

Table 4: Fine-tuning CLIP or LLama3 using RandLoRA different random distributions or base sparsity.

Model	Sparsity	Accuracy
CLIP-ViT-B/32 - uniform	0%	85.98
CLIP-ViT-B/32 - normal	0%	85.61
CLIP-ViT-B/32 - binary	0%	85.52
CLIP-ViT-B/32	66%	85.43
CLIP-ViT-B/32	93%	85.57
CLIP-ViT-B/32	98%	84.35
CLIP-ViT-B/32	99%	83.34
LLama3-8b	0%	85.59
LLama3-8b	66%	85.42

study of RandLoRA where we create two variants. The first variant RandLoRA-a restricts the rank of ΔW to r by averaging the matrices before multiplication: $\Delta W = \left(\sum_{i=1}^{N} B_i \Lambda_i\right)\left(\sum_{i=1}^{N} A_i \Gamma_i\right)$. The second variant RandLoRA-b fixes N to rank$(\Delta W)/r/2$ so that the update is half rank, and setting the rank of the random bases so that the amount of training parameters remains the same as RandLoRA-r. All variants thus train the same amount of parameters, only the rank of the update changes. We report accuracy results when training on 100% of the 22 datasets for the ViT-B/32 architecture of CLIP in Table 3. We observe that with an equal number of trainable parameters, the larger the rank of the update the better the results. These results comfort our conclusions on the importance of large rank updates, especially when fine-tuning CLIP architectures.

6.4 Sparse random matrices

Section 4.3 proves that RandLoRA provides a bounded approximation of W given any random matrices B_i and A_i drawn from a probability distribution whose defined measure is absolutely continuous with respect to the Lebesgue measure. This prompts further questions about possible sparse or memory-efficient random matrices respecting theorem 4.1. We establish a direct link with literature on random projections which has proposed sparse random matrices constructed to satisfy the Lindenstrauss & Johnson (1984) lemma on distance-preserving embeddings. We specifically experiment with the sparse random matrix construction proposed in Bingham & Mannila (2001) , later extended in Li et al. (2006), where elements of B_i and A_i are assigned as -1 with probability $\frac{1}{s}$, 0 with probability $1 - \frac{2}{s}$ and 1 with probability $\frac{1}{s}$ with s an integer in $[2, \sqrt{D}]$ for $W \in \mathbb{R}^{D \times d}$.We then normalize these matrices to preserve vectors of unit length. We discuss wether these bases preserve the full rank constraint in appendix C.6. Table 4 reports early results achieved when using random bases constructed in this way as part of RandLoRA's update. We study sparsity ratios $s = 2$ to create binary matrices and $s \in \{6, \sqrt{D}, 100, 200\}$ to construct $\{66\%, 93\%, 98\%, 99\%\}$ sparse matrices where $\sqrt{D}$ is the maximum sparsity recommended in Li et al. (2006). Remarkably, when following the recommendations of Li et al. (2006) we observe performances comparable non-sparse matrices, while theoretically largely reducing memory requirements and compute needs. Larger sparsity ratios however lead to accuracy degradations. These observations open possibilities for further improvements of RandLoRA using compute-optimized random bases.

6.5 Summary of differences with related random bases algorithms

Preceding research has explored learning linear combinations of random bases for parameter efficient fine-tuning, with two prominent studies VeRA (Kopiczko et al., 2024) and NoLA (Koohpayegani et al., 2024) warranting particular attention. We first note that a key distinction between our proposed method, RandLoRA, and existing approaches lies in their objectives. Whereas VeRA and NoLA, focus on approximating a Low-Rank Adaptation (LoRA) of W using a further reduced amount of parameters, RandLoRA strives to approximate the full-rank weight update. Specifically, VeRA employs a similar decomposition, $W = \sum_{i=1}^{N} U_i \Sigma_i V_i$, but only approximates the first block as $B_0 \Lambda_0 A_0 \Gamma_0$, yielding a parameter-efficient yet low-rank update. In contrast, RandLoRA approximates all N blocks, effectively providing a full-rank update for W. Where VeRA and NoLA propose a hyper-parameter efficient method for applications content with a low-rank update, we aim to address failure cases through full rank updates. Moreover, our formulation, as expressed in equation equation 4, offers flexibility in trainable parameter selection, spanning from VeRA's parameter count for $r = \text{rank}(W)$ to the full parameter count of W for $r = 1$. This adaptability enables RandLoRA to generalize better and achieve fine-tuning accuracy comparable to standard fine-tuning, particularly with the amount of training data increases, as reported in Section 5.

6.6 Limitations

Despite the efficacy of RandLoRA, we identify three key limitations that would warrant subsequent investigations.

A key limitation of RandLoRA lies in the additional computational overhead incurred by weight update calculations. This results in notable increases in training time for larger models, as quantified in Appendix C.6.1. However because RandLoRA's weight update is valid for any linearly independent random bases there exists opportunities for optimization. Specifically, future enhancements would focus on implementing matmul-free matrix combinations as an efficient use of the ternary sparse random bases. Indeed, an efficient implementation would simplify the matrix product of B by $\Lambda A \Gamma$ to simple aggregations, eliminating floating-point arithmetic (Li et al., 2006). Although CUDA kernels for such operations are currently unavailable (Zhu et al., 2024), their future development would significantly accelerate RandLoRA training and reduce compute.

Another avenue for future exploration lies in investigating the existence of non-random, optimal bases B_i and A that could provide more effective directions reducing ϵ in equation equation 6 and leading to accelerated or improved convergence. The discovery of such bases, potentially through large-scale experimental searches or analytical derivations from pre-trained models, could significantly enhance the efficiency of RandLoRA. Elucidating the properties and construction of these optimal bases presents a compelling research direction, warranting further investigation.

Finally, hybrid solutions that synergistically combine the strengths of LoRA and RandLoRA could be developed. Specifically, LoRA could be leveraged to accurately estimate the most critical components of the SVD of W, while RandLoRA would capture the remaining spectral information in a parameter-efficient manner. However, designing such a hybrid algorithm poses significant challenges due to the disparate training objectives. A viable starting point could be to utilize RandLoRA to complement an existing LoRA representation that has not achieved satisfactory convergence on a training task. By addressing these limitations, future research can further refine RandLoRA and push its potential for efficient full-rank fine-tuning.

7 Conclusion

This paper introduces RandLoRA, a method achieving parameter efficiency and low memory cost while enabling full rank model updates. Our findings underscore the critical importance of full-rank updates when fine-tuning pre-trained architectures and we observe that our approach surpasses LoRA's performance for an equal parameter count, highlighting the value of full-rank updates in large model fine-tuning. Through extensive experiments across diverse tasks we demonstrated the efficacy of our method. While RandLoRA incurs additional computational overhead due to random basis multiplications, memory consumption remains contained and we provide venues for reducing this compute in practice. As a results, RandLoRA offers a viable alternative to LoRA for fine-tuning large pre-trained models on consumer-grade hardware. Our results have significant implications for efficient and effective model adaptation, prompting for future research in scalable and versatile full-rank fine-tuning techniques.

References

Marah Abdin, Sam Ade Jacobs, Ammar Ahmad Awan, Jyoti Aneja, Ahmed Awadallah, Hany Awadalla, Nguyen Bach, Amit Bahree, Arash Bakhtiari, Harkirat Behl, et al. Phi-3 technical report: A highly capable language model locally on your phone. *arXiv preprint arXiv:2404.14219*, 2024.

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pp. 7319–7328. Association for Computational Linguistics, Aug 2021. URL `https://aclanthology.org/2021.acl-long.568`.

Klaudia Bałazy, Mohammadreza Banaei, Karl Aberer, and Jacek Tabor. Lora-xs: Low-rank adaptation with extremely small number of parameters. *arXiv preprint arXiv:2405.17604*, 2024.

Ella Bingham and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. In *International Conference on Knowledge Discovery and Data mining (ACM SIGKDD)*, 2001.

Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on Artificial Intelligence (AAAI)*, 2020.

Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. *arXiv preprint arXiv:1905.10044*, 2019.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Zeyu Han, Chao Gao, Jinyang Liu, Jeff Zhang, and Sai Qian Zhang. Parameter-efficient fine-tuning for large models: A comprehensive survey. *arXiv preprint arXiv:2403.14608*, 2024.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-Rank Adaptation of Large Language Models. In *International Conference on Learning Representations (ICLR)*, 2022.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Ka-Wei Lee. Llm-adapters: An adapter family for parameter-efficient fine-tuning of large language models. *arXiv preprint arXiv:2304.01933*, 2023.

Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023a.

Muhammad Uzair Khattak, Syed Talal Wasim, Muzammal Naseer, Salman Khan, Ming-Hsuan Yang, and Fahad Shahbaz Khan. Self-regulating prompts: Foundational model adaptation without forgetting. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023b.

Soroush Abbasi Koohpayegani, KL Navaneet, Parsa Nooralinejad, Soheil Kolouri, and Hamed Pirsiavash. NOLA: Compressing LoRA using Linear Combination of Random Basis. In *International Conference on Learning Representations (ICLR)*, 2024.

Dawid Jan Kopiczko, Tijmen Blankevoort, and Yuki Markus Asano. Vera: Vector-based random matrix adaptation. In *International Conference on Learning Representations (ICLR)*, 2024.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning (ICML)*, 2019.

A. Krizhevsky, I. Sutskever, and G. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems (NeurIPS)*, 2012.

Chunyuan Li, Heerad Farkhoor, Rosanne Liu, and Jason Yosinski. Measuring the instrinsic dimension of objective landscapes. In *ICLR*, Vancouver, Canada, 30 Apr–3 May 2018. URL `https://openreview.net/pdf?id=ryup8-WCW`.

Ping Li, Trevor J Hastie, and Kenneth W Church. Very sparse random projections. In *ACM SIGKDD international conference on Knowledge discovery and data mining*, 2006.

W Johnson J Lindenstrauss and J Johnson. Extensions of lipschitz maps into a hilbert space. *Contemp. Math*, 1984.

Vijay Lingam, Atula Tejaswi, Aditya Vavre, Aneesh Shetty, Gautham Krishna Gudur, Joydeep Ghosh, Alex Dimakis, Eunsol Choi, Aleksandar Bojchevski, and Sujay Sanghavi. SVFT: Parameter-Efficient Fine-Tuning with Singular Vectors. In *International Conference on Machine Learning Workshops (ICMLW)*, 2024.

Shih-Yang Liu, Chien-Yi Wang, Hongxu Yin, Pavlo Molchanov, Yu-Chiang Frank Wang, Kwang-Ting Cheng, and Min-Hung Chen. Dora: Weight-decomposed low-rank adaptation. In *International Conference on Machine Learning (ICML)*, 2024.

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. *arXiv:1907.11692*, 2019.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. Pissa: Principal singular values and singular vectors adaptation of large language models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.

Parsa Nooralinejad, Ali Abbasi, Soroush Abbasi Koohpayegani, Kossar Pourahmadi Meibodi, Rana Muhammad Shahroz Khan, Soheil Kolouri, and Hamed Pirsiavash. Pranc: Pseudo random networks for compacting deep models. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, 2017.

Maxime Oquab, Timothée Darcet, Theo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, Russell Howes, Po-Yao Huang, Hu Xu, Vasu Sharma, Shang-Wen Li, Wojciech Galuba, Mike Rabbat, Mido Assran, Nicolas Ballas, Gabriel Synnaeve, Ishan Misra, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning Robust Visual Features without Supervision, 2023.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 2019.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, 2021.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 2021.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan LeBras, and Yejin Choi. Socialiqa: Commonsense reasoning about social interactions. *arXiv preprint arXiv:1904.09728*, 2019.

Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Chengzhong Xu. HydraLoRA: An Asymmetric LoRA Architecture for Efficient Fine-Tuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations (ICLR)*, 2019.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, et al. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*, 2024.

Maxime Zanella and Ismail Ben Ayed. Low-Rank Few-Shot Adaptation of Vision-Language Models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? *arXiv preprint arXiv:1905.07830*, 2019.

Frederic Z Zhang, Paul Albert, Cristian Rodriguez-Opazo, Anton van den Hengel, and Ehsan Abbasnejad. Knowledge Composition using Task Vectors with Learned Anisotropic Scaling. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024a.

Ji Zhang, Shihan Wu, Lianli Gao, Heng Tao Shen, and Jingkuan Song. Dept: Decoupled prompt tuning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024b.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Nikos Karampatziakis, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations (ICLR)*, 2023.

Ruiyi Zhang, Rushi Qiang, Sai Ashish Somayajula, and Pengtao Xie. AutoLoRA: Automatically Tuning Matrix Ranks in Low-Rank Adaptation Based on Meta Learning. *arXiv preprint arXiv:2403.09113*, 2024c.

Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Rui-Jie Zhu, Yu Zhang, Ethan Sifferman, Tyler Sheaves, Yiqiao Wang, Dustin Richmond, Peng Zhou, and Jason K Eshraghian. Scalable MatMul-free Language Modeling. *arXiv preprint arXiv:2406.02528*, 2024.