One Spin at a Time: Sequential Subspace Rotations for Parameter-Efficient Fine-Tuning

Anonymous authors
Paper under double-blind review

Abstract

In this work, we introduce SOARA (Subspace Orthogonal Adaptation via Rotational Alignment), a novel family of parameter-efficient fine-tuning (PEFT) algorithms that navigate the manifold of pretrained weights through geometric alignment. While existing SVD-based methods prioritize magnitude-driven adaptation, for example by modifying singular values (SALT, SVFT) or reparameterizing basis vectors via additive low-rank updates (PiSSA), they often overlook the intrinsic rotational symmetries of the feature space.

In contrast, SOARA treats the pretrained principal singular vectors as a foundational, fixed basis and adapts the model by learning lightweight rotational transformations within these subspaces. This preserves the representational integrity of the pretrained features while allowing for precise alignment with downstream task geometries. We provide two distinct paths for enforcing orthogonal parametrization of rotational matrices: (1) parametrization via regularization, and (2) parameterizations via sequential Givens rotations or butterfly decompositions. By operating purely through rotations, SOARA avoids the representational collapse and shift in feature distribution frequently observed in additive or purely scale-based adaptation schemes.

Our empirical results on large-scale architectures, including ViT-B16 and DeBERTa-v3-base, demonstrate that SOARA matches or exceeds the performance of state-of-the-art PEFT methods across vision and language benchmarks. SOARA achieves these gains with a minimal parameter footprint, suggesting that rotational alignment provides a superior inductive bias for preserving and repurposing the complex feature manifolds of foundation models.

1 Introduction

The remarkable success of large-scale foundation models is predicated on their ability to capture complex, high-dimensional data manifolds within their weight distributions. However, as these models scale toward trillions of parameters, the computational and memory costs of full-parameter fine-tuning have become prohibitively high. This has catalyzed the development of Parameter-Efficient Fine-Tuning (PEFT), which aims to adapt pretrained backbones by optimizing only a minimal fraction of their parameters. Established paradigms such as Adapter modules (Houlsby et al., 2019), Prompt tuning (Lester et al., 2021), and Low-Rank Adaptation (LoRA) (Hu et al., 2022) have demonstrated that significant task-specific gains can be achieved through additive low-rank updates.

An emerging line of research has recently anchored PEFT within the spectral geometry of pretrained weights. PiSSA (Meng, 2024), for instance, leverages the Singular Value Decomposition (SVD) ($W_0 = USV^\top$) to isolate and fine-tune the principal singular directions, empirically validating that downstream adaptation is often dominated by shifts in the model's top-tier singular subspace. In the "ultra-low parameter" regime, where the budget per layer is restricted to a few multiples of the model dimension, methods like SVFT (Lingam, 2024) have argued for allocating the parameter budget primarily to a narrow band around the diagonal of the singular-value matrix S. While these methods are theoretically grounded in maximizing the 'effective rank' of the adapter, they rely on an important, though unproven, assumption that increasing spectral density (rank) is the most efficient proxy for downstream performance.

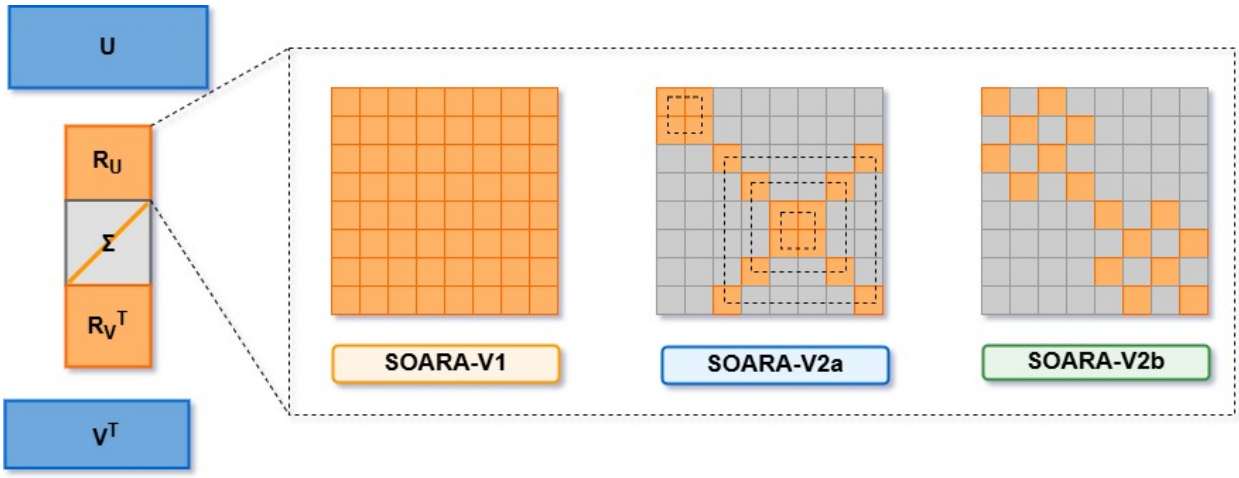


Figure 1: An overview of SOARA. The original weights W are decomposed into U, σ, and V. The three matrices on the right show the pattern of trainable (orange) and zero (grey) elements in three different variants of rotational layers. The optional residual matrix (in case of rank-splitting) has not been shown here. In SOARA-V2a, each of the four dotted squares represents an elementary Givens rotation.

We challenge this assumption by proposing a shift from magnitude-driven scaling to geometric alignment. We argue that in restricted parameter regimes, it is more effective to utilize the remaining budget for adapting the singular directions themselves (i.e., U and V) via lightweight, symmetry-preserving rotations, rather than simply widening the diagonal band of singular values.

1.1 Contributions

We introduce SOARA (Subspace Orthogonal Adaptation via Rotational Alignment) as illustrated in Fig. 1, a family of SVD-based PEFT algorithms designed to navigate the representational manifold through directional alignment. Given a weight matrix $W^* = USV^\top$, SOARA reparameterizes the adaptation in the principal subspace as:

$$W' = UR_USR_V^\top V^\top \tag{1}$$

where R_U and R_V are learnable, lightweight rotation matrices. We move beyond the "rank-maximization" focus of existing spectral methods, showing that rotational alignment provides a superior inductive bias for transfer learning by preserving the intrinsic structure of the pretrained features. We propose two distinct parameterizations for R_U, R_V: SOARA-V1, utilizing dense rotations with orthogonality regularization, and SOARA-V2, employing exactly-orthogonal structured rotations via sequential Givens layers or Butterfly factorizations (Liu et al., 2024b). SOARA-V2 is enabled by a distinct modeling formulation: we design the structured layers so that their infinitesimal generators are Frobenius-orthogonal (Theorem 3), which makes phased coordinate-descent optimization well-posed and enables a merge-and-discard training schedule that never materializes a dense $r \times r$ rotation during training (unlike coupled Givens-based orthogonal fine-tuning schemes such as qGOFT (Ma et al., 2024)). Empirically, on CIFAR-100, SOARA-V2b achieves Top-1 accuracy of 92.32% with a ViT-B/16 (Dosovitskiy et al., 2021) using only 0.129M parameters, significantly outperforming the SVFT-Banded baseline (84.65–84.85% with 0.165M parameters). On the GLUE benchmark (DeBERTa-v3-base (He et al., 2021)), SOARA matches state-of-the-art results while consuming approximately 50% fewer parameters than competitive SVD-based baselines like SVFT.

2 Prior Art

The evolution of parameter-efficient fine-tuning (PEFT) reflects a shift from structural modifications to high-dimensional geometric alignment. We categorize these methods by their fundamental impact on the model's representation manifold.

Architectural Bottlenecks and Prompting: Early PEFT strategies focused on modifying the model's computation graph. Adapter tuning (Houlsby et al., 2019) inserts lightweight bottleneck modules within frozen layers, learning task-specific residuals while maintaining the backbone's integrity. Prompt tuning (Lester et al., 2021) operates in the input space by prepending learnable 'soft prompts' to input embeddings, essentially steering the frozen model via prefix-based context. While these methods achieve significant parameter savings, they often introduce inference latency or struggle with the 'optimization instability' typical of non-linear bottleneck layers.

Additive Low-Rank Perturbations: The LoRA family (Hu et al., 2022) introduced the paradigm of additive weight updates, $W = W^* + AB$, where A and B are low-rank matrices. PiSSA (Meng, 2024) optimizes this by initializing these factors from the principal SVD components of W^*. DoRA (Liu et al., 2024a) further refines this by decoupling magnitude and directional updates to mimic the learning dynamics of full-parameter fine-tuning. Despite their popularity, additive updates can inadvertently "smear" the pretrained feature distribution by shifting the basis vectors away from their original manifold.

Spectral Scaling and Value-Shift Methods: Rather than adding new content, spectral methods modify the importance of existing features. SALT (Elsayed et al., 2025) and SAM-PARSER (Peng et al., 2022) update only the diagonal singular values of the SVD, effectively re-scaling the contribution of pretrained basis vectors. Similarly, SVDiff (Han, 2023) and SVF (Sun et al., 2024) apply shifts to singular values to regularize the adaptation process. These methods are remarkably efficient but are fundamentally constrained: they can emphasize or suppress existing features, but they cannot "rotate" the feature space to align with a novel task geometry. Spectral Adapter (Zhang & Pilanci, 2024) further explores fine-tuning directly in spectral coordinates, combining spectral updates with orthogonal transformations.

Orthogonal and Subspace-Based Alignment: Recognizing the importance of preserving pairwise neuron relationships, Orthogonal Fine-Tuning (OFT) (Qiu et al., 2023) and its butterfly-factorized variant BOFT (Liu et al., 2024b) apply a global orthogonal transform R to the weights, ensuring an isometric adaptation. OFTv2 (Qiu et al., 2025) improves the efficiency of this approach using Cayley-Neumann approximations. More recently, PSOFT (Wu et al., 2026) studies orthogonal adaptation restricted to principal subspaces, and qGOFT (Ma et al., 2024) improves the parameter-efficiency of (quasi-) orthogonal Givens-based updates.

SOARA advances this trajectory by proposing a subspace-aware rotational alignment. Unlike OFT, which rotates the entire weight matrix, SOARA identifies the principal subspaces through SVD and learns rotations specifically within these high-variance directions. This combines the geometric rigor of orthogonality with the targeted efficiency of spectral methods, providing a superior inductive bias for adapting models without representational collapse. Relative to principal-subspace orthogonal fine-tuning (PSOFT) (Wu et al., 2026) and spectral-space adapters (Zhang & Pilanci, 2024), SOARA's key novelty is not only restricting the adaptation to a principal subspace, but a generator-orthogonal factorization that supports optimal phased optimization and merge-and-discard training, yielding strong memory efficiency without sacrificing the expressiveness of exact orthogonality.

3 Problem Setup and Geometric Motivation

Let $W^* \in \mathbb{R}^{m\times n}$ represent a pretrained weight matrix with a singular value decomposition (SVD) given by $W^* = U_1 S_1 V_1^\top$, where $U_1 \in \mathbb{R}^{m\times m}$ and $V_1 \in \mathbb{R}^{n\times n}$ define the principal singular subspaces. During a full fine-tuning regime, the weights evolve toward a target state $W = U_2 S_2 V_2^\top$, where both the singular values (spectral magnitudes) and the singular vectors (representational basis) are subject to optimization.

3.1 Why Rotations on SVD Subspaces?

The core challenge of parameter-efficient fine-tuning (PEFT) is to approximate the transformation from (U_1, S_1, V_1) to (U_2, S_2, V_2) using a minimal parameter budget. We posit that for the model to retain its representational integrity, the transformation of the basis must be symmetry-preserving.

3.1.1 Full-Rank Preservation

When the adapted weight W retains the fundamental rank properties of W^*, the relationship between the pretrained and fine-tuned singular vectors is essentially a change-of-basis. Specifically, there exist orthogonal matrices $R_u \in O(m)$ and $R_v \in O(n)$ such that:

$$U_2 = U_1 R_u, \qquad V_2 = V_1 R_v \tag{2}$$

This establishes that the evolution of the model during downstream adaptation can be modeled as a rotational alignment of its internal feature manifolds. Rather than learning entirely new basis matrices, a process that is parameter-intensive and risks catastrophic forgetting of the pretrained distribution, we can simply optimize the lightweight rotations that align the existing principal subspaces with the target task geometry.

By projecting these transformations onto the principal r-dim subspaces, this parameterization allows us to compactly encode the change-of-basis using only $O(r^2)$ (or even $O(r)$ with structured factorizations) trainable parameters. We formally prove that SOARA provides an 'Isometry Protection' (Theorem 5) bound that prevents spectral drift, and identify a fundamental 'Rotational Alignment Gap' (Theorem 6) in magnitude-only scaling methods.

4 SOARA-V1: Spectral Regularization of Subspace Rotations

The most direct instantiation of rotational alignment involves learning the $r \times r$ transformation matrices R_u and R_v through unconstrained optimization, while imposing a geometric prior to ensure isometry.

4.1 Orthogonality-Constrained Optimization

In SOARA-V1, we parameterize $R_u, R_v \in \mathbb{R}^{r\times r}$ as dense matrices. To encourage these operators to remain on the manifold of orthogonal transformations, we augment the primary task loss $\mathcal{L}_{\text{task}}$ with a spectral soft-constraint:

$$\mathcal{L}_{\text{ortho}} = \lambda \left(\|R_u^\top R_u - I_r\|_F^2 + \|R_v^\top R_v - I_r\|_F^2 \right) \tag{3}$$

where I_r is the r-dimensional identity matrix and λ is a penalty coefficient (typically 10^{-4}).

4.2 Rank-Wise Splitting and Subspace Isolation

We decompose the pretrained weight matrix W^* into its principal and residual components based on its singular value energy:

$$W_{\text{pri}} = U_1[:, :r]\, S_1[:r, :r]\, V_1[:, :r]^\top, \tag{4}$$

$$W_{\text{res}} = U_1[:, r:]\, S_1[r:, r:]\, V_1[:, r:]^\top \tag{5}$$

By freezing the residual W_{res}, the adapted weight W' is reconstructed as:

$$W' = U_1[:, :r]\, (R_u S_{\text{train}} R_v^\top)\, V_1[:, :r]^\top + W_{\text{res}} \tag{6}$$

where S_{train} represents a learnable diagonal matrix initialized from $S_1[:r, :r]$. For a given rank r, SOARA-V1 introduces $2r^2 + r$ trainable parameters per layer.

Remark 1 (Decoupling Rotation and Scaling): It is important to distinguish SOARA from strictly isometric fine-tuning methods like BOFT. BOFT applies a global orthogonal transform, preserving all pairwise feature angles exactly but lacking magnitude plasticity. By decoupling rotational alignment (R_u, R_v) from magnitude scaling (S_{train}), SOARA enforces structural rigidity (the basis vectors never collapse) while permitting the anisotropic scaling necessary for tasks that demand varying feature magnitudes.

5 SOARA-V2: Exact Orthogonality via Structural Decompositions

To advance beyond the soft-constraints of regularization, we propose SOARA-V2, which utilizes structured factorizations to enforce exact mathematical isometry.

5.1 SOARA-V2a: Sequential Givens Alignment

SOARA-V2a decomposes each $r \times r$ rotation matrix into a product of $(r-1)$ sparse layers of disjoint 2D Givens rotations. A Givens rotation $G_{pq}(\theta)$ in $\mathbb{R}^r$ acts on the coordinate axes (p, q) with an angle θ:

$$G_{pq}(\theta) = I_r + (\cos\theta - 1)(e_p e_p^\top + e_q e_q^\top) - \sin\theta(e_p e_q^\top - e_q e_p^\top) \tag{7}$$

We group these degrees of freedom into $(r-1)$ sparse layers $\{L_i\}_{i=1}^{r-1}$. For even r, each layer L_i contains $r/2$ disjoint Givens rotations. By construction, each layer L_i and the composed rotation are strictly orthogonal (see Theorem 2). During training phase i, we optimize the angles $\theta^{(i)}$ for the active layer L_i while the remaining layers are held fixed. Once a phase is complete, the learned rotation is merged into the frozen bases:

$$U \leftarrow UL_i, \qquad V \leftarrow L_i V \tag{8}$$

5.1.1 Rank-Wise Splitting and Complexity

We apply rank-wise splitting as detailed in Section 4.2:

$$W' = U_r R_u S_r R_v^\top V_r^\top + W_{\text{res}} \tag{9}$$

where $R_u = \prod_{\ell=1}^{r-1} L_\ell$. Only $\lfloor r/2 \rfloor$ angles are active during any single training phase.

Remark 2 (Full $O(r)$ Coverage): While Givens rotations restrict R_u and R_v to the Special Orthogonal group $SO(r)$ (determinant $+1$), SOARA retains full $O(r)$ representational capacity. By allowing negative entries in the learnable diagonal S_{train}, the transformation naturally incorporates reflections (determinant -1). This spans the complete orthogonal space without the computational overhead of Riemannian manifold optimization.

5.2 SOARA-V2b: Butterfly Factorization

SOARA-V2b leverages a butterfly factorization structure (Liu et al., 2024b). We define a butterfly factor $BF(k)$ for a given block size $k \geq 2$ as:

$$BF(k) = \begin{bmatrix} D_1 & D_2 \\ D_3 & D_4 \end{bmatrix} \in \mathbb{R}^{k \times k} \tag{10}$$

where each $D_i = \text{diag}(d_i)$ for vectors $d_i \in \mathbb{R}^{k/2}$. We construct a butterfly component $\tilde{B}(d, k)$ as a block-diagonal matrix:

$$\tilde{B}(d, k) = \text{diag}\big(BF_1(k), \ldots, BF_{d/k}(k)\big) \in \mathbb{R}^{d \times d} \tag{11}$$

For a dimension $d = 2^N$, the resulting dense rotation matrix $B(d)$ is:

$$B(d) = \tilde{B}(d, d)\, \tilde{B}(d, d/2) \cdots \tilde{B}(d, 2) \tag{12}$$

To maintain exact isometry, each primitive 2×2 block in $\tilde{B}(d, 2)$ is parameterized as a planar rotation:

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{13}$$

Crucially, the linear generators for distinct butterfly components are Frobenius-orthogonal, which prevents parameter interference during phased optimization (Theorem 4).

6 Experimental Results

This section presents the evaluation of the proposed methods and their comparison with existing methods on vision and natural language understanding & reasoning benchmarks. Tuning/fairness protocol: we carefully retrained and retuned Full Fine-Tuning (FFT) and LoRA across our benchmarks to ensure they are optimized comparably. Unless stated otherwise, vision and NLU experiments were run on an NVIDIA A100 (40GB) machine; large-model memory profiling uses an NVIDIA A100 (80GB) to allow baselines that exceed 40GB. For the vision and NLU experiments, model optimization is achieved with the AdamW optimizer, batchsize of 32 and a learning rate of 10^{-3} with a cosine annealing scheduler.

Table 1: Performance comparison on vision benchmarks. Top-1 accuracy (%) for ViT-B/16 across five diverse datasets.

Method	# Params (↓)	CIFAR-100	DTD	SUN397	FER2013	FGVCAircraft	Avg. (↑)
Full Fine-tuning	86,630,561	92.4	72.4	75.0	68.2	52.6	72.12
BitFit (Ben Zaken et al., 2022)	358,741	81.0	72.6	71.8	60.4	45.9	66.34
Adapter-tuning	1,505,654	90.6	71.0	74.2	63.4	52.4	70.32
AdapterDrop (Rücklé et al., 2021)	174,646	88.4	70.2	72.3	46.9	35.6	62.68
LoRA	219,601	90.6	70.4	73.6	62.7	54.9	70.44
Transformer-probing	3,198,999	86.9	72.0	76.7	60.7	45.5	68.36
LoRA-Fix	148,704	88.3	65.5	72.0	53.4	51.7	66.18
LayerNorm Tuning	75,413	71.7	69.0	72.0	52.7	51.0	63.28
Attention Tuning	28,405,278	85.7	69.2	73.8	55.2	51.9	67.16
LePE Tuning (Dong et al., 2022)	167,225	90.8	69.8	73.2	60.0	49.3	68.62
RPB Tuning	145,920	87.0	70.4	72.4	50.9	51.4	66.42
KAdaptation (He et al., 2023)	114,079	91.2	71.4	**75.1**	**63.8**	55.5	71.40
SOARA-V2b (BF seq)	129,024	**92.32**	**74.096**	73.652	62.901	**59.379**	**72.47**
SOARA-V1 (r=16)	38,016	90.68	72.30	73.90	64.78	52.63	70.86
SOARA-V2a (Givens-seq, r=16)	2,304	89.11	71.49	67.69	60.11	49.96	67.67

6.1 Vision Benchmark

We evaluate the efficacy of SOARA on a diverse suite of vision benchmarks using a ViT-B/16 backbone pretrained on ImageNet-21k. Our evaluation spans five core benchmarks namely, CIFAR-100 (Krizhevsky & Hinton, 2009), DTD (Cimpoi et al., 2014), SUN397 (Xiao et al., 2010), FER2013 (Goodfellow et al., 2013), and FGVCAircraft (Maji et al., 2013), alongside specialized robustness checks on Food-101 (F101) (Bossard et al., 2014) and RESISC45 (R45) (Ullah et al., 2022). We apply SOARA to the q, k, v and three dense linear projections per transformer layer, totaling 72 adapted layers.

6.1.1 Main Results

As reported in Table 1, SOARA-V2b (Butterfly sequential) emerges as the dominant performer across nearly all metrics. Remarkably, it achieves a mean accuracy of 72.47%, which not only exceeds all PEFT baselines but also surpasses full-model fine-tuning (72.12%) while using only 129k trainable parameters, a reduction of over 670×. The performance of our regularization-based variant, SOARA-V1 ($r = 16$), is equally compelling; it matches the mean accuracy of LoRA and Adapter-tuning (70.86% vs. 70.44% and 70.32%) while utilizing just 17.3% and 2.5% of their respective parameter budgets. Even in the extreme ultra-low-parameter regime, SOARA-V2a (Givens-seq) maintains significant representational capacity, delivering 67.67% average accuracy with only 2k total trainable parameters. We note that surpassing full fine-tuning is not unprecedented in PEFT. For instance, PiSSA (Meng, 2024) also exceeds FFT on GSM8K (Table 4). We attribute such behavior to the implicit regularization of (near-)isometric constraints, which bounds spectral drift from pretrained weights (Theorem 5) and can mitigate overfitting while preserving pretrained structure.

6.1.2 Rotational Alignment vs. Spectral Scaling

The results in Table 2 provide a definitive comparison against existing SVD-based paradigms like SVFT and VeRA. On specialized datasets like F101 and R45, both SVFT and VeRA exhibit substantial performance degradation, likely due to representational drift. In contrast, SOARA remains highly robust, maintaining a 15–17% absolute lead over SVFT on the R45 benchmark. This performance gap confirms our central hypothesis, i.e., rather than merely scaling the magnitude of existing features, adapting the singular directions via structured rotations provides a more expressive and regularized inductive bias.

6.2 Natural Language Understanding

We evaluate SOARA on the GLUE benchmark (Wang et al., 2018) with DeBERTa-v3-base (184M). The results are reported in Table 3. SOARA demonstrates a superior trade-off between parameter budget and

Table 2: Comparative analysis [Top-1 accuracy (%)] of SVD-based PEFT & robustness on additional datasets.

Method	# Params	C100	F101	R45
Classification Head	–	78.58	75.14	64.42
VeRA (r=256)	24.6K	84.00	74.02	71.86
SVFT^r (d=4)	165.4K	84.85	76.45	74.53
SVFT^b (d=4)	165.4K	84.65	76.51	75.12
SVFT^p	18.5K	83.78	74.43	70.55
SOARA-V2b (BF seq)	129K	**92.32**	85.16	**92.032**
SOARA-V1 (r=16)	38K	90.68	**86.657**	91.714
SOARA-V2a (r=16)	2.3K	89.11	85.806	90.667

Table 3: GLUE development-set results (%) with adapted DeBERTa-v3-base.

Method	Params (↓)	MNLI	SST-2	MRPC	CoLA	QNLI	QQP	RTE	STS-B	Avg. (↑)
Full Fine-tuning	184M	89.90	95.63	89.46	69.19	94.03	**92.40**	83.75	91.60	88.25
HAdapter	1.22M	90.13	95.53	89.95	68.64	94.11	91.91	84.48	91.48	88.28
PAdapter	1.18M	90.33	95.61	89.46	68.77	94.29	92.04	85.20	91.54	88.41
LoRA$_G$ (Hu et al., 2022)	1.33M	90.65	94.95	89.95	69.82	93.87	91.99	85.20	91.60	88.50
LoRA$_K$	1.33M	89.96	95.64	90.28	70.69	93.84	92.03	84.84	91.68	88.62
DoRA (Liu et al., 2024a)	1.27M	90.29	95.79	90.93	70.85	94.10	92.07	86.04	91.79	88.98
AdaLoRA (Zhang, 2023)	1.27M	**90.76**	96.10	90.69	71.45	**94.55**	92.23	88.09	91.84	89.46
PiSSA (Meng, 2024) (r=8)	1.33M	90.37	96.22	91.50	**73.12**	94.43	92.33	88.69	92.00	89.83
LoRA* (r=8)	1.33M	90.65	94.95	89.95	69.82	93.87	91.99	85.20	91.60	88.50
DoRA (r=4)	0.75M	89.92	95.41	89.10	69.37	94.14	91.53	87.00	91.80	88.53
BOFT* (b=8, m=2)	0.75M	90.25	**96.44**	92.40	72.95	94.23	92.10	88.81	91.92	**89.89**
LoRA (r=1)	0.17M	90.12	95.64	86.43	69.13	94.18	91.43	87.36	91.52	88.23
BitFit	0.10M	89.37	94.84	87.75	66.96	92.24	88.41	78.70	91.35	86.20
VeRA (Kopiczko et al., 2024) (r=1024)	0.09M	89.93	95.53	87.94	69.06	93.24	90.40	87.00	88.71	87.73
SVFT^p	0.06M	89.69	95.41	88.77	70.95	94.27	90.16	87.24	91.80	88.54
SVFT (d=2)	0.28M	89.97	95.99	88.99	72.61	93.90	91.50	88.09	91.73	89.10
SOARA-V2a (r=85)	0.012M	86.23	93.92	86.78	60.40	90.24	87.27	77.26	89.07	83.90
SOARA-V2b (r=768)	0.129M	89.31	95.41	92.85	73.10	93.63	89.39	86.28	**92.02**	89.00
SOARA-V1 (r=64)	0.59M	89.85	95.76	**93.60**	73.11	94.03	91.11	**88.81**	91.30	89.70

task performance. SOARA-V2b matches the performance of SVFT ($d = 2$) while utilizing less than half the trainable parameters (0.129M vs. 0.28M). SOARA-V1 ($r = 64$) achieves an average accuracy of 89.70%, approaching the performance of PiSSA (89.83%) and AdaLoRA (89.46%) despite utilizing only a fraction of the parameters. In the ultra-low regime, SOARA-V2a recovers over 95.1% of the full fine-tuning performance while using a mere 0.0065 of the total model parameters.

6.3 Mathematical Reasoning

To test the scalability and reasoning capacity of SOARA, we evaluate it on the GSM8K benchmark (Cobbe et al., 2021) using Gemma-7B (Team, 2024). The results are reported in Table 4. SOARA-V1 ($r = 128$) achieves 76.50% accuracy, notably exceeding Full-FT (74.67%), reinforcing the hypothesis that rotational constraints act as a powerful inductive bias. SOARA-V1 reaches 98.3% of the performance of PiSSA while requiring only 25.8% of its parameter budget (6.45M vs. 25M). SOARA-V2b attains 76.09% accuracy, effectively matching SVFT-R (76.81%) while consuming only 7.1% of its parameter footprint.

6.3.1 Runtime and Memory Overhead

We report peak GPU memory (GB) for GSM8K fine-tuning runs (MetaMathQA-full, 3 epochs). These measurements highlight that SOARA's sequential merge-and-discard strategy can substantially reduce peak VRAM compared to coupled baselines that require materializing larger orthogonal transforms.

Table 4: Zero-shot accuracy (%) on GSM8K with Gemma-7B.

Method	# Params	GSM8K
Full-FT	8.5B	74.67
LoRA (r=32)	68.8M	76.57
DoRA (r=16)	35.5M	74.52
BOFT (b=8, m=2)	2.90M	71.79
DoRA (r=1)	3.26M	74.37
LoRA (r=1)	0.82M	72.40
VeRA (r=1024)	0.43M	71.11
SVFT-P	0.43M	73.50
SVFT-R^d	19.8M	76.81
PiSSA (r=8)	25.00M	**77.78**
SOARA-V1 (r=128)	6.45M	76.50
SOARA-V2b (BF seq)	1.40M	76.09

Table 5: Memory overhead on GSM8K (Gemma-7B).

Method	# Params	Peak GPU (GB)	GSM8K (%)
SVFT-P	0.43M	76.26	73.50
SVFT-R^d	19.8M	77.04	76.81
SOARA-V2b (BF seq)	1.40M	38.47	76.09

On Llama-3-8B (MetaMathQA-full, 3 epochs), SOARA-V2a and SOARA-V2b finish in ∼114 and ∼130 hours, respectively. Since many orthogonal PEFT baselines do not report wall-clock times for LLMs, direct runtime comparisons are limited, but the peak memory reduction is practically significant.

6.3.2 Commonsense Reasoning on Llama-3-8B

We further evaluate SOARA-V2b on Llama-3-8B across commonsense reasoning benchmarks to test generalization on modern large-scale architectures.

6.4 Ablation Studies

All reported results are mean accuracy over 3 seeds per configuration.

6.4.1 Stability of Sequential Training: Cycle Invariance

We evaluate SOARA-V2a on FGVC Aircraft with a fixed budget of 20 epochs and rank $r = 16$. As illustrated in Fig. 2, the convergence curves for total cycles $\in \{1, 3, 5, 7\}$ are virtually indistinguishable across training and validation loss, suggesting that SOARA is remarkably robust to the cycle hyperparameter.

6.4.2 The Power of Ultra-Low Rank Alignment

We evaluate SOARA-V1 across five vision datasets with $r \in \{2, 4, 8, 16\}$. The convergence plots in Fig. 3 reveal that increasing the rank from $r = 2$ to $r = 16$ significantly accelerates convergence and improves final accuracy, with diminishing returns beyond $r = 8$ for several benchmarks.

6.4.3 Manifold Visualization: t-SNE Cluster Analysis

We perform a t-SNE visualization of the CLS-token representations from the intermediate layer index 6 of the ViT backbone on the FGVC Aircraft test split. The visualization in Fig. 4 reveals a clear correlation

Table 6: Memory overhead on GSM8K (Llama-3-8B).

Method	# Params	Peak GPU (GB)	GSM8K (%)
LoRA (r=32)	56.6M	58.41	75.89
SVFT-R	15.98M	71.52	75.32
SVFT-P	0.483M	73.95	69.22
SOARA-V2b (BF seq)	1.44M	38.60	77.20

Table 7: Commonsense reasoning accuracy (%) on Llama-3-8B.

Method	ARC-c	ARC-e	BoolQ	HellaSwag	OBQA	PIQA	SIQA	WinoGrande	Avg
SOARA-V2b (%)	79.01	92.89	85.42	94.99	85.80	88.52	81.99	85.71	**86.79**

between rank and cluster purity. At $r = 2$ and $r = 4$, class clusters exhibit significant overlap, while at $r = 16$ and in SOARA-V2b (Butterfly), we observe significantly tighter within-class variance and cleaner inter-class separation.

7 Conclusion

In this work, we introduced SOARA, a principled framework for PEFT grounded in the geometric properties of the weight manifold. By shifting the focus from additive magnitude-driven updates to the rotational alignment of principal SVD subspaces, we provide a mechanism that adapts foundation models while preserving their representational integrity. Our dual-path approach—SOARA-V1 via spectral regularization & SOARA-V2 via exact Givens and Butterfly factorizations—demonstrates that enforcing mathematical isometry is a powerful inductive bias for transfer learning. Our empirical evaluations across vision and language modalities confirm that SOARA matches or exceeds the performance of established baselines like LoRA and SVFT, often with 10–100× fewer parameters.

7.1 Limitations and Future Directions

While SOARA establishes a robust geometric foundation for PEFT, several avenues for future research remain: (1) *Architectural Generalization*: the applicability of rotational alignment to convolutional layers warrants investigation. (2) *Quantization and Compression*: Integrating SOARA with deployment-centric techniques such as QLoRA (Dettmers et al., 2023) or NOLA (Koohpayegani et al., 2024) presents a promising direction for extreme edge-device optimization. (3) *Distributed and Federated Learning*: The communication-efficient nature of SOARA-V2 suggests significant potential for federated learning compared to federated variants like SLoRA (Babakniya et al., 2023).

Broader Impact Statement

This paper presents work whose goal is to advance the field of machine learning by reducing the computational cost of fine-tuning large models. This has potential positive societal consequences by democratizing access to model adaptation.

Conflict of Interest

The authors declare that they have no conflict of interest to disclose.

Table 8: Ablation 2.1: rotating U only, V only, or both (ViT-B/16).

SOARA-V1 variant	CIFAR-100	FGVC Aircraft
rotate U only	89.70	49.83
rotate V only	89.94	51.19
rotate U+V	**90.68**	**52.63**

Table 9: Ablation 2.2: learnable vs. frozen S_{train} (mean accuracy gain in pp).

SOARA variant	CIFAR-100 (Δ)	FGVC Aircraft (Δ)
V1 (learnable)	+0.38	+0.63
V2b (learnable)	+1.85	+3.37

A Theoretical Analysis: The Geometry of Effective Rank

In this section, we provide a formal justification for the performance trade-offs observed in SVD-based adapters. Specifically, we analyze why SVFT-style "banded" patterns are designed to maximize the stable rank of an adapter, while highlighting the inherent geometric limitations of such sparsity constraints.

Problem Setup: Consider a generic SVFT-style adapter as a sparse linear combination of orthonormal bases. Let $\{u_i\}_{i=1}^m$ and $\{v_j\}_{j=1}^n$ be the orthonormal bases derived from the pretrained weights W^*. For a sparsity pattern $S \subseteq [m] \times [n]$ with $|S| = k$ non-zero entries, we define the sparse coefficient matrix $M \in \mathbb{R}^{m\times n}$ and its corresponding adapter perturbation P' as:

$$P' = \sum_{(i,j)\in S} \alpha_{ij}\, u_i v_j^\top \tag{14}$$

By the unitary invariance of singular values, the spectrum of the physical adapter P' is mathematically identical to that of the sparse coefficient matrix M (i.e., $\sigma_l(P') = \sigma_l(M)$).

Observation 1 (The SVFT Expressivity Dilemma): If M were a fully dense matrix, the adapter $UMV^\top$ would possess maximum representational power, theoretically capable of encompassing arbitrary rotations, reflections, and scalings. However, to maintain strict parameter efficiency, methods like SVFT constrain M to a sparse, banded structure. While this banded structure mathematically preserves the stable rank of M (as we prove below), it fundamentally cripples M's ability to act as an effective rotation matrix. Thus, SVFT sacrifices geometric alignment (direction) in favor of spectral density (magnitude). SOARA bypasses this dilemma by explicitly parameterizing M as a composition of exact rotations and scalings ($R_u S_{train} R_v^\top$).

A.1 Maximizing Information Density via Stable Rank

To measure the "richness" of the learned transformation, we utilize the **stable rank**, defined as the ratio of the squared Frobenius norm to the squared spectral norm:

$$\text{srank}(A) = \frac{\|A\|_F^2}{\|A\|_2^2} = \frac{\sum_{\ell=1}^r \sigma_\ell^2}{\sigma_1^2} \tag{15}$$

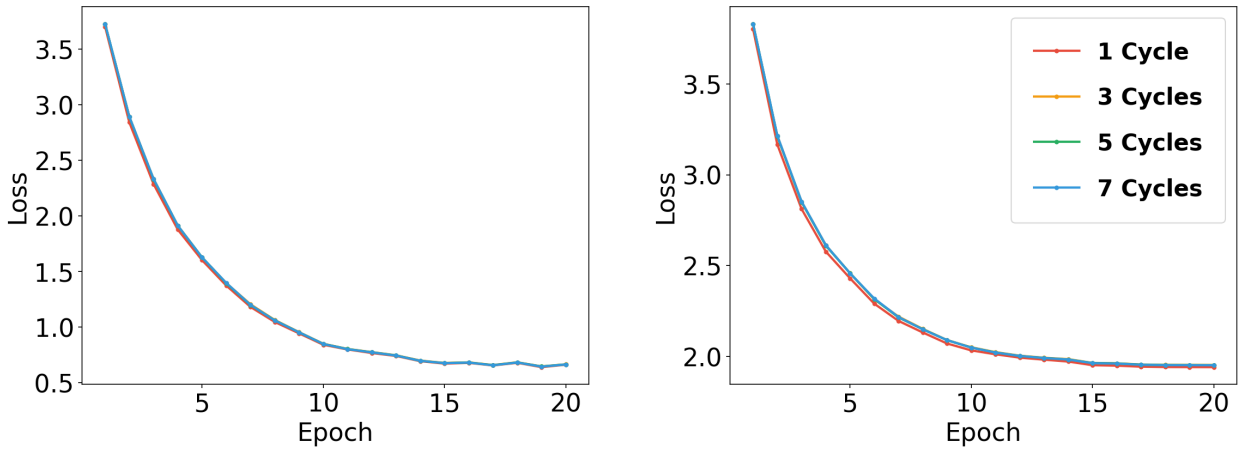
The stable rank is a robust measure of how evenly the "energy" of a matrix is distributed across its singular directions, avoiding the instability of the algebraic rank.

Lemma 1. *For a fixed parameter budget $\|M\|_F^2 = k$, the stable rank is maximized when the spectral norm $\sigma_1(M)$ is minimized, which occurs when the non-zero singular values are distributed uniformly.*

Proof. By definition, $\text{srank}(M) = k/\sigma_1^2$. To maximize this ratio for a fixed k, one must minimize the largest singular value σ_1^2. Since $\sum \sigma_\ell^2 = k$, the minimum possible value for σ_1^2 occurs when all r non-zero singular values are equal, yielding $\sigma_1^2 = k/r$ and thus $\text{srank}(M) = r$. □

Table 10: Ablation 2.3: exact orthogonality vs. soft regularization under matched per-layer parameter budget.

SOARA variant	CIFAR-100	FGVC Aircraft
V1 (soft-reg, r=16)	89.81	52.03
V2a (Givens, r=23)	90.11	50.17
V2b (Butterfly, r=64)	**90.53**	**57.79**



(a) Train loss

(b) Validation loss

Figure 2: Cycle invariance in SOARA-V2a demonstrated on FGVC Aircraft. Performance remains nearly identical for cycles 1 through 7.

A.2 Row/Column Degree Balance and Spectral Norm Bounds

A key design principle behind SVFT-style sparsity is to avoid creating "hub" rows or columns with many non-zero entries. Such hubs inflate the operator norm $\|M\|_2 = \sigma_1(M)$, which in turn drives down the stable rank. For a sparsity pattern $S \subseteq [m] \times [n]$, define the row- and column-degrees

$$r_i := |\{j : (i,j) \in S\}|, \quad c_j := |\{i : (i,j) \in S\}|. \tag{16}$$

Since $\sum_i r_i = \sum_j c_j = |S| = k$, the average degrees are k/m and k/n.

Lemma 2. *(Unavoidable maximum degree) For any pattern S with $|S| = k$, the maximum degrees are bounded below by the pigeonhole principle:*

$$\max_i r_i \geq \lceil k/m \rceil \quad \textit{and} \quad \max_j c_j \geq \lceil k/n \rceil \tag{17}$$

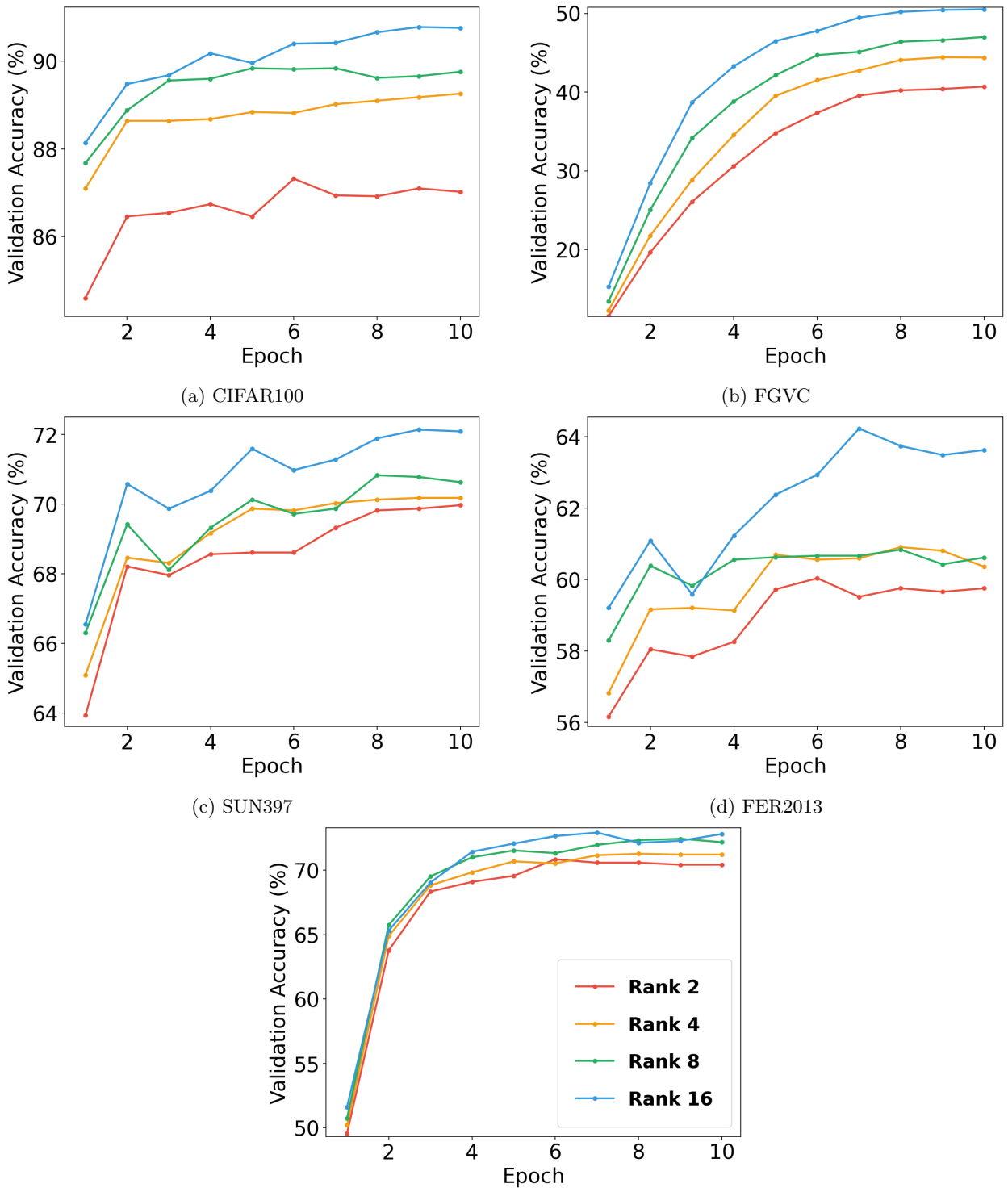
Let us denote these theoretical minimums as d_r and d_c. From random matrix theory, if the non-zero entries of M are drawn from a sub-Gaussian distribution, the spectral norm is heavily governed by these maximum degrees:

$$\sigma_1(M) \lesssim \sqrt{\max_i r_i} + \sqrt{\max_j c_j} \tag{18}$$

Proof. The maximum degrees strictly satisfy $\max_i r_i \geq \lceil k/m \rceil$ and $\max_j c_j \geq \lceil k/n \rceil$. Since the sum of all degrees equals the total non-zero entries ($\sum_{i=1}^m r_i = k$ and $\sum_{j=1}^n c_j = k$), the peak degree is mathematically bounded below by the integer ceiling of the mean distribution. □

A.3 Singular-Value Concentration Under Degree Constraints

Assume the non-zero entries of M are random with zero mean and unit variance, meaning the expected Frobenius energy is $\mathbb{E}\|M\|_F^2 = k$. Balancing the degrees strictly controls the spectral norm.



(a) CIFAR100

(b) FGVC

(c) SUN397

(d) FER2013

(e) DTD

Figure 3: Impact of subspace rank on convergence. Validation accuracy vs. epoch for $r \in \{2, 4, 8, 16\}$.

Proposition 1. *(Heuristic; sparse random-matrix intuition) Let $M \in \mathbb{R}^{m\times n}$ have support S with $|S| = k$ and i.i.d. $\mathcal{N}(0,1)$ entries on S. If $\max_i r_i \le d_r$ and $\max_j c_j \le d_c$, then with high probability:*

$$\sigma_1(M) \lesssim \sqrt{d_r} + \sqrt{d_c}, \tag{19}$$

Since the stable rank is defined as $\mathrm{srank}(M) = \|M\|_F^2/\sigma_1(M)^2$, degree imbalance (which increases σ_1) severely penalizes the stable rank. To maximize the stable rank for a given parameter budget k, we must minimize the spectral norm, which dictates that the structural degrees must be as close to the uniform lower bounds (d_r, d_c) as possible.

Proof. By standard concentration inequalities (Latała, 2005), the spectral norm of a zero-mean random matrix is asymptotically dominated by the maximum expected ℓ_2 norms of its rows and columns (Bandeira & van Handel, 2016). For $\mathcal{N}(0,1)$ entries, the expected squared ℓ_2 norm of row i and column j corresponds precisely to their respective degrees, r_i and c_j. Therefore, $\sigma_1(M) \lesssim \sqrt{\max_i r_i} + \sqrt{\max_j c_j} \le \sqrt{d_r} + \sqrt{d_c}$, demonstrating that structural degree balance tightly bounds the principal singular value. □

A.4 Why Banded Patterns Are a Natural Choice

A banded/near-diagonal pattern is a constructive way to achieve perfect degree balance while keeping the support connected.

Theorem 1. *(Degree-balanced supports promote near-uniform spectra) Among supports S of size k, patterns that attain the near-minimal maxima $\max_i r_i \approx d_r$ and $\max_j c_j \approx d_c$ minimize the spectral norm $\sigma_1(M)$. Banded patterns perfectly balance these degrees by design, thereby maximizing the stable rank for a given parameter budget.*

Proof Sketch: By Proposition 1, balancing d_r and d_c controls the spectral norm upper bound. Because the stable rank is defined as $\mathrm{srank}(M) = k/\sigma_1(M)^2$, minimizing the spectral norm analytically guarantees a higher stable rank. Banded matrices achieve the theoretical minimum for maximum row/column degrees, ensuring the adapter's energy is maximally distributed across the subspace rather than collapsed into a single dominant direction.

A.5 Conclusion of the Stable-Rank Argument

Putting the pieces together, the standard SVFT motivation can be rigorously summarized as:

1. By **Observation 1**, the adapter's spectral properties are equivalent to those of the sparse coefficient matrix M.
2. Maximizing the stable rank requires minimizing the spectral norm $\sigma_1(M)$ for a fixed parameter budget k (Lemma 1).
3. Row/column degree balance (Lemma 2) is required to prevent structural hubs that inflate $\sigma_1(M)$.
4. Banded patterns achieve optimal degree balance (Theorem 1), thereby minimizing the spectral norm and maximizing the stable rank.

Discussion: The analysis above justifies exactly why SVFT utilizes banded patterns: it is the optimal strategy to maximize stable rank. However, it highlights a fatal geometric flaw: stable rank is not spatial alignment. While a banded matrix successfully spreads energy across singular values, its rigid diagonal structure fundamentally prevents it from capturing the cross-coordinate off-diagonal elements required to perform a rigid mathematical rotation. This provides the theoretical motivation for SOARA: instead of merely maximizing the spread of spectral energy (stable rank), we must parameterize our budget to explicitly rotate the basis vectors (R_u, R_v) to align with the downstream task geometry.

B SOARA: Orthogonality of Givens Layers

Theorem 2. *Each Givens layer L_i and the composed rotation $R_u = \prod_{i=1}^{r-1} L_i$ are strictly orthogonal.*

Proof. A planar Givens rotation $G_{pq}(\theta)$ is orthogonal ($G^\top G = I$). In a single layer L_i, all rotations act on disjoint index pairs, so L_i is block-diagonal of 2×2 rotation blocks: $L_i = \mathrm{diag}(R(\theta_1), R(\theta_2), \ldots, I)$. Since each block is orthogonal, $L_i^\top L_i = I$. The product of orthogonal matrices is orthogonal: $R_u^\top R_u = \left(\prod_{i=r-1}^{1} L_i^\top\right)\left(\prod_{i=1}^{r-1} L_i\right) = I_r$. □

Theorem 3. *Let $A^{(k_1)}$ and $A^{(k_2)}$ be the infinitesimal generators of two rotation layers acting on mutually disjoint index pairs. Then $\langle A^{(k_1)}, A^{(k_2)} \rangle_F = 0$.*

Proof. If the layers act on disjoint index pairs ($\mathcal{P}_{k_1} \cap \mathcal{P}_{k_2} = \varnothing$), then every pair $\{p, q\}$ is distinct from every $\{i, j\}$, and $\mathrm{tr}(A_{pq}^\top A_{ij})$ is non-zero only when pairs coincide. Thus every term vanishes. □

B.1 Orthogonality of Butterfly Component Generators

Lemma 3. *If $k_1 \neq k_2$, then $\mathcal{P}_{k_1} \cap \mathcal{P}_{k_2} = \varnothing$.*

Proof. In a butterfly component, $\{a, b\} \in \mathcal{P}_k$ iff $b - a = k/2$. If a pair existed in both $\mathcal{P}_{k_1}$ and $\mathcal{P}_{k_2}$, then $k_1 = k_2$, a contradiction. □

Theorem 4. *The linear generators for distinct butterfly components are Frobenius-orthogonal.*

Proof. Let A_{pq} be the elementary skew-symmetric generator for pair $\{p, q\}$. The Frobenius inner product is:

$$\langle A^{(k_1)}, A^{(k_2)} \rangle_F = \sum_{\{p,q\} \in \mathcal{P}_{k_1}} \sum_{\{i,j\} \in \mathcal{P}_{k_2}} \alpha_{pq} \beta_{ij} \, \mathrm{tr}(A_{pq}^\top A_{ij}). \tag{20}$$

Since $\mathcal{P}_{k_1} \cap \mathcal{P}_{k_2} = \varnothing$, every pair $\{p, q\}$ is distinct from $\{i, j\}$. The trace $\mathrm{tr}(A_{pq}^\top A_{ij})$ is only non-zero if the pairs are identical. Thus, every term in the sum is zero, and $\langle A^{(k_1)}, A^{(k_2)} \rangle_F = 0$. □

This confirms that SOARA's structured factorizations avoid parameter interference. Because the generators are orthogonal, optimizing one layer of the butterfly or Givens sequence does not overwrite the geometric progress made by another, leading to the cycle invariance observed in our results.

Remark on Non-Linear Mapping: While the linear generators are orthogonal, the actual rotation matrices are obtained via a non-linear map, such as the Cayley transform:

$$R = (I + A)(I - A)^{-1} \tag{21}$$

or the matrix exponential. It is important to note that while these maps preserve the orthogonality of the resulting matrix R, they do not generally preserve the linear orthogonality of the generators in the matrix space. However, the Frobenius orthogonality of the generators ensures that at the start of optimization (the infinitesimal regime), the update directions for different butterfly components are perfectly decoupled.

C Generalization Gap and Spectral Stability

Theorem 5. *Let W^* be a pretrained weight matrix with principal singular values S, and let W' be its SOARA-adapted version. Then:*

$$\|W' - W^*\|_2 \leq \|S_{train} - S\|_\infty + \|S\|_\infty \cdot (\|R_u - I\|_2 + \|R_v - I\|_2) \tag{22}$$

Proof. Let $W^* = USV^\top$ and $W' = U(R_u S_{\text{train}} R_v^\top)V^\top$. Since the spectral norm is unitarily invariant (multiplying by orthogonal $U^\top, V$ preserves norms):

$$\|\Delta W\|_2 = \|R_u S_{\text{train}} R_v^\top - S\|_2$$

We decompose the error into a scaling term and a rotation term by adding/subtracting $R_u S R_v^\top$:

$$R_u S_{\text{train}} R_v^\top - S = R_u(S_{\text{train}} - S)R_v^\top + (R_u S R_v^\top - S)$$

Using the triangle inequality:

$$\|\Delta W\|_2 \leq \|R_u(S_{\text{train}} - S)R_v^\top\|_2 + \|R_u S R_v^\top - S\|_2$$

- *Scaling Term:* Since R_u, R_v are isometries, $\|R_u A R_v^\top\|_2 = \|A\|_2$. Thus, the first term simplifies to $\|S_{\text{train}} - S\|_\infty$.
- *Rotation Term:* We further decompose: $R_u S R_v^\top - S = R_u S(R_v^\top - I) + (R_u - I)S$. Applying $\|AB\| \leq \|A\|\|B\|$ and noting $\|R_u\| = 1$: $\leq \|S\|_\infty \|R_v - I\|_2 + \|R_u - I\|_2 \|S\|_\infty$.

Summing these terms completes the proof. □

Remark: Unlike additive PEFT methods where deviation is theoretically unbounded ($\Delta W = BA$), SOARA's deviation is constrained by the pretrained energy $\|S\|_\infty$. In the fine-tuning regime where rotations are small ($\|R - I\|_2 \approx \theta$), the deviation scales linearly with the angular task alignment, providing a Lipschitz-like stability guarantee with constant $L \propto \sigma_{\max}(W^*)$.

Theorem 6. *Let $W_{opt} = U_2 S_2 V_2^\top$ be the optimal weight matrix for a downstream task. If the task requires a rotation of the singular basis (i.e., the principal subspace of W_{opt} is misaligned with the pretrained basis U_1, V_1 by principal angles θ_i), then any magnitude-only scaling method (e.g., SVFT) suffers an irreducible approximation error bounded from below by:*

$$\min_{\hat{S}} \|W_{opt} - U_1 \hat{S} V_1^\top\|_F^2 \geq \sum_{i=1}^{r} \sigma_{2,i}^2 \sin^2(\theta_i) \tag{23}$$

where $\sigma_{2,i}$ are the task-optimal singular values.

Proof. We seek to minimize the Frobenius norm of the residual $E = W_{\text{opt}} - U_1 \hat{S} V_1^\top$ with respect to the diagonal matrix $\hat{S}$.

- *Decomposition:* Expanding the norm squared:

$$J(\hat{S}) = \|U_2 S_2 V_2^\top\|_F^2 + \|U_1 \hat{S} V_1^\top\|_F^2 - 2\,\text{tr}\left((U_1 \hat{S} V_1^\top)^\top (U_2 S_2 V_2^\top)\right)$$

  Since U, V are orthonormal, $\|U_1 \hat{S} V_1^\top\|_F^2 = \|\hat{S}\|_F^2 = \sum_i \hat{s}_i^2$. The cross-term simplifies using the trace property:

$$\text{tr}(V_1 \hat{S} U_1^\top U_2 S_2 V_2^\top) = \text{tr}(\hat{S} \cdot \underbrace{U_1^\top U_2 S_2 V_2^\top V_1}_{M})$$

  The objective is minimized when $\hat{s}_i$ matches the diagonal elements of M:

$$\hat{s}_i^{\text{opt}} = M_{ii} = (u_{1,i}^\top u_{2,i})\sigma_{2,i}(v_{2,i}^\top v_{1,i})$$

- *Geometric Simplification:* Without loss of generality, assume the misalignment is primarily in the left subspace (U) and the right subspace is aligned ($V_1 \approx V_2$). Then $v_{2,i}^\top v_{1,i} \approx 1$, and $u_{1,i}^\top u_{2,i} = \cos\theta_i$. Thus, $\hat{s}_i^{\text{opt}} = \sigma_{2,i} \cos\theta_i$.

- *Irreducible Error Calculation:* Substituting $\hat{s}_i^{\text{opt}}$ back:

$$\text{Error} = \sum_i \sigma_{2,i}^2 + \sum_i (\sigma_{2,i} \cos\theta_i)^2 - 2\sum_i (\sigma_{2,i} \cos\theta_i)^2$$

$$= \sum_i \sigma_{2,i}^2 - \sum_i \sigma_{2,i}^2 \cos^2\theta_i = \sum_i \sigma_{2,i}^2 \sin^2\theta_i$$

This error is non-zero whenever $\theta_i \neq 0$, proving that magnitude scaling cannot recover the optimal weights if the subspace has rotated.

□

This identifies a fundamental *Alignment Gap*. If a task requires the model to re-orient its internal logic (e.g., shifting from general language to mathematical reasoning), magnitude scaling can never close the gap, no matter how much you increase the rank.

D Detailed Parameter Calculation

For a transformer with L layers and k target projections per layer:

- **SOARA-V1**: $L \times k \times (2r^2 + r)$
- **SOARA-V2a** (Givens; per phase): $L \times k \times (2 \times \lfloor r/2 \rfloor + r)$
- **SOARA-V2b** (Butterfly; per phase): let $d = 2^{\lceil \log_2 r \rceil}$, then $L \times k \times (d + r)$

E Givens Rotation Construction

A Givens rotation $G_{pq}(\theta)$ is an $r \times r$ identity matrix with four modified entries:

$$G_{pq}(\theta)_{pp} = G_{pq}(\theta)_{qq} = \cos\theta,$$
$$G_{pq}(\theta)_{pq} = -\sin\theta, \quad G_{pq}(\theta)_{qp} = \sin\theta.$$

For SOARA-V2a, we generate $(r-1)$ layers of disjoint pairings.

Example ($r = 5$). When r is odd, we use the same round-robin procedure with the odd-r handling in the algorithm below. One valid pairing schedule is:

- Layer 0: pairs $(0, 1), (3, 4)$
- Layer 1: pairs $(1, 4), (2, 3)$
- Layer 2: pairs $(0, 4), (1, 2)$
- Layer 3: pairs $(0, 3), (4, 2)$
- Layer 4: pairs $(0, 2), (3, 1)$

In each layer, the listed pairs are disjoint, and across layers every unordered pair of coordinates appears exactly once.

E.1 Round-Robin Pairing Algorithm

Closed-form description (even r). Let v_0 be a fixed node (we take $v_0 = 0$), and let the remaining nodes be arranged in a list $(v_1, \ldots, v_{r-1})$. At layer $i \in \{1, \ldots, r-1\}$, define

$$P_i \;\coloneqq\; \{(v_0, v_1)\} \;\cup\; \{(v_j, v_{r-j})\}_{j=2}^{r/2}, \tag{24}$$

then rotate the list by one step,

$$(v_1, \ldots, v_{r-1}) \leftarrow (v_{r-1}, v_1, \ldots, v_{r-2}). \tag{25}$$

By construction, each P_i is a matching (disjoint pairs), and across layers every unordered pair of coordinates appears exactly once.

Illustration ($r = 6$). For instance, $P_1 = \{(0,1), (2,3), (4,5)\}$ induces

$$L_1 = G_{01}(\theta_{01}) \cdot G_{23}(\theta_{23}) \cdot G_{45}(\theta_{45}), \tag{26}$$

Algorithm 1 Round-robin disjoint pairings for Givens layers

```
Input: Dimension r, number of layers n_layers
nodes <- [0,1,...,r-1]
if r is odd:
  append dummy node -1 to nodes
fixed <- nodes[0]
rot <- nodes[1:]
all_pairings <- []
for i = 0 to min(n_layers, |nodes|-1)-1:
  pairs <- []
  if rot[0] != -1:
    add (fixed, rot[0]) to pairs
  for j = 1 to floor(|rot|/2):
    if rot[j] != -1 and rot[-j] != -1:
      add (rot[j], rot[-j]) to pairs
  append pairs to all_pairings
  rot <- [rot[-1]] + rot[:-1]
return all_pairings
```

F Explicit Butterfly Factors for $b = 1$

A single $BF(8)$ is an 8×8 matrix of four independent rotations on pairs $(0,4)$, $(1,5)$, $(2,6)$, $(3,7)$:

$$BF(8; \theta_1, \theta_2, \theta_3, \theta_4) = \begin{bmatrix} c_1 & 0 & 0 & 0 & -s_1 & 0 & 0 & 0 \\ 0 & c_2 & 0 & 0 & 0 & -s_2 & 0 & 0 \\ 0 & 0 & c_3 & 0 & 0 & 0 & -s_3 & 0 \\ 0 & 0 & 0 & c_4 & 0 & 0 & 0 & -s_4 \\ s_1 & 0 & 0 & 0 & c_1 & 0 & 0 & 0 \\ 0 & s_2 & 0 & 0 & 0 & c_2 & 0 & 0 \\ 0 & 0 & s_3 & 0 & 0 & 0 & c_3 & 0 \\ 0 & 0 & 0 & s_4 & 0 & 0 & 0 & c_4 \end{bmatrix} \tag{27}$$

where $c_i \coloneqq \cos\theta_i$ and $s_i \coloneqq \sin\theta_i$.

G Training Hyperparameters

Detailed hyperparameter configurations are provided in Tables 11 and 12.

Table 11: GLUE (DeBERTa-v3-base) hyperparameters per task and SOARA variant ("–" indicates not applicable).

Dataset / Variant	λ_{ortho}	WD	LR	r	Epochs	BS	Total cycles
MRPC / V2b	–	3.1×10^{-4}	3.6×10^{-2}	768	20	32	3.0
QNLI / V2b	–	5.8×10^{-4}	1.0×10^{-3}	768	15	32	3.0
RTE / V2b	–	1.1×10^{-4}	7.9×10^{-3}	768	40	32	3.0
CoLA / V2b	–	2.1×10^{-4}	7.4×10^{-3}	768	25	32	3.0
STS-B / V2b	–	4.9×10^{-3}	8.6×10^{-3}	768	19	32	3.0
SST-2 / V2b	–	1.9×10^{-4}	2.2×10^{-3}	768	19	32	3.0
MNLI / V2b	–	1.6×10^{-3}	2.5×10^{-3}	768	15	32	3.0
QQP / V2b	–	1.2×10^{-4}	2.3×10^{-3}	768	19	32	3.0
MRPC / V1	2.1×10^{-2}	1.5×10^{-4}	3.2×10^{-4}	64	15	32	–
QNLI / V1	3.0×10^{-2}	1.2×10^{-5}	9.5×10^{-5}	64	10	16	–
RTE / V1	4.6×10^{-3}	3.3×10^{-4}	4.0×10^{-4}	64	50	32	–
CoLA / V1	2.1×10^{-2}	4.2×10^{-4}	2.3×10^{-4}	64	20	16	–
STS-B / V1	3.8×10^{-2}	1.9×10^{-5}	4.2×10^{-4}	64	25	16	–
SST-2 / V1	4.5×10^{-2}	1.7×10^{-5}	8.3×10^{-5}	85	15	32	–
MNLI / V1	4.7×10^{-2}	5.4×10^{-5}	1.5×10^{-4}	85	11	32	–
QQP / V1	5.7×10^{-2}	2.6×10^{-4}	1.8×10^{-4}	85	15	32	–
MRPC / V2a	–	5.7×10^{-4}	9.5×10^{-4}	85	15	32	5.0
QNLI / V2a	–	2.2×10^{-5}	4.7×10^{-4}	85	12	32	5.0
RTE / V2a	–	3.2×10^{-3}	7.9×10^{-4}	85	50	32	5.0
CoLA / V2a	–	1.3×10^{-3}	8.7×10^{-4}	85	20	32	5.0
STS-B / V2a	–	1.0×10^{-4}	9.8×10^{-4}	85	25	32	5.0
SST-2 / V2a	–	8.1×10^{-5}	9.0×10^{-4}	85	15	32	5.0
QQP / V2a	–	4.3×10^{-4}	3.0×10^{-4}	85	15	32	5.0

For GSM8K (Gemma-7B), both SOARA-V1 (r=128) and SOARA-V2b (r=3072) use: 3 epochs, per-device batch size 1 (effective 128), cosine scheduler with 0.03 warmup ratio, weight decay 0.0, Adam ϵ=1×10^{-10}, λ_{ortho}=0.

References

Sara Babakniya, Ahmed Elkordy, Yahya Ezzeldin, Qingfeng Liu, Kee-Bong Song, Mostafa El-Khamy, and Salman Avestimehr. SLoRA: Federated parameter efficient fine-tuning of language models. International Workshop on Federated Learning in the Age of Foundation Models (NeurIPS Workshop), 2023.

Afonso S Bandeira and Ramon van Handel. Sharp nonasymptotic bounds on the norm of random matrices with independent entries. *THE ANNALS of PROBABILITY*, 44(4):2479–2506, 2016. doi: 10.1214/15-AOP1025.

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In Smaranda Muresan, Preslav Nakov, and Aline Villavicencio (eds.), *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pp. 1–9, Stroudsburg, PA, USA, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-short.1.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *Computer Vision – ECCV 2014*, pp. 446–461, Cham, 2014. Springer International Publishing. doi: 10.1007/978-3-319-10599-4_29.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3606–3613, Los Alamitos, CA, USA, June 2014. IEEE.

Table 12: Vision hyperparameters per dataset and SOARA variant ("–" indicates not applicable).

Dataset / Variant	λ_{ortho}	WD	LR	r	Epochs	BS	Total cycles
CIFAR100 / V2b	–	–	2.0×10^{-4}	768	10	16	3.0
DTD / V2b	–	–	9.0×10^{-4}	768	40	32	3.0
SUN397 / V2b	–	–	2.6×10^{-4}	768	20	32	3.0
FER2013 / V2b	–	–	4.3×10^{-4}	768	10	32	3.0
FGVCAircraft / V2b	–	–	1.9×10^{-3}	768	30	32	3.0
Food101 / V2b	–	–	1.9×10^{-3}	768	10	16	3.0
Resisc45 / V2b	–	–	3.2×10^{-3}	768	10	16	3.0
CIFAR100 / V1 (r=16)	4.6×10^{-2}	2.7×10^{-4}	1.5×10^{-4}	16	10	32	–
Food101 / V1 (r=16)	4.3×10^{-2}	9.2×10^{-4}	5.8×10^{-4}	16	10	32	–
Resisc45 / V1 (r=16)	2.7×10^{-2}	7.2×10^{-4}	9.9×10^{-4}	16	10	32	–
DTD / V1 (r=16)	1.7×10^{-3}	1.4×10^{-4}	9.8×10^{-4}	16	40	32	–
SUN397 / V1 (r=16)	4.8×10^{-2}	6.9×10^{-4}	1.9×10^{-4}	16	10	32	–
FER2013 / V1 (r=16)	4.6×10^{-2}	1.1×10^{-4}	6.0×10^{-4}	16	10	32	–
FGVCAircraft / V1 (r=16)	1.9×10^{-3}	1.9×10^{-5}	9.1×10^{-4}	16	30	32	–
CIFAR100 / V2a (r=16)	–	–	4.3×10^{-4}	16	10	32	3.0
DTD / V2a (r=16)	–	–	4.7×10^{-4}	16	40	32	3.0
FER2013 / V2a (r=16)	–	–	6.0×10^{-3}	16	10	32	3.0
FGVCAircraft / V2a (r=16)	–	–	3.3×10^{-3}	16	30	32	3.0
SUN397 / V2a (r=16)	–	–	2.6×10^{-4}	16	10	32	3.0
Food101 / V2a (r=16)	–	–	3.4×10^{-4}	16	10	32	3.0
Resisc45 / V2a (r=16)	–	–	4.1×10^{-2}	16	10	32	3.0

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems, 2021.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. QLoRA: Efficient finetuning of quantized LLMs. In *Thirty-seventh Conference on Neural Information Processing Systems*, Red Hook, NY, USA, 2023. Curran Associates, Inc.

Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, and Baining Guo. Cswin transformer: A general vision transformer backbone with cross-shaped windows. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12114–12124, Los Alamitos, CA, USA, 2022. IEEE.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, Virtual, 2021. OpenReview.net.

Abdelrahman Elsayed, Sarim Hashmi, Mohammed Elseiagy, Hu Wang, Mohammad Yaqub, and Ibrahim Almakky. Salt: Parameter-efficient fine-tuning via singular value adaptation with low-rank transformation. In *Proceedings of the British Machine Vision Conference (BMVC)*, London, UK, 2025. BMVA Press.

Ian J. Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, Radu Ionescu, Marius Popescu, Cristian Grozea, James Bergstra, Jingjing Xie, Lukasz Romaszko, Bing Xu, Zhang Chuang, and Yoshua Bengio. Challenges in representation learning: A report on three machine learning contests, 2013.

Ligong Han. SVDiff: Compact parameter space for diffusion fine-tuning. In *IEEE/CVF International Conference on Computer Vision (CVPR)*, Los Alamitos, CA, USA, 2023. IEEE.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. Debertav3: Improving deberta using electra-style pre-training with gradient-disentangled embedding sharing. arXiv preprint arXiv:2111.09543, 2021.

Xuehai He, Chuanyuan Li, Pengchuan Zhang, Jianwei Yang, and Xin Eric Wang. Parameter-efficient model adaptation for vision transformers. In *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, AAAI'23/IAAI'23/EAAI'23, Palo Alto, CA, USA, 2023. AAAI Press. ISBN 978-1-57735-880-0. doi: 10.1609/aaai.v37i1.25160.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2790–2799, Long Beach, CA, USA, 2019. PMLR.

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*, Virtual, 2022. OpenReview.net.

Soroush Abbasi Koohpayegani, Navaneet K L, Parsa Nooralinejad, Soheil Kolouri, and Hamed Pirsiavash. NOLA: Compressing loRA using linear combination of random basis. In *The Twelfth International Conference on Learning Representations*, Virtual, 2024. OpenReview.net.

Dawid J. Kopiczko, Tijmen Blankevoort, and Yuki M. Asano. VeRA: Vector-based random matrix adaptation. In *International Conference on Learning Representations (ICLR)*, Virtual, 2024. OpenReview.net.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report 0, University of Toronto, Toronto, Ontario, 2009.

Rafał Latała. Some estimates of norms of random matrices. *Proceedings of the American Mathematical Society*, 133(5):1273–1282, 2005. doi: 10.1090/S0002-9939-04-07800-1.

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*, pp. 3045–3059, Stroudsburg, PA, USA, 2021. Association for Computational Linguistics.

Vijay Lingam. SVFT: Parameter-efficient fine-tuning with singular vectors. In *Advances in Neural Information Processing Systems*, Red Hook, NY, USA, 2024. Curran Associates, Inc.

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 *Proceedings of the 41st International Conference on Machine Learning*, Vienna, Austria, 2024a. Proceedings of Machine Learning Research (PMLR).

Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, Yandong Wen, Michael J. Black, Adrian Weller, and Bernhard Schölkopf. Parameter-efficient orthogonal finetuning via butterfly factorization. In *International Conference on Learning Representations (ICLR)*, Virtual, 2024b. OpenReview.net.

Xinyu Ma, Xu Chu, Zhibang Yang, Yang Lin, Xin Gao, and Junfeng Zhao. Parameter efficient quasi-orthogonal fine-tuning via givens rotation. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org, 2024.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft, 2013.

Fanxu Meng. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems*, Red Hook, NY, USA, 2024. Curran Associates, Inc.

Zelin Peng, Zhengqin Xu, et al. SAM-PARSER: Fine-tuning SAM efficiently by parameter space reconstruction. In *Advances in Neural Information Processing Systems (NeurIPS)*, Red Hook, NY, USA, 2022. Curran Associates, Inc.

Zeju Qiu, Weiyang Liu, Haiwen Feng, Yuxuan Xue, Yao Feng, Zhen Liu, Dan Zhang, Adrian Weller, and Bernhard Schölkopf. Controlling text-to-image diffusion by orthogonal finetuning. In *Advances in Neural Information Processing Systems (NeurIPS)*, Red Hook, NY, USA, 2023. Curran Associates, Inc.

Zeju Qiu, Weiyang Liu, Adrian Weller, and Bernhard Schölkopf. Orthogonal finetuning made scalable. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 31946–31963, Stroudsburg, PA, USA, 2025. Association for Computational Linguistics.

Andreas Rücklé, Gregor Geigle, Max Glockner, Tilman Beck, Jonas Pfeiffer, Nils Reimers, and Iryna Gurevych. AdapterDrop: On the efficiency of adapters in transformers. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih (eds.), *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 7930–7946, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.626.

Yanpeng Sun, Qiang Chen, Xiangyu He, et al. Singular value fine-tuning: Few-shot segmentation requires few-parameters fine-tuning. In *AAAI Conference on Artificial Intelligence*, Palo Alto, CA, USA, 2024. AAAI Press.

Gemma Team. Gemma: Open models based on gemini research and technology, 2024.

Ihsan Ullah, Dustin Carrion, Sergio Escalera, Isabelle Guyon, Mike Huisman, Felix Mohr, Jan N. van Rijn, Haozhe Sun, Joaquin Vanschoren, and Phan Anh Vu. Meta-album: Multi-domain meta-dataset for few-shot image classification. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks*, NIPS'22, Red Hook, NY, USA, 2022. Curran Associates, Inc.

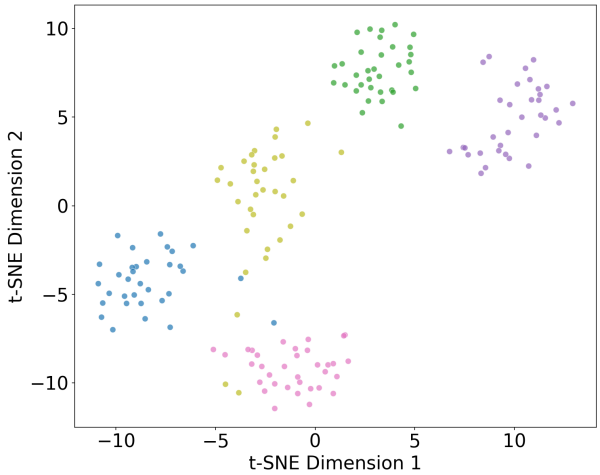
Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pp. 353–355, Brussels, Belgium, 2018. Association for Computational Linguistics.

Fei Wu, Jia Hu, Geyong Min, and Shiqiang Wang. Efficient orthogonal fine-tuning with principal subspace adaptation. In *The Fourteenth International Conference on Learning Representations*, 2026. URL `https://openreview.net/forum?id=FSHrinMArK`.

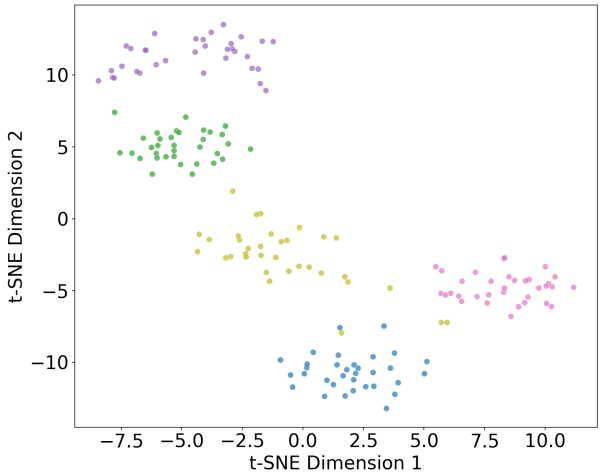
Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 3485–3492, Los Alamitos, CA, USA, 2010. IEEE. doi: 10.1109/CVPR.2010.5539970.

Fangzhao Zhang and Mert Pilanci. Spectral adapter: Fine-tuning in spectral space. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 130819–130847. Curran Associates, Inc., 2024. doi: 10.52202/079017-4158. URL `https://proceedings.neurips.cc/paper_files/paper/2024/file/ec2b1931cbda8e4c1a601ff5ff81c4a6-Paper-Conference.pdf`.

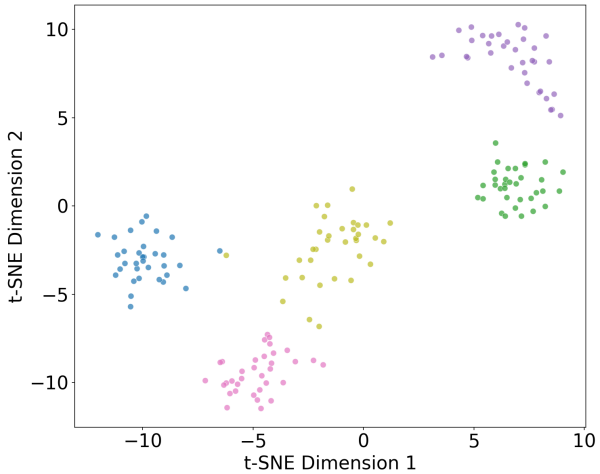
Qingru Zhang. AdaLoRA: Adaptive budget allocation for parameter-efficient fine-tuning. In *International Conference on Learning Representations*, Virtual, 2023. OpenReview.net.



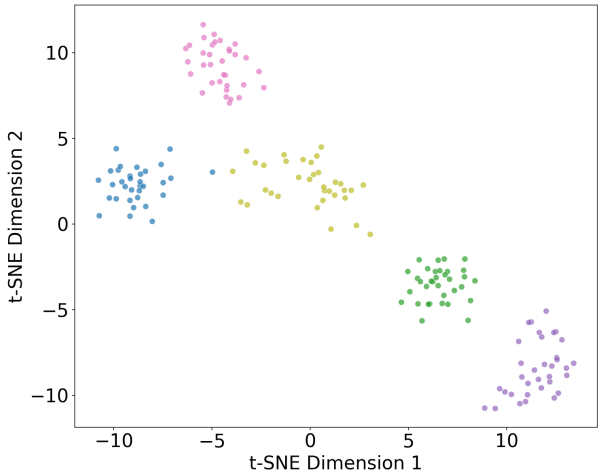
(a) SOARA-V1 ($r = 2$)



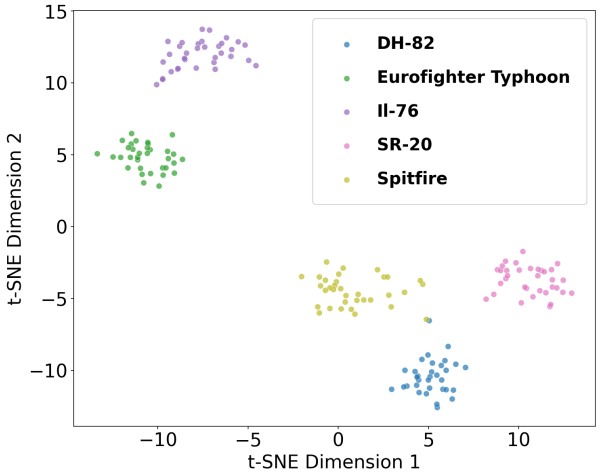
(b) SOARA-V1 ($r = 4$)



(c) SOARA-V1 ($r = 8$)



(d) SOARA-V1 ($r = 16$)



(e) SOARA-V2b

Figure 4: SOARA ablation on FGVC Aircraft: t-SNE visualization of intermediate ViT representations (CLS token) for five selected classes.