On the Representation Geometry of LoRA Model Merging

Chenyang Lu[*1], **Jiaru Li**[3*], **Jinman Zhao**[†1,2],
Xinran Chen[4], **Yining Wang**[2], **Renyi Cai**[1], **Yuchen Li**[4], **Chao He**[5]
[1]Yankun Inc. [2]University of Toronto,
[3]Northwestern University, [4]Baidu Inc. [5]University of Oxford

Abstract

Low-Rank Adaptation (LoRA) is widely used for parameter-efficient fine-tuning, yet merging multiple task-specific LoRA updates without additional training remains challenging. Most existing LoRA merging methods rely on SVD-based alignment, which emphasizes globally shared structure across tasks. In this work, we show that LoRA merging performance can be further improved by combining SVD with CUR decomposition. Through a representation-level analysis, we find that SVD-based decompositions primarily model shared components across tasks, while CUR-based decompositions better preserve task-specific and localized updates. These two perspectives are geometrically misaligned and exhibit complementary advantages, revealing an inherent trade-off between capturing shared structure and preserving task-specific information in LoRA model merging. Guided by this analysis, we propose a training-free merging procedure that explicitly combines the shared structure captured by SVD with the task-specific components preserved by CUR. Experiments on both vision and language benchmarks demonstrate consistent improvements over existing gradient-free LoRA merging methods. Our code is available at https://github.com/lcytoronto/knots_cur.

1 Introduction

Low-Rank Adaptation (LoRA; Hu et al., 2023) has become a widely adopted parameter-efficient fine-tuning (Han et al., 2024) paradigm for adapting large pre-trained models across both language (Dubey et al., 2024) and vision (Radford et al., 2021) domains. In practice, models are often fine-tuned independently for multiple downstream tasks. Supporting multi-task inference, continual learning, or modular deployment therefore requires the ability to merge multiple task-specific LoRA updates into a single model without performing additional fine-tunings.

Most existing LoRA model merging methods (Ilharco et al., 2023a; Yadav et al., 2023) share a common design principle: task-specific updates are first aligned into a shared representation space, the merging operation is then performed. Under this paradigm, KnOTS (Stoica et al., 2025) further formalizes the alignment process as a matrix decomposition problem, constructing a shared low-dimensional representation basis for LoRA updates via singular value decomposition (SVD). In this representation, updates from different tasks are projected onto the same set of basis vectors, thereby achieving consistent alignment across tasks. This class of approaches implicitly assumes that a single global representation basis is sufficient to capture the structure shared across tasks, serving as a unified space for model merging. In other training, transport-based fine-grained matching has also been shown to alleviate the limitations of coarse global alignment (Zhang et al., 2025c; Wu et al., 2025).

In this work, we revisit LoRA model merging from a representation geometry perspective. Rather than immediately proposing a new merging algorithm, we focus on examining the representation bases and geometric structures induced by existing alignment and decomposition strategies, with the goal of understanding the assumptions they make about LoRA updates. Through a systematic analysis of LoRA-induced weight updates, we uncover three key empirical observations. First, the representation bases induced by SVD and column-selection-based CUR decompositions are systematically misaligned at the subspace level, indicating that they capture different directions in parameter space. Second, the two decompositions exhibit complementary advantages, with SVD emphasizing globally shared structure and CUR better preserving localized, task-specific components. Fi-

[*]Equally Contribution
[†]Corresponding Author

Findings of the Association for Computational Linguistics: ACL 2026, pages 5289–5304

nally, we show that these observations admit a simple theoretical interpretation based on separating shared and task-specific structures in LoRA updates, motivating a closer examination of the representation assumptions underlying existing LoRA merging paradigms.

To address the representation-level limitations identified above, we take an analysis-driven approach to LoRA model merging that reconsiders how task-specific updates should be aligned and combined. Rather than assuming a single shared representation basis, our work explicitly examines the geometric and energetic structure of LoRA-induced weight updates and its implications for model merging. Our main contributions are:

1. We present a systematic representation-level analysis, showing that commonly used alignment strategies rely on a strong single-subspace assumption that is not supported by the geometry of LoRA updates. Through subspace overlap and energy decomposition analyses, we demonstrate that different decomposition strategies prioritize either globally shared representations or task-specific, distinctive components, revealing an inherent trade-off between sharedness and specificity.

2. Guided by these insights, we propose a merging procedure that explicitly separates and recombines globally shared and task-specific structures in LoRA updates. Our method remains fully training-free.

3. We conduct extensive experiments on both vision and language benchmarks, covering a diverse set of datasets and model architectures. The results show that our approach consistently improves merging performance over existing baselines, validating the practical benefits of revisiting representation assumptions.

2 Related Work

Parameter-Efficient Fine-Tuning Parameter-efficient fine-tuning (PEFT) aims to adapt large pre-trained models to downstream tasks by updating only a small subset of parameters, thereby reducing training and storage costs. A foundational PEFT approach is Adapter Tuning (Houlsby et al., 2019) that inserts lightweight task-specific layers while keeping the backbone model frozen. BitFit (Ben Zaken et al., 2022) further demonstrates that even updating only bias terms can yield competitive performance, highlighting the redundancy of full-parameter adaptation. Beyond weight-based adaptations, prompt-based methods such as Prompt Tuning (Lester et al., 2021; Liu et al., 2022; Xiao et al., 2026b) and Prefix-Tuning (Li and Liang, 2021) adapt models by learning task-specific inputs rather than modifying model parameters. Several approaches have been proposed to improve the efficiency of AI systems, including layer-specific tuning (Fan et al., 2025a,b), model pruning (Yu et al., 2025), data efficiency (Zhao et al., 2026; Zhou et al., 2024b,a), long-context modeling (Li et al., 2025b,c; Shi et al., 2026), knowledge distillation (Dong et al., 2026b) and optimizer (Guan et al., 2025).

LoRA and its Variations Among PEFT methods, Low-Rank Adaptation (LoRA) (Hu et al., 2022) has emerged as one of the most widely adopted approaches due to its simplicity and effectiveness. Following its success, numerous extensions of LoRA have been proposed to improve efficiency, expressivity, or stability. AdaLoRA (Zhang et al., 2023b) dynamically adjusts the rank during fine-tuning based on parameter importance, while DyLoRA (Valipour et al., 2023) and related methods (Liu et al., 2024; Wang et al., 2024) introduce dynamic or sparse updates to the low-rank matrices. LoRA+ (Hayou et al., 2024) shows that optimal training requires asymmetric learning rates for the two low-rank factors, with significantly smaller updates for the projection matrix. LoRA-FA (Zhang et al., 2023a) further explores this direction by freezing the projection matrix and updating only the expansion matrix. Other variants extend LoRA along orthogonal dimensions, including integrating Mixture-of-Experts mechanisms (Luo et al., 2024; Dou et al., 2024; Qing et al., 2024) and decomposing adaptation matrices into finer-grained blocks (Ren et al., 2024; Mao et al., 2024; Tian et al., 2024). A broad spectrum of additional LoRA-based methods has also been proposed (Li et al., 2025a; Xiao et al., 2026a; Zhang et al., 2025d; Zhao et al., 2025a; Zhong and Zhou, 2024, *inter alia*).

Model Merging Model merging aims to combine multiple task-specific models into a single model without additional training. For fully fine-tuned models, simple linear merging techniques such as task arithmetic (Ilharco et al., 2023b; Dong et al., 2026a) and its extensions (Yadav et al., 2023;

Yu et al., 2024) have been shown to work surprisingly well, a phenomenon often attributed to mode connectivity (Garipov et al., 2018; Draxler et al., 2018) and implicit weight disentanglement across tasks (Ortiz-Jimenez et al., 2023). However, these assumptions do not readily transfer to peft settings. In particular, merging LoRA-adapted models is substantially more challenging, as low-rank constraints induce task-specific subspaces that are often misaligned, leading to severe interference under naive merging (Tang et al., 2025). Recent work addresses this by explicitly modeling representation structure: KnOTS (Stoica et al., 2025) aligns task updates via an SVD-constructed shared basis, LoRA-LEGO (Zhao et al., 2025b) reduces interference through rank-wise decomposition and clustering, and LoRI (Zhang et al., 2025a) enforces approximate subspace orthogonality during training. Additional methods improve compositionality (Yin and Wang, 2025) through sparsification or structured merging (Yadav et al., 2023; Yu et al., 2024; Miyano and Arase, 2025; Prabhakar et al., 2025). Despite these advances, most approaches treat the construction of the shared representation basis as fixed or method-specific, focusing primarily on how task-specific components are combined.

3 Motivation and Background

3.1 Problem Statement

LoRA-based Parameter-Efficient Fine-Tuning LoRA (Hu et al., 2022) has become one of the most widely adopted PEFT techniques due to its simplicity, efficiency, and strong empirical performance.

Instead of updating all parameters of a pre-trained model, LoRA injects a small number of trainable parameters by constraining task-specific updates to be low-rank. Concretely, consider a pre-trained weight matrix $W^{pt} \in \mathbb{R}^{d\times k}$ in a neural network layer. LoRA models its task-specific update as a low-rank decomposition:

$$\Delta W = \underline{B}\underline{A}, \tag{1}$$

where $\underline{B} \in \mathbb{R}^{d\times r}$ and $\underline{A} \in \mathbb{R}^{r\times k}$ are trainable matrices with rank $r \ll \min(d,k)$. We use underlines to denote *trainable* parameters introduced by LoRA. The adapted model parameters are then given by

$$W' = W^{pt} + \Delta W = W^{pt} + \underline{B}\underline{A}, \tag{2}$$

where the original pre-trained weights W^{pt} remain frozen during fine-tuning.

LoRA Model Merging Consider a pre-trained model with l layers, parameters are denoted by

$$\theta^{pt} = \{W_1^{pt}, \ldots, W_j^{pt}, \ldots, W_l^{pt}\}. \tag{3}$$

Suppose the model is fine-tuned using LoRA on n different downstream tasks, producing n task-specific models. For the i-th task, LoRA induces a set of weight updates

$$\tau^{(i)} = \{\Delta W_1^{(i)}, \ldots, \Delta W_j^{(i)}, \ldots, \Delta W_l^{(i)}\}, \tag{4}$$

where $\Delta W_j^{(i)} \in \mathbb{R}^{d\times k}$ denotes the low-rank update applied to the j-th layer. The parameters of the i-th LoRA-finetuned model are therefore given by

$$\begin{aligned}\theta^{(i)} &= \theta^{pt} + \tau^{(i)} \\ &= \{W_1^{pt} + \Delta W_1^{(i)}, \ldots, W_l^{pt} + \Delta W_l^{(i)}\}.\end{aligned} \tag{5}$$

Given a collection of LoRA-finetuned models $\{\theta^{(1)}, \ldots, \theta^{(n)}\}$, the goal of *LoRA model merging* is to construct a ***single*** merged model by combining their task-specific updates. Formally, this amounts to producing a merged set of updates

$$\begin{aligned}&\tau^{(\text{merged})} \\ =&\{\Delta W_1^{(\text{merged})}, \ldots, \Delta W_l^{(\text{merged})}\},\end{aligned} \tag{6}$$

and the corresponding merged parameters without additional fine-tuning as

$$\theta^{(\text{merged})} = \theta^{pt} + \tau^{(\text{merged})}. \tag{7}$$

3.2 SVD-based Alignment

KnOTS (Stoica et al., 2025) is a representative for merging LoRA-finetuned models. For a fixed layer j, KnOTS first concatenates the task updates along the column dimension and applies SVD:

$$\begin{aligned}&\text{SVD}\big([\Delta W_j^{(1)}; \ldots; \Delta W_j^{(n)}]\big) \\ &= U_j \Sigma_j V_j^T = U_j \Sigma_j \begin{bmatrix} V_j^{(1)} & \cdots & V_j^{(n)} \end{bmatrix}^\top\end{aligned} \tag{8}$$

where U_j and Σ_j are shared across all tasks, and each $V_j^{(i)}$ corresponds to the task-specific component of $\Delta W_j^{(i)}$ in the aligned space. Under this decomposition, each update can be written as

$$\Delta W_j^{(i)} = U_j \Sigma_j (V_j^{(i)})^\top. \tag{9}$$

KnOTS then applies an existing model merging method (e.g., TIES) directly to $\{V_j^{(1)}, \ldots, V_j^{(n)}\}$

to obtain a merged representation $V_j^{(\text{merged})}$. The merged update is reconstructed as

$$\Delta W_j^{(\text{merged})} = U_j \Sigma_j (V_j^{(\text{merged})})^\top \tag{10}$$

and added to the corresponding pre-trained weights W_j^{pt}. By construction, all task-specific updates are projected into a shared representation space defined by U_j, ensuring alignment across tasks.

3.3 Motivation

All the experiments in this section are based on LoRA Adapters released by Stoica et al. (2025).

3.3.1 Alternative Decompositions

By construction, SVD produces an orthonormal basis that captures directions with the largest global variance under a fixed rank constraint. Beyond SVD, alternative decompositions exist that impose different structural biases. One such approach is CUR decomposition that approximates a matrix ΔW as

$$\Delta W \approx CUR, \tag{11}$$

where C and R consist of subsets of columns and rows of ΔW. Unlike SVD, CUR induces sparse and data-dependent representation bases.

3.3.2 Geometric Misalignment

Observation 1: *SVD and CUR induce systematically different representation bases that are geometrically misaligned at the subspace level.*

In our analysis, SVD constructs a dense orthonormal basis that captures globally dominant directions of ΔW, while CUR induces a sparse, data-dependent basis by selecting columns with the largest ℓ_2 norms. Specifically, for ViT-B/32, we concatenate LoRA updates (i.e., ΔW) across attention submodules and select the top-k columns by norm to form the CUR matrix C, which is then orthonormalized via QR decomposition.

To quantify the alignment between the bases induced by SVD and CUR, we measure the overlap between their corresponding column spaces. Given the matrices inducing the SVD- and CUR-based representations, we first extract orthonormal bases of their column spaces via QR decomposition. Let $Q_1 \in \mathbb{R}^{d\times k}$ and $Q_2 \in \mathbb{R}^{d\times k}$ denote the resulting bases. We then compute the singular values of $Q_1^\top Q_2$ by:

$$Q_1^\top Q_2 = U \Sigma V^\top \tag{12}$$

where the singular values $\{\sigma_i\}_{i=1}^k$ correspond to the cosines of the principal angles between the two subspaces. We define the subspace overlap as:

$$\text{Overlap}(Q_1, Q_2) = \frac{1}{k} \sum_{i=1}^{k} \sigma_i^2, \tag{13}$$

which equals 1 when the two subspaces coincide and decreases as they become more misaligned.

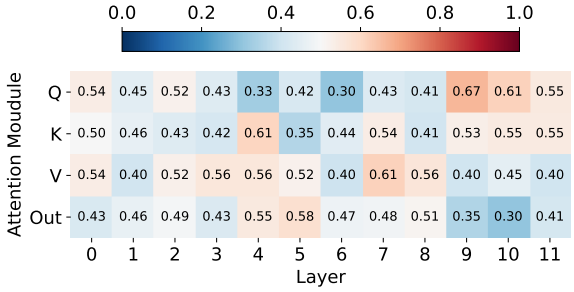

Figure 1: Overlap ratios of LoRA updates across attention submodules for each transformer layer..

As shown in Figure 1, the overlap remains consistently low across layers and attention modules. This indicates that the two bases span largely different directions in parameter space, reflecting different and potentially complementary representation biases in modeling LoRA updates.

3.3.3 Complementary Energy Capture

Observation 2: *CUR and SVD decomposition provide complementary updates.*

The SVD-based merging method introduces a strong bias toward globally aligned directions, which can underrepresent task-discriminative or localized structures in $\Delta W^{(t)}$. In particular, when a task update is concentrated in specific rows or columns that are not well aligned with dominant global components, such information may be attenuated by SVD projection.

To quantify this information loss during the SVD merging procedure, we measure the change in task-specific update energy after SVD projection: given a shared SVD subspace spanned by U_r, we measure the energy captured by this subspace as

$$E_{\text{SVD}}(\Delta W) = \|U_r U_r^\top \Delta W\|_F^2, \tag{14}$$

and define energy loss as

$$E_{\text{res}}(\Delta W) = \|\Delta W\|_F^2 - \|U_r U_r^\top \Delta W\|_F^2. \tag{15}$$

Table 1 reports the original energy of $\Delta W_j(t)$, SVD-project energy E_{SVD}, energy loss (O-S), and the fraction of total energy captured, across datasets. The results show that the energy capture

Dataset	Original Energy	SVD Energy	O-S	Energy Ratio
Cars	31.69	30.34	1.35	0.958
DTD	2.32	1.56	0.76	0.671
EuroSAT	19.40	18.03	1.37	0.929
GTSRB	1.59	1.22	0.37	0.768
MNIST	21.89	20.58	1.31	0.940
RESISC45	1.70	1.34	0.36	0.789
SUN397	4.85	4.28	0.57	0.881
SVHN	37.06	35.96	1.10	0.958

Table 1: Energy retained by SVD projection.

Dataset	Original Energy	Projected Energy	O-C	Energy Ratio
Cars	31.69	31.48	0.21	0.993
DTD	2.32	2.18	0.14	0.939
EuroSAT	19.40	19.20	0.20	0.990
GTSRB	1.59	1.49	0.09	0.942
MNIST	21.89	21.70	0.19	0.991
RESISC45	1.70	1.61	0.09	0.945
SUN397	4.85	4.72	0.14	0.972
SVHN	37.06	36.88	0.18	0.995

Table 2: Energy retained by CUR projection.

fraction is at 95.8%, with the absolute energy loss larger than 0.36.

CUR matrix approximation, in contrast, represents a matrix using a subset of its original rows and columns. As a result, CUR tends to preserve localized or task-specific structures in $\Delta W_j^{(t)}$ that may be attenuated by dense low-rank projections. We perform an analogous energy analysis for CUR-based projections, with results shown in Table 2. Note that CUR preserves a significantly higher fraction of the energy (> 0.939) and causes much smaller absolute energy loss (O-C<0.21) than SVD. Taken together with Observation 1, these results highlight a clear difference in the representation biases induced by SVD and CUR.

3.3.4 Hybrid Combination is Never Worse

Theorem 1: *Let the optimal merged update be ΔW^*, and let the SVD- and CUR-based merged updates be $\Delta W^{(\mathrm{SVD})}$ and $\Delta W^{(\mathrm{CUR})}$, respectively. Define their errors (with respect to ΔW^*) as*

$$\epsilon_s = \Delta W^* - \Delta W^{(\mathrm{SVD})}, \tag{16}$$

$$\epsilon_c = \Delta W^* - \Delta W^{(\mathrm{CUR})}. \tag{17}$$

Then, under the conditions that the two errors are non-positively correlated:

$$\langle \epsilon_s, \epsilon_c \rangle_F \leq 0, \tag{18}$$

then we can construct a hybrid update:

$$\Delta W = \alpha \Delta W^{(\mathrm{SVD})} + (1-\alpha)\Delta W^{(\mathrm{CUR})}, \tag{19}$$

such that, for some $\alpha \in [0,1]$, ΔW has a smaller (or equal) Frobenius error than using $\Delta W^{(\mathrm{SVD})}$ or $\Delta W^{(\mathrm{CUR})}$ alone when they represent ΔW^*.

Proof. We may express the two non-hybrid updates as

$$\Delta W^{(\mathrm{SVD})} = \Delta W^* - \epsilon_s, \tag{20}$$

$$\Delta W^{(\mathrm{CUR})} = \Delta W^* - \epsilon_c, \tag{21}$$

so the hybrid update can be written as

$$\Delta W = \Delta W^* - \alpha\epsilon_s + (1-\alpha)\epsilon_c. \tag{22}$$

The square-norm error for the hybrid method then becomes

$$||\epsilon_\alpha||^2 = ||\Delta W^* - \Delta W||^2 \tag{23}$$

$$= ||\alpha\epsilon_s + (1-\alpha)\epsilon_c||^2. \tag{24}$$

Under the condition that $\langle \epsilon_s, \epsilon_c \rangle_F \leq 0$, the last line gives

$$||\epsilon_\alpha||^2 \leq \alpha^2||\epsilon_s||^2 + (1-\alpha)^2||\epsilon_c||^2. \tag{25}$$

We can minimize the right-hand side by taking

$$\alpha = \frac{||\epsilon_c||^2}{||\epsilon_c||^2 + ||\epsilon_s||^2} \in [0,1], \tag{26}$$

so that inequality becomes

$$||\epsilon_\alpha||^2 \leq \frac{||\epsilon_s||^2||\epsilon_c||^2}{||\epsilon_s||^2 + ||\epsilon_c||^2} \tag{27}$$

$$\leq \min\left\{||\epsilon_s||^2, ||\epsilon_c||^2\right\}, \tag{28}$$

as the theorem predicts. The equality can be reached when either $||\epsilon_s||^2$ or $||\epsilon_s||^2$ is zero. □

Note that Theorem 1 serves to justify why a hybrid weighting can be beneficial when SVD and CUR capture complementary information.

4 Method

Motivated by our analysis on the representation assumptions in LoRA merging, we propose a merging procedure that explicitly accounts for both shared and task-specific structures in LoRA updates.

SVD and CUR Decomposition of LoRA Updates We compute the SVD-based merged update ΔW^{SVD} following Section 3.2. The CUR-based merged update is constructed in a parallel manner.

Method	Datasets								Average
	Cars	DTD	EuroSAT	GTSRB	MNIST	RESISC45	SUN397	SVHN	
Per-Task Absolute Accuracies of each finetuned model (%)									
Full Finetune*	74.0	58.3	99.0	92.7	99.3	88.4	64.5	96.2	84.1
Merged Models Normalized Per-Task Accuracies Against Finetuned Models (%)									
RegMean*	80.2	71.3	37.9	47.3	43.1	70.5	93.9	43.0	60.9
TA*	82.0	73.6	48.8	42.1	53.1	71.5	97.5	41.2	63.7
Fisher*	84.5	72.4	44.4	55.8	47.8	70.9	96.1	39.2	63.9
TIES	82.2	72.8	49.0	38.4	56.3	68.2	96.9	45.7	63.7
DARE-TIES	82.7	70.9	42.9	36.8	56.3	66.5	95.4	49.3	62.6
KnOTS-TIES	83.8	73.6	48.8	49.2	68.9	67.8	96.0	53.7	67.7
KnOTS-DARE-TIES	83.1	73.0	46.1	45.7	61.3	68.3	95.9	48.2	65.2
CoreSpace-TIES	84.7	**76.5**	52.2	**50.4**	67.4	**71.2**	96.5	50.2	68.6
LoRA-LEGO	82.4	72.5	42.4	41.8	51.0	68.4	96.9	40.0	62.0
Ours-TIES	**84.8**	73.5	**54.5**	41.8	**78.8**	69.5	**97.9**	**59.1**	**70.0**

Table 3: *ViT-B/32* per-task accuracies of merged models normalized against finetuned models (%). Results with * are taken from Stoica et al. (2025), rest of the results are replicated by ourselves.

We concatenate the task-specific updates $[\Delta W^{(1)}; \ldots; \Delta W^{(n)}]$ and apply CUR decomposition:

$$\begin{aligned} &\mathrm{CUR}\big([\Delta W^{(1)}; \ldots; \Delta W^{(n)}]\big) \\ =&C\left[U^{(1)}R^{(1)}; \ldots; U^{(n)}R^{(n)}\right]^{\top} \end{aligned} \tag{29}$$

where C consists of selected columns from the original updates, and each $R^{(i)}$ represents selected rows from the original updates. each $U^{(i)}R^{(i)}$ represents the task-specific coefficients in the CUR-aligned space. We then apply the same merging operator (TIES) to the set $\{U^{(1)}R^{(1)}, \ldots, U^{(n)}R^{(n)}\}$ to obtain a merged representation $(UR)^{\mathrm{merged}}$. The CUR-based merged update is reconstructed as

$$\Delta W^{\mathrm{CUR}} = C\big((UR)^{\mathrm{merged}}\big)^{\top}. \tag{30}$$

Residual Energy-Based Weighting We first consider an energy-ratio-based mixing coefficient defined by

$$\alpha_{\mathrm{energy}} = \frac{E^{\mathrm{SVD}}}{E^{\mathrm{SVD}} + E^{\mathrm{CUR}}}, \tag{31}$$

where E^{SVD} and E^{CUR} denote the residual energies of the SVD- and CUR-based merged updates, respectively. This formulation reflects the relative contribution of residual energy from each branch. To obtain a more stable weighting, we convert residual energy into update magnitude as:

$$M = \sqrt{E} = \|\Delta W\|_F, \tag{32}$$

where we take the square root of E because squared Frobenius norms can over-emphasize large-magnitude residuals (Golub and Van Loan, 2013). We then define the final mixing coefficient based on residual magnitudes as

$$\alpha = \frac{M^{\mathrm{SVD}}}{M^{\mathrm{SVD}} + M^{\mathrm{CUR}}}. \tag{33}$$

Final Merging The final merged update is computed as a convex combination of the two branches:

$$\Delta W = \alpha\, \Delta W^{\mathrm{SVD}} + (1-\alpha)\, \Delta W^{\mathrm{CUR}}. \tag{34}$$

5 Results

5.1 Experimental Setup

Benchmark Selection We conduct experiments on both vision and language benchmarks. For vision Transformers, we follow prior LoRA merging work and consider a diverse set of image classification datasets, including Cars (Krause et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2018), GTSRB (Stallkamp et al., 2011), MNIST (LeCun, 1998), RESISC45 (Cheng et al., 2017), SUN397 (Xiao et al., 2016), and SVHN (Netzer et al., 2011). For language models, we evaluate on multiple natural language inference (NLI) benchmarks, including SNLI (Bowman et al., 2015), MNLI (Williams et al., 2018), SICK (Marelli et al., 2014), QNLI (Wang et al., 2019), RTE (Wang et al., 2019), and SciTail (Khot et al., 2018). These datasets span different data

Method	Datasets						Average
	SNLI	MNLI	SICK	QNLI	RTE	SciTail	
Per-Task Absolute Accuracies (%)							
Full Finetune	92.5	90.3	91.6	94.5	89.9	96.5	92.2
Per-Task Accuracies of Merged Models (Normalized)							
TIES	92.3	96.1	66.2	80.3	96.0	94.9	87.6
DARE-TIES	**94.3**	95.1	87.6	69.2	97.8	96.8	90.6
KnOTS-TIES	90.3	93.5	**94.6**	80.0	98.6	97.9	92.9
KnOTS-DARE-TIES	93.3	94.1	87.8	70.7	**99.4**	96.6	90.8
CoreSpace-TIES	92.1	93.5	93.6	**83.7**	99.2	97.7	93.3
LoRA-LEGO	92.2	59.9	69.0	59.0	83.9	83.3	74.5
Ours-TIES	92.9	**94.8**	93.2	82.0	97.0	**97.9**	**93.5**

Table 4: *LLaMA-3 8B* per-task accuracies of merged models normalized against finetuned models (%).

Metric	Ensemble	TA	TIES	DARE-TIES	K-TIES	K-DARE-TIES	LoRA-LEGO	Ours
Hits@1	39.7	42.7	43.7	43.8	47.1	45.5	42.9	**47.9**
Hits@3	60.9	64.0	65.4	66.0	67.9	66.8	64.5	**68.6**
Hits@5	70.2	72.7	73.9	74.2	76.1	75.3	73.5	**76.2**

Table 5: Joint-task (Union) evaluation on eight vision benchmarks. We report Hits@k over the unified label space of 748 classes.

sources, label distributions, and reasoning requirements, and are commonly used to assess sentence-level semantic representations. Detailed statistics for all benchmarks are provided in Appendix A.

Model Selection For vision experiments, we adopt *ViT-B/32* (Radford et al., 2021) as the backbone architecture which provides a strong and stable baseline for image classification tasks. For language experiments, we use the *Llama-3-8B* (Dubey et al., 2024) model, a representative large-scale decoder-only language model that has been extensively adopted in recent studies. Both models are fine-tuned using LoRA and share a common pre-trained initialization within each modality.

Baseline Selection We compare our method against a range of representative gradient-free LoRA model merging baselines including Task Arithmetic (TA) (Ilharco et al., 2023a), RegMean (Jin et al., 2023), TIES (Yadav et al., 2023), DARE-TIES (Yu et al., 2024), KnOTS (Stoica et al., 2025), Corespace (Panariello et al.) and LoRA-LEGO (Zhao et al., 2025b) in our evaluation. These baselines cover a variety of merging strategies, including linear combination, sparsity-based merging, and subspace alignment methods. Detailed descriptions of all baseline methods are provided in Appendix B.

Training Detail Due to space limitation, see Appendix C for more information.

5.2 Main Results

ViT Results As shown in Table 3, our method achieves the best overall performance among all baselines on *ViT-B/32*, reaching the highest average normalized accuracy of 70%. Compared to existing gradient-free LoRA merging approaches, our method consistently improves or matches per-task performance across datasets, leading to a stronger merged model overall. Notably, our method substantially outperforms Fisher-based approaches reported in the table, which require additional training or task-specific statistics and are therefore not training-free. These results indicate that our approach more effectively preserves task-specific knowledge when merging models.

LLM Results Table 4 summarizes the results on *Llama-3 8B* across all NLI tasks. Our method achieves the best overall performance among all baselines, reaching the highest average normalized accuracy of 93.5%. In addition to the strong average performance, our method attains competitive or leading results on most of the individual tasks, demonstrating its effectiveness in merging LoRA-finetuned LLMs without additional training.

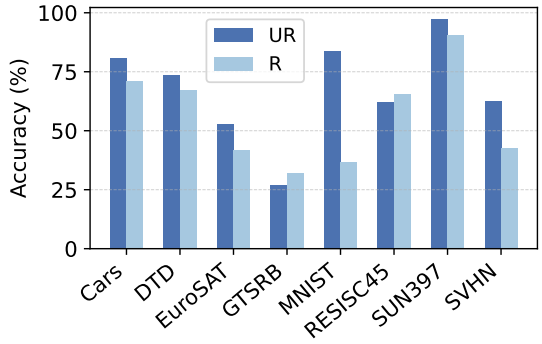

Figure 2: Comparison between merging in the update-representation space and merging in the reduced space after CUR decomposition.

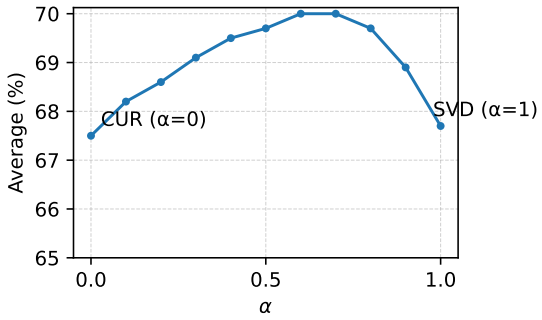

Figure 3: Effect of interpolating between CUR-based and SVD-based decompostions. The experiment is conducted on *VIT-B/32*.

5.3 Other Study

All experiments here are based on *VIT-B/32*.

Joint-Task Evaluation Beyond per-task evaluation, we further consider a more challenging *joint-task* setting to assess whether a merged model can function as a truly general model. Following KnOTS (Stoica et al., 2025), this setting evaluates merged models over the *union* of inputs and labels from all eight vision benchmarks, rather than evaluating each task independently. Specifically, we aggregate the label spaces of all eight datasets and remove duplicates, resulting in a unified label space with 748 unique classes. Each test image is then evaluated against this full label set, regardless of its original dataset.

Due to the large label space and semantic overlap between classes across datasets (e.g., fine-grained scene categories), we report performance using the Hits@k metric. Hits@k measures the fraction of examples for which the ground-truth label appears among the model's top-k predictions, with Hits@1 corresponding to standard accuracy. Table 5 reports the joint-task results for all merging methods. Detailed broken down are listed in Appendix E. Our method consistently outperforms all baselines across all Hits@k metrics. These results demonstrate that our approach produces a more effective general model under joint-task evaluation.

Merging UR vs. Merging R We further compare two ways of applying CUR-based merging after the CUR decomposition: merging in the update-representation space (UR) versus merging directly in the reduced space (R). In both settings, the CUR decomposition itself is identical; the only difference lies in the space where the merge operation is performed. As shown in Figure 2, merging in UR consistently outperforms merging in R across all vision benchmarks. The performance gap is particularly pronounced on tasks with strong task-specific characteristics, such as MNIST and SVHN. This indicates that performing merge directly in the reduced space introduces an additional information bottleneck, where task-discriminative directions are suppressed before the merging step.

In contrast, merging in the UR subspace preserves richer task-specific variations induced by the selected columns in CUR, thereby enabling the merge operation to more effectively balance shared and task-specific components. These results suggest that the effectiveness of CUR-based merging critically depends on the representation space in which merging is performed, with UR being a substantially more favorable choice than R.

Interpolation with CUR We study the effect of interpolating between SVD-based and CUR-based updates by varying the interpolation coefficient α. We sweep α from 0 to 1, where $\alpha = 0$ corresponds to a purely CUR-based update and $\alpha = 1$ corresponds to a purely SVD-based update. Figure 3 shows the average performance as a function of α on VIT. We observe a clear unimodal trend: performance improves as α increases from 0, reaches its peak around $\alpha \approx 0.6$, and then gradually degrades as α approaches 1. This suggests that combining CUR-style and SVD-style updates yields a stronger merged model than using either decomposition alone. Detailed per-dataset results are reported in Appendix D.

Combining with Alternative LoRA Merging Strategies We also explore combining our approach with alternative LoRA merging strategies LoRA-LEGO (Zhao et al., 2025b). Detailed exper-

imental results are reported in Appendix F. We did not observe consistent performance improvements surpassing our proposed method. We conjecture that this is because LoRA-LEGO primarily operates at the adapter composition level and does not explicitly address the representation-level trade-off.

6 Conclusions

We revisit LoRA model merging from a representation level perspective and examine the underlying assumptions in existing training-free merging methods. Our analysis shows that SVD-based alignment primarily captures globally shared structure across tasks while CUR-based decompositions better preserve task-specific and localized updates. Guided by this insight, we propose a simple training-free merging procedure that combines the complementary strengths of SVD and CUR. Experiments on both vision and language benchmarks demonstrate consistent improvements over existing LoRA merging baselines, highlighting the importance of accounting for both shared and task-specific structures in LoRA model merging.

7 Limitations

Our analysis focuses on the geometric structure of LoRA-induced parameter updates in weight space. While this representation-level view provides clear insights into shared and task-specific components, it does not explicitly account for non-linear interactions introduced by the forward pass or activation dynamics. Understanding how these geometric properties translate to function-level behavior remains an important direction for future work. Moreover, our study is primarily grounded in supervised fine-tuning (SFT) settings, and it is unclear whether the observed patterns generalize to other adaptation paradigms, such as reinforcement learning (Zhang et al., 2025b).

References

Elad Ben Zaken, Yoav Goldberg, and Shauli Ravfogel. 2022. BitFit: Simple parameter-efficient fine-tuning for transformer-based masked language-models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 1–9, Dublin, Ireland. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Gong Cheng, Junwei Han, and Xiaoqiang Lu. 2017. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105(10):1865–1883.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3606–3613.

Junhao Dong, Raoof Zare Moayedi, Yew-Soon Ong, and Seyed-Mohsen Moosavi-Dezfooli. 2026a. Allies teach better than enemies: Inverse adversaries for robust knowledge distillation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Junhao Dong, Cong Zhang, Xinghua Qu, Sua Qi Rong, Nguyen Duc Thai, Wenbo Pan, Xinfeng Li, Tongliang Liu, Piotr Koniusz, and Yew-Soon Ong. 2026b. Tug-of-war no more: Harmonizing accuracy and robustness in vision-language models via stability-aware task vector merging. In *The Fourteenth International Conference on Learning Representations*.

Shihan Dou, Enyu Zhou, Yan Liu, Songyang Gao, Wei Shen, Limao Xiong, Yuhao Zhou, Xiao Wang, Zhiheng Xi, Xiaoran Fan, Shiliang Pu, Jiang Zhu, Rui Zheng, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. LoRAMoE: Alleviating world knowledge forgetting in large language models via MoE-style plugin. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1932–1945, Bangkok, Thailand. Association for Computational Linguistics.

Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. 2018. Essentially no barriers in neural network energy landscape. In *International conference on machine learning*, pages 1309–1318. PMLR.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Yuchun Fan, Yongyu Mu, YiLin Wang, Lei Huang, Junhao Ruan, Bei Li, Tong Xiao, Shujian Huang, Xiaocheng Feng, and Jingbo Zhu. 2025a. SLAM: Towards efficient multilingual reasoning via selective language alignment. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 9499–9515, Abu Dhabi, UAE. Association for Computational Linguistics.

Yuchun Fan, Yilin Wang, Yongyu Mu, Lei Huang, Bei Li, Xiaocheng Feng, Tong Xiao, and JingBo Zhu. 2025b. Language-specific layer matters: Efficient

multilingual enhancement for large vision-language models. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 12473–12500, Suzhou, China. Association for Computational Linguistics.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P Vetrov, and Andrew G Wilson. 2018. Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in neural information processing systems*, 31.

Gene H Golub and Charles F Van Loan. 2013. *Matrix computations*. JHU press.

Yunchuan Guan, Yu Liu, Ke Zhou, Hui Li, Sen Jia, Zhiqi Shen, Ziyang Wang, Xinglin Zhang, Tao Chen, Jenq-Neng Hwang, and Lei Li. 2025. Learning an efficient optimizer via hybrid-policy sub-trajectory balance. *Preprint*, arXiv:2511.00543.

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

Soufiane Hayou, Nikhil Ghosh, and Bin Yu. 2024. LoRA+: Efficient low rank adaptation of large models. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 17783–17806. PMLR.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. 2018. Introducing eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. In *IGARSS 2018 - 2018 IEEE International Geoscience and Remote Sensing Symposium*, pages 204–207.

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

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. 2023. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5276, Singapore. Association for Computational Linguistics.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023a. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023b. Editing models with task arithmetic. In *The Eleventh International Conference on Learning Representations*.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2023. Dataless knowledge fusion by merging weights of language models. In *The Eleventh International Conference on Learning Representations*.

Tushar Khot, Ashish Sabharwal, and Peter Clark. 2018. Scitail: A textual entailment dataset from science question answering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1).

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 2013. 3d object representations for fine-grained categorization. In *2013 IEEE International Conference on Computer Vision Workshops*, pages 554–561.

Yann LeCun. 1998. The mnist database of handwritten digits. *http://yann. lecun. com/exdb/mnist/*.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Kaiyang Li, Shaobo Han, Qing Su, Wei Li, Zhipeng Cai, and Shihao Ji. 2025a. Uni-loRA: One vector is all you need. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Wenhao Li, Yuxin Zhang, Gen Luo, Haiyuan Wan, Ziyang Gong, Fei Chao, and Rongrong Ji. 2025b. Spotlight attention: Towards efficient llm generation via non-linear hashing-based kv cache retrieval. *Preprint*, arXiv:2508.19740.

Wenhao Li, Yuxin Zhang, Gen Luo, Daohai Yu, and Rongrong Ji. 2025c. Training long-context LLMs efficiently via chunk-wise optimization. In *ACL 2025 findings*.

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

Xiao Liu, Kaixuan Ji, Yicheng Fu, Weng Tam, Zhengxiao Du, Zhilin Yang, and Jie Tang. 2022. P-tuning: Prompt tuning can be comparable to fine-tuning across scales and tasks. In *Proceedings of the 60th*

Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 61–68, Dublin, Ireland. Association for Computational Linguistics.

Zeyu Liu, Souvik Kundu, Anni Li, Junrui Wan, Lianghao Jiang, and Peter Beerel. 2024. AFLoRA: Adaptive freezing of low rank adaptation in parameter efficient fine-tuning of large models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 161–167, Bangkok, Thailand. Association for Computational Linguistics.

Tongxu Luo, Jiahe Lei, Fangyu Lei, Weihao Liu, Shizhu He, Jun Zhao, and Kang Liu. 2024. Moelora: Contrastive learning guided mixture of experts on parameter-efficient fine-tuning for large language models. *Preprint*, arXiv:2402.12851.

Yulong Mao, Kaiyu Huang, Changhao Guan, Ganglin Bao, Fengran Mo, and Jinan Xu. 2024. DoRA: Enhancing parameter-efficient fine-tuning with dynamic rank distribution. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11662–11675, Bangkok, Thailand. Association for Computational Linguistics.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).

Ryota Miyano and Yuki Arase. 2025. Adaptive LoRA merge with parameter pruning for low-resource generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19353–19366, Vienna, Austria. Association for Computational Linguistics.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, and 1 others. 2011. Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning*, volume 2011, page 7. Granada.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. 2023. Task arithmetic in the tangent space: Improved editing of pre-trained models. *Advances in Neural Information Processing Systems*, 36:66727–66754.

Aniello Panariello, Daniel Marczak, Simone Magistri, Angelo Porrello, Bartłomiej Twardowski, Andrew D Bagdanov, Simone Calderara, and Joost van de Weijer. Accurate and efficient low-rank model merging in core space. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Akshara Prabhakar, Yuanzhi Li, Karthik Narasimhan, Sham Kakade, Eran Malach, and Samy Jelassi. 2025. LoRA soups: Merging LoRAs for practical skill composition tasks. In *Proceedings of the 31st International Conference on Computational Linguistics: Industry Track*, pages 644–655, Abu Dhabi, UAE. Association for Computational Linguistics.

Peijun Qing, Chongyang Gao, Yefan Zhou, Xingjian Diao, Yaoqing Yang, and Soroush Vosoughi. 2024. AlphaLoRA: Assigning LoRA experts based on layer training quality. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 20511–20523, Miami, Florida, USA. Association for Computational Linguistics.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, and 1 others. 2021. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR.

Pengjie Ren, Chengshun Shi, Shiguang Wu, Mengqi Zhang, Zhaochun Ren, Maarten de Rijke, Zhumin Chen, and Jiahuan Pei. 2024. MELoRA: Mini-ensemble low-rank adapters for parameter-efficient fine-tuning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3052–3064, Bangkok, Thailand. Association for Computational Linguistics.

Jingzhe Shi, Qinwei Ma, Hongyi Liu, Hang Zhao, Jenq-Neng Hwang, and Lei Li. 2026. Intrinsic entropy of context length scaling in LLMs. In *The Fourteenth International Conference on Learning Representations*.

Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. 2011. The german traffic sign recognition benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks*, pages 1453–1460.

George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. 2025. Model merging with SVD to tie the knots. In *The Thirteenth International Conference on Learning Representations*.

Dennis Tang, Prateek Yadav, Yi-Lin Sung, Jaehong Yoon, and Mohit Bansal. 2025. LoRA merging with SVD: Understanding interference and preserving performance. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*.

Chunlin Tian, Zhan Shi, Zhijiang Guo, Li Li, and Cheng zhong Xu. 2024. HydraloRA: An asymmetric loRA architecture for efficient fine-tuning. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Mojtaba Valipour, Mehdi Rezagholizadeh, Ivan Kobyzev, and Ali Ghodsi. 2023. DyLoRA:

Parameter-efficient tuning of pre-trained models using dynamic search-free low-rank adaptation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 3274–3287, Dubrovnik, Croatia. Association for Computational Linguistics.

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

Haoyu Wang, Tianci Liu, Ruirui Li, Monica Xiao Cheng, Tuo Zhao, and Jing Gao. 2024. RoseLoRA: Row and column-wise sparse low-rank adaptation of pre-trained language model for knowledge editing and fine-tuning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 996–1008, Miami, Florida, USA. Association for Computational Linguistics.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Hui Wu, Haoquan Zhai, Yuchen Li, Hengyi Cai, Peirong Zhang, Yidan Zhang, Lei Wang, Chunle Wang, Yingyan Hou, Shuaiqiang Wang, and Dawei Yin. 2025. Mara: A multimodal adaptive retrieval-augmented framework for document question answering. In *Proceedings of the 33rd ACM International Conference on Multimedia*, MM '25, page 4329–4338, New York, NY, USA. Association for Computing Machinery.

Jianxiong Xiao, Krista A Ehinger, James Hays, Antonio Torralba, and Aude Oliva. 2016. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision*, 119(1):3–22.

Xi Xiao, Chenrui Ma, Yunbei Zhang, Chen Liu, Zhuxuanzi Wang, Yanshu Li, Lin Zhao, Guosheng Hu, Tianyang Wang, and Hao Xu. 2026a. Not all directions matter: Toward structured and task-aware low-rank adaptation. *Preprint*, arXiv:2603.14228.

Xi Xiao, Yunbei Zhang, Lin Zhao, Yiyang Liu, Xiaoying Liao, Zheda Mai, Xingjian Li, Xiao Wang, Hao Xu, Jihun Hamm, Xue Lin, Min Xu, Qifan Wang, Tianyang Wang, and Cheng Han. 2026b. Prompt-based adaptation in large-scale vision models: A survey. *Preprint*, arXiv:2510.13219.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2023. TIES-merging: Resolving interference when merging models. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Yutong Yin and Zhaoran Wang. 2025. Are transformers able to reason by connecting separated knowledge in training data? In *The Thirteenth International Conference on Learning Representations*.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are super mario: absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Lei Yu, Jingcheng Niu, Zining Zhu, Xi Chen, and Gerald Penn. 2025. Sheaf discovery with joint computation graph pruning and flexible granularity. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 8822–8837, Suzhou, China. Association for Computational Linguistics.

Juzheng Zhang, Jiacheng You, Ashwinee Panda, and Tom Goldstein. 2025a. LoRI: Reducing cross-task interference in multi-task low-rank adaptation. In *Second Conference on Language Modeling*.

Longteng Zhang, Lin Zhang, Shaohuai Shi, Xiaowen Chu, and Bo Li. 2023a. Lora-fa: Memory-efficient low-rank adaptation for large language models fine-tuning. *Preprint*, arXiv:2308.03303.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. 2023b. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*.

Qiyuan Zhang, Fuyuan Lyu, Zexu Sun, Lei Wang, Weixu Zhang, Wenyue Hua, Haolun Wu, Zhihan Guo, Yufei Wang, Niklas Muennighoff, Irwin King, Xue Liu, and Chen Ma. 2025b. A survey on test-time scaling in large language models: What, how, where, and how well? *Preprint*, arXiv:2503.24235.

Tong Zhang, Kuofeng Gao, Jiawang Bai, Leo Yu Zhang, Xin Yin, Zonghui Wang, Shouling Ji, and Wenzhi Chen. 2025c. Pre-training CLIP against data poisoning with optimal transport-based matching and alignment. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9825–9838, Suzhou, China. Association for Computational Linguistics.

Xueyan Zhang, Jinman Zhao, Zhifei Yang, Yibo Zhong, Shuhao Guan, Linbo Cao, and Yining Wang. 2025d. UORA: Uniform orthogonal reinitialization adaptation in parameter efficient fine-tuning of large models. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11709–11728, Vienna, Austria. Association for Computational Linguistics.

Jinman Zhao, Erxue Min, Hui Wu, Ziheng Li, Zexu Sun, Hengyi Cai, Shuaiqiang Wang, Xu Chen, and Gerald Penn. 2026. Beyond step pruning: Information theory based step-level optimization for self-refining large language models. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(41):34941–34949.

Jinman Zhao, Xueyan Zhang, Jiaru Li, Jingcheng Niu, Yulan Hu, Erxue Min, and Gerald Penn. 2025a. Tiny budgets, big gains: Parameter placement strategy in parameter super-efficient fine-tuning. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 6315–6333, Suzhou, China. Association for Computational Linguistics.

Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, and Fei Wu. 2025b. Merging loRAs like playing LEGO: Pushing the modularity of loRA to extremes through rank-wise clustering. In *The Thirteenth International Conference on Learning Representations*.

Yibo Zhong and Yao Zhou. 2024. Pear: Pruning and sharing adapters in visual parameter-efficient fine-tuning. *Preprint*, arXiv:2409.19733.

Xiaoling Zhou, Ou Wu, and Nan Yang. 2024a. Adversarial training with anti-adversaries. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10210–10227.

Xiaoling Zhou, Wei Ye, Zhemg Lee, Rui Xie, and Shikun Zhang. 2024b. Boosting model resilience via implicit adversarial data augmentation. *Preprint*, arXiv:2404.16307.

A Benchmark Statistics

See Table 6.

B Baseline Methods

In this section, we provide detailed descriptions of all baseline methods used in our experiments. All baselines are applied in a gradient-free manner and operate directly on LoRA weight updates $\{\Delta W_j^{(i)}\}$, following the LoRA model merging formulation described in the main text.

Task Arithmetic (TA). Task Arithmetic (TA) (Ilharco et al., 2023a) merges models by linearly combining their task-specific updates. For each layer j, TA computes the merged update as

$$\Delta W_j^{(\text{merged})} = \sum_{i=1}^{n} \alpha_i \Delta W_j^{(i)},$$

where $\alpha_i \geq 0$ is a scaling coefficient associated with the i-th task. In practice, TA typically uses a shared scaling coefficient across all tasks, which is tuned on a held-out validation set. The merged update is then added to the pre-trained weights to obtain the final model.

RegMean. RegMean (Jin et al., 2023) performs model merging by aligning task updates through a closed-form regression objective. For each layer, RegMean estimates a transformation that minimizes the discrepancy between task-specific updates under a locally linear approximation. The transformed updates are then averaged to produce the merged update. Unlike TA, RegMean explicitly accounts for parameter misalignment across tasks during merging.

TIES. TIES (Yadav et al., 2023) extends Task Arithmetic by mitigating parameter interference between task updates. Before merging, TIES prunes parameters with small magnitudes and resolves sign conflicts by retaining only parameters that share a dominant sign across tasks. The remaining parameters are then linearly combined using scaling coefficients, similar to TA. This procedure aims to reduce destructive interference during merging.

DARE-TIES. DARE-TIES (Yu et al., 2024) further improves robustness by introducing stochastic sparsification into the merging process. DARE randomly drops parameters in each task update according to a Bernoulli distribution and rescales the remaining parameters to preserve magnitude. The resulting sparse updates are then merged using the TIES procedure. Due to the stochastic nature of DARE, multiple random seeds are typically evaluated, and the best-performing merged model is selected.

KnOTS. KnOTS (Stoica et al., 2025) is a subspace-alignment-based approach for merging LoRA-finetuned models. For each layer, KnOTS concatenates task updates and applies singular value decomposition (SVD) to obtain a shared representation space. Merging is performed on the aligned task-specific components, and the merged update is reconstructed using the shared basis. This approach enables existing merging methods, such as TA or TIES, to operate in an aligned space.

Core Space. Core Space Merging (Panariello et al.) is a low-rank merging framework that performs model merging in a shared *core space* without reconstructing full weight updates. For each layer, it derives a common alignment basis from the LoRA factors, projects task-specific updates into this reversible low-dimensional space, and then applies standard merging methods within the projected representation. Because the projection is

Dataset	Task	#Train	#Val	#Test
Vision Benchmarks				
Cars	Image Classification	8,144	–	8,041
DTD	Texture Classification	1,880	1,880	1,880
EuroSAT	Image Classification	21,600	2,700	2,700
GTSRB	Traffic Sign Recognition	39,209	2,640	12,630
MNIST	Digit Classification	60,000	–	10,000
RESISC45	Scene Classification	18,900	2,100	6,300
SUN397	Scene Classification	19,850	-	19,850
SVHN	Digit Classification	73,257	–	26,032
Language Benchmarks (NLI)				
SNLI	Natural Language Inference	549,367	9,842	9,824
MNLI	Natural Language Inference	392,702	20,000	20,000
SICK	Natural Language Inference	4,439	495	4,906
QNLI	Natural Language Inference	104,743	5,463	5,463
RTE	Natural Language Inference	2,490	277	3,000
SciTail	Natural Language Inference	23,596	1,304	2,126

Table 6: Dataset statistics for all vision and language benchmarks used in our experiments.

information-preserving and its dimensionality depends only on the LoRA rank and the number of tasks, Core Space retains the efficiency of low-rank adaptation while improving merging accuracy and reducing computational cost.

LoRA-LEGO. LoRA-LEGO (Zhao et al., 2025b) is a compositional LoRA merging method that constructs merged models by selecting and combining LoRA modules across tasks. It formulates LoRA merging as a modular composition problem and learns task-specific combinations without modifying the base model. In our experiments, we adopt the gradient-free variant of LoRA-LEGO and apply it following the standard protocol described in the original work.

C Training Detail

We used a CLIP based ViT-B/32 model from Hugging Face. The original model is finetuned with LORA using rank 16 and alpha 16. LoRA is used in query,key,value and output projection layer. Dropout is set to be 0.1. For our method, we first finetuned coefficient and top-k for KNOTS-TIES and CUR with a linear seach by finding the best co-efficient on a default top-K to be 20, following pruning top-k with the best co-efficient. The random dare seed was set to 421 and big seed was set to 420.

All of our experiments were conducted on 1 NVIDIA H200 GPUGPU with 150 GB of memory. Training took approximately 8 hours on VIT model.

D CUR Interpolation Detail

See Table 7 for detailed numbers.

E Joint Task Performance

See Table 9.

F Combine with LoRA-LEGO

See Table 8.

G Use of Large Language Models

Large language models were used only for grammatical editing and language polishing. They were not used for research design, analysis, experiments, or interpretation.

α	Cars	DTD	EuroSAT	GTSRB	MNIST	RESISC45	SUN397	SVHN	Avg
0.0	81.0	73.5	52.7	27.1	83.6	62.3	97.4	62.7	67.5
0.1	82.1	74.0	53.8	28.1	83.9	63.6	97.6	62.4	68.2
0.2	82.8	74.1	54.5	29.6	83.7	64.6	97.4	62.1	68.6
0.3	83.5	74.3	54.9	31.4	83.6	65.9	97.5	61.7	69.1
0.4	84.1	74.2	55.0	33.6	83.1	66.9	97.7	61.2	69.5
0.5	84.3	74.2	54.7	36.5	82.1	68.0	97.5	60.6	69.7
0.6	84.5	73.6	54.7	39.7	80.7	69.0	97.5	60.0	70.0
0.7	84.8	73.6	54.4	42.8	78.3	69.6	97.3	58.8	70.0
0.8	84.7	73.4	53.8	45.5	75.8	70.0	96.9	57.4	69.7
0.9	84.4	72.7	52.4	47.6	72.6	69.3	96.5	55.6	68.9
1.0	83.8	73.7	48.9	50.0	69.0	67.8	96.1	53.8	67.7

Table 7: Effect of SVD mixing ratio α on per-task performance (normalized against finetuned models).

α	Cars	DTD	EuroSAT	GTSRB	MNIST	RESISC45	SUN397	SVHN	Avg
0.0	82.4	72.5	42.4	41.8	51.0	68.4	96.9	40.0	62.0
0.1	81.2	75.4	46.2	36.6	50.6	67.4	98.3	34.6	61.3
0.2	81.7	76.1	47.7	37.9	52.7	68.2	98.5	36.7	62.4
0.3	82.7	76.0	49.1	39.5	55.1	68.6	98.4	38.7	63.5
0.4	83.5	76.0	49.8	41.4	57.5	69.1	98.3	41.0	64.6
0.5	84.1	76.0	50.7	43.1	59.9	69.2	98.4	43.2	65.6
0.6	84.4	74.8	51.2	44.6	62.2	69.3	98.1	45.5	66.3
0.7	84.6	74.5	51.4	46.0	64.3	69.4	97.6	47.7	66.9
0.8	84.5	74.6	50.9	47.3	66.3	69.1	96.9	50.0	67.5
0.9	84.3	73.8	49.9	48.4	67.6	68.6	96.5	52.1	67.7
1.0	83.8	73.7	48.8	49.2	69.0	67.8	96.0	53.7	67.7

Table 8: Effect of KNOTS mixing ratio α on per-task performance with LORA-lego model(normalized against finetuned models).

Method	Metric	Cars	DTD	EuroSAT	GTSRB	MNIST	RESISC45	SUN397	SVHN	Union
	Hits@1	58.8	39.4	**15.5**	30.7	34.0	52.8	62.1	22.7	39.7
Ensemble	Hits@3	83.4	60.0	**30.0**	55.3	55.1	77.1	84.1	40.5	60.9
	Hits@5	90.9	71.8	40.0	67.5	66.4	85.5	90.1	50.8	70.2
	Hits@1	61.1	**40.8**	14.6	37.8	31.6	59.4	**62.4**	27.6	42.7
TA	Hits@3	84.8	63.0	25.6	64.4	47.5	83.2	**84.5**	47.7	64.0
	Hits@5	91.9	73.1	36.3	76.0	55.6	89.7	**90.3**	58.0	72.7
	Hits@1	60.7	39.6	12.1	35.5	33.4	58.7	61.6	32.9	43.7
TIES	Hits@3	85.0	61.4	20.6	63.5	48.2	82.7	83.8	53.6	65.4
	Hits@5	92.1	72.3	28.1	75.5	54.0	89.2	89.9	64.2	73.9
	Hits@1	61.2	39.5	11.6	33.9	34.0	56.7	60.3	35.0	43.8
DARE-TIES	Hits@3	85.2	61.0	16.1	64.0	49.0	82.3	82.5	57.0	66.0
	Hits@5	92.2	73.2	18.2	74.5	54.3	89.0	88.9	67.8	74.2
	Hits@1	61.9	**40.8**	15.1	**45.5**	**39.2**	58.7	60.3	37.1	47.1
KnOTS-TIES	Hits@3	85.7	**64.0**	20.4	**69.8**	52.4	83.6	82.4	57.9	67.9
	Hits@5	92.5	**74.6**	27.4	**80.3**	58.0	90.3	88.7	**68.2**	76.1
	Hits@1	61.4	40.3	14.2	42.3	36.5	58.6	60.5	34.7	45.5
KnOTS-DARE-TIES	Hits@3	85.1	63.4	19.3	67.8	51.5	**83.9**	82.6	55.9	66.8
	Hits@5	92.2	74.5	25.0	78.7	57.5	**90.4**	89.0	66.9	75.3
	Hits@1	61.0	40.1	13.2	39.5	29.6	59.3	61.6	29.0	42.9
LoRA-LEGO	Hits@3	84.8	62.7	20.4	66.4	46.6	83.4	83.7	49.8	64.5
	Hits@5	91.8	72.8	26.8	77.8	55.0	89.5	89.9	61.4	73.5
	Hits@1	**62.8**	40.2	12.5	38.3	**45.0**	**59.4**	61.7	**40.0**	**47.9**
Ours	Hits@3	**86.2**	62.0	27.2	64.5	**55.9**	83.6	84.0	**59.0**	**68.6**
	Hits@5	**93.1**	73.1	**41.3**	75.6	**60.0**	90.2	89.8	67.3	**76.2**

Table 9: Joint-Task Performances (%). Best results for each dataset and metric are highlighted in bold.