PRISM: Spectral Pruning and Reconstruction for Parameter-Efficient Model Merging of MLLMs

Wuxuan Shi[1], **Haotian Chen**[1], **He Li**[1], **Qiang Yang**[3], **Mang Ye**[1,2*]
[1]School of Computer Science, Wuhan University, Wuhan, China
[2] Taikang Center for Life and Medical Sciences, Wuhan University, Wuhan, China
[3] PolyU Academy for Artificial Intelligence,
Hong Kong Polytechnic University, Hong Kong, China
{wuxuanshi, yemang}@whu.edu.cn
https://github.com/ShiWuxuan/PRISM

Abstract

Parameter-Efficient Fine-Tuning (PEFT) has become a dominant paradigm for adapting Multimodal Large Language Models (MLLMs) to specific tasks, resulting in a proliferation of task-specific expert models. Merging these experts into one universal model offers a promising path to creating versatile systems without costly retraining. However, existing methods for full fine-tuning merging often falter in parameter-efficient model merging, as they manipulate weights directly while ignoring the intrinsic geometric structure of PEFT modules. When diverse tasks exhibit conflicting feature directions, this inevitably leads to destructive interference. To address this issue, we propose PRISM, a data-free framework that reframes the merging of PEFT modules as a spectral signal reconstruction problem. PRISM operates in two stages: (1) Spectral Pruning, which decouples task updates into singular components and retains only high-energy directions to attenuate task-irrelevant noise; and (2) Energy-Prioritized Orthogonal Reconstruction, which prioritizes dominant components and projects overlapping vectors onto orthogonal directions to eliminate inter-task interference. We curate a benchmark comprising diverse multimodal tasks to evaluate our method. Extensive experiments demonstrate that PRISM significantly outperforms state-of-the-art baselines.

1 Introduction

The rapid advancement of Multimodal Large Language Models (MLLMs) has established them as a robust foundation for diverse applications. To adapt these general-purpose models to specific domains, fine-tuning on downstream task data is typically required. As model sizes scale, the cost of updating all parameters becomes prohibitively expensive. Consequently, parameter-efficient fine-tuning (PEFT) [13] has emerged as a popular paradigm that balances performance with computational efficiency. Nevertheless, building isolated experts for different domains leads to the proliferation of models. This raises a critical question: how can we integrate these specialized capabilities into a single, versatile system? Model merging is a promising avenue to realize this vision [40, 53, 60]. It aims to integrate multiple separate PEFT experts into one model with multi-task ability, effectively circumventing the need for retraining or access to raw private data [37, 52].

Despite the intuitive appeal of model merging, existing approaches are primarily designed for full fine-tuning (FFT) techniques [15, 55]. They encounter significant limitations when applied to the low-rank updates of PEFT, as illustrated in Figure 1 (b). The fundamental flaw is that they treat weight matrices as mere collections of numbers, manipulating them directly while neglecting the intrinsic

*Corresponding Author: Mang Ye

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

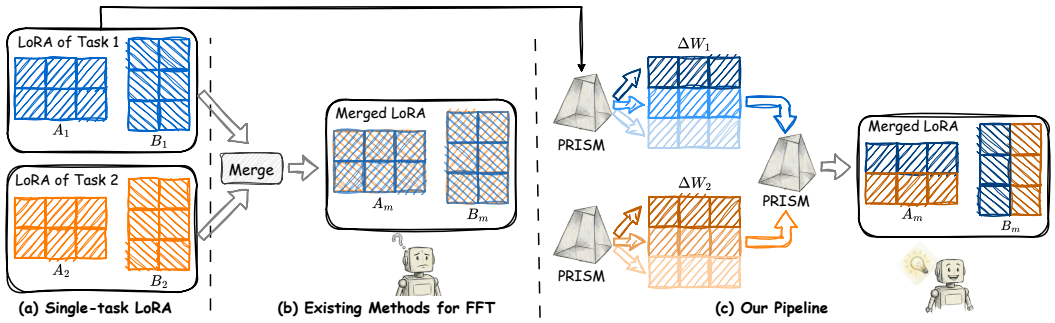

Figure 1: Comparison of merging paradigms. (a) Independent LoRA modules tailored for diverse tasks. (b) FFT-based methods often lead to degradation as they neglect the geometric structure of updates. (c) Our Pipeline: PRISM extracts dominant spectral components and realigns them in an orthogonal space, effectively mitigating inter-task conflicts while preserving essential expertise.

structural patterns of the learned parameters. Specifically, naive aggregation fails to distinguish between critical task features and insignificant noise, treating all updates with equal importance. Furthermore, these methods overlook the spatial conflict between different experts. When distinct tasks update the model in incompatible directions, direct summation leads to destructive interference, compromising the integrity of the merged model [5]. This issue becomes more apparent when integrating experts from heterogeneous domains.

To address these challenges, we propose PRISM, a novel framework that reframes the merging of PEFT modules as a spectral signal reconstruction problem. As shown in Figure 1 (c), PRISM operates in two stages to extract and re-adjust spectral components from multiple task-specific updates. First, we design Spectral Pruning, which decomposes parameter updates into distinct components and retains only the most significant “high-energy" signals. This ensures that the merged model prioritizes task-essential knowledge while filtering out the irrelevant noise that accumulates during independent fine-tuning. Second, we propose Energy-Prioritized Orthogonal Reconstruction. By projecting overlapping features onto a set of orthogonal bases according to their importance, this mechanism mitigates the spatial conflicts between diverse experts. This structured approach enables PRISM to integrate diverse expertise into a unified model, achieving a more robust balance of specific knowledge across different domains.

To rigorously evaluate the efficacy of PRISM, we designed a challenging evaluation setup for parameter-efficient merging in MLLM using the comprehensive datasets constructed by [62]. It integrates datasets from specialized fields such as remote sensing, medical imaging, autonomous driving, and finance, along with various capability-oriented tasks including GUI agent interaction and mathematical reasoning. Extensive experiments demonstrate that PRISM significantly outperforms existing state-of-the-art (SOTA) merging baselines.

The main contributions of this work are summarized:

- We propose a spectral-based perspective for PEFT model merging, which analyzes the importance of parameter updates through singular value decomposition to identify task-essential features.
- We develop the PRISM framework, which introduces Spectral Pruning to filter out low-energy noise and Energy-Prioritized Orthogonal Reconstruction to resolve conflicts between task vectors.
- We establish a robust evaluation protocol across a diverse cross-domain benchmark, where PRISM demonstrates its superior capacity for parameter-efficient merging of MLLMs.

2 Related Work

Multimodal large language models. The remarkable success of Large Language Models (LLMs) is largely attributed to the exponential growth in training data and model parameters [35, 2, 45]. Primarily built upon auto-regressive and decoder-only architectures, these models have demonstrated

exceptional capabilities in both generation [8] and classification [46] tasks. To bridge the gap between textual cognition and visual modalities, MLLMs further extend this paradigm by incorporating visual perception through aligned vision encoders [23, 3]. With the rapid expansion of application scenarios, MLLMs have been increasingly adapted to specialized domains. However, training separate domain-specific MLLMs leads to fragmented expertise and substantial storage overhead. These challenges have spurred exploration of efficient knowledge integration strategies, particularly those that avoid full retraining while preserving multi-task capabilities.

Parameter-efficient tuning. Adapting pre-trained models to downstream tasks via full-parameter fine-tuning is often computationally prohibitive and prone to catastrophic forgetting [33, 22], eroding the robust representations acquired from massive pre-training corpora. To mitigate these challenges, PEFT [10] has been proposed as a resource-conscious alternative. By freezing the pre-trained backbone and optimizing only a minimal set of lightweight auxiliary modules, PEFT strategies achieve performance comparable to full fine-tuning while significantly reducing memory overhead. The landscape of efficient tuning encompasses diverse methodologies, ranging from prompt learning [17, 21] to adapter-based variants such as $(\text{IA})^3$ [25] and LoRA [13, 28, 67]. In this work, we specifically adopt Low-Rank Adaptation (LoRA). Beyond its status as the de facto standard for large model adaptation, LoRA's inherent decomposition into low-rank matrices provides the explicit geometric structure necessary for the spectral analysis and signal reconstruction central to our proposed framework.

Model merging. Model merging [54, 51, 49] is defined as the consolidation of distinct expert models into a single universal system to facilitate multi-task competence. Building upon the foundational Task Arithmetic paradigm [16] which treats task updates as vectors, subsequent works [32, 48, 7] introduce mechanisms such as sparsification and isotropic scaling to mitigate parameter interference in FFT models. Recently, this line of research has been extended to PEFT scenarios [42, 34, 66]. LoRAHub [14] investigate the composition of low-rank adapters, typically formulating merging as a weight routing or selection problem within the parameter space. More closely related to our work, RobustMerge [58] prunes parameters with small magnitudes that contribute to direction instability to preserve the key directional information for each task during merging. DOMerging [65] learns additive perturbations to LoRA factors through iterative, layer-wise optimization with pairwise orthogonality constraints, followed by magnitude-direction decoupling and merging. However, most existing methods directly manipulate weight updates and overlook the intrinsic low-rank geometry of LoRA updates. In contrast, PRISM explicitly performs spectral pruning and reconstruction under the prescribed merged LoRA rank budget, **requiring neither external data nor additional manually designed hyperparameters.**

3 Methodology

3.1 Preliminary

Parameter-efficient tuning. In this work, we employ Low-Rank Adaptation (LoRA) [13] as the primary fine-tuning technique. LoRA optimizes efficiency by decomposing the weight update into two low-rank matrices. Given a pre-trained weight matrix $\mathbf{W}_0 \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$ within a MLLM, the incremental update $\Delta\mathbf{W}$ is reparameterized as the product of two low-rank matrices $\mathbf{B} \in \mathbb{R}^{d_{\text{out}} \times r}$ and $\mathbf{A} \in \mathbb{R}^{r \times d_{\text{in}}}$, where $r \ll \min(d_{\text{in}}, d_{\text{out}})$. For a given task i, the updated weight is expressed as:

$$\mathbf{W}_i = \mathbf{W}_0 + \Delta\mathbf{W}_i = \mathbf{W}_0 + \mathbf{B}_i \cdot \mathbf{A}_i. \tag{1}$$

We treat the weight increment $\Delta\mathbf{W}_i$ as the carrier of task-specific expertise.

Parameter-efficient model merging. Consider a set of N task-specific experts $\{(\mathbf{A}_i, \mathbf{B}_i)\}_{i=1}^{N}$ fine-tuned from a common frozen backbone. The goal of parameter-efficient merging is to derive a unified adapter $(\mathbf{A}_m, \mathbf{B}_m)$ that effectively consolidates the collective knowledge of all task-specific updates. Formally, the unified model constructed can be defined as:

$$\mathbf{W}_m = \mathbf{W}_0 + \mathbf{B}_m \cdot \mathbf{A}_m = \mathbf{W}_0 + \lambda \sum_{i=1}^{N} \Phi(\Delta\mathbf{W}_i), \tag{2}$$

where $\Phi(\cdot)$ stands for the merging algorithm and λ is the global scaling coefficient applied to the aggregated task updates before they are added to the pre-trained weights. The merging process aims to maintain robust performance across all N tasks while minimizing inter-task interference.

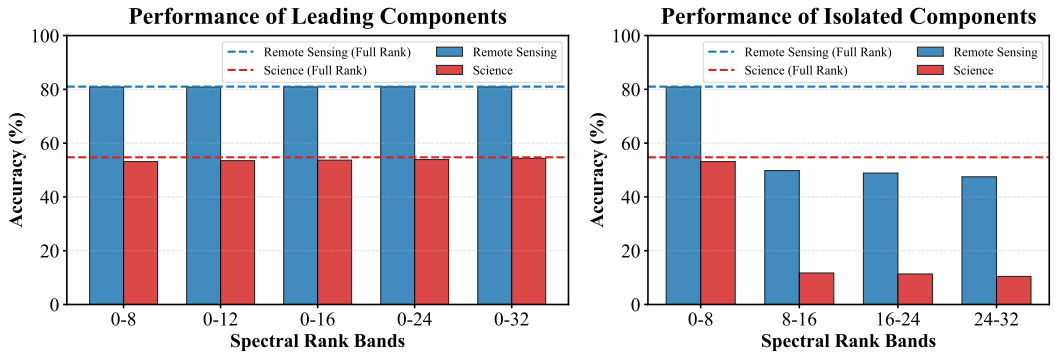

Figure 2: Performance of LoRA adapters reconstructed from different spectral rank bands. Leading spectral components rapidly recover full-task performance, whereas isolated mid- and low-energy components fail to support tasks independently, indicating strong spectral concentration and redundancy in LoRA. Dashed lines indicate full-rank LoRA performance.

3.2 Motivation and Observation

LoRA has become a standard technique for PEFT. While the rank of a LoRA module explicitly limits the dimensionality of the update space, it does not prescribe how task-relevant information is distributed within that space. In particular, it remains unclear whether all rank components contribute equally to downstream task performance, or whether certain directions play a more dominant role. This ambiguity is especially critical in the context of parameter-efficient model merging. Existing merging strategies typically aggregate LoRA updates by treating all low-rank components uniformly, assuming that each rank direction carries useful and compatible information. This uniform aggregation may introduce redundant or even conflicting directions, thereby degrading the performance of the merged model. These considerations motivate a closer examination of the internal structure of LoRA updates from a spectral perspective.

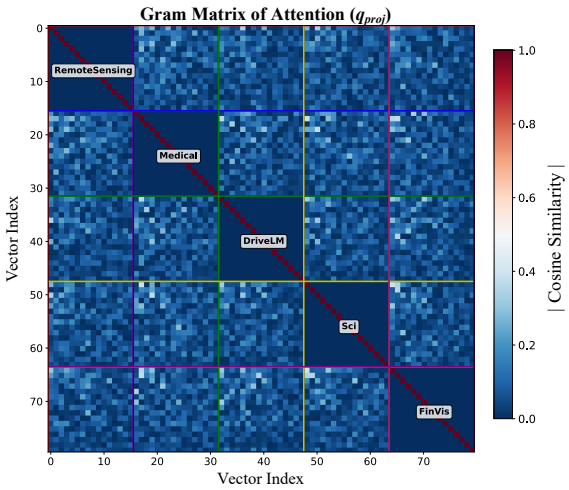

Figure 3: Cross-task spectral similarity of LoRA attention updates. Gram matrices of the top-k right singular vectors of LoRA-induced q_{proj} weights, computed across five tasks. Color intensity denotes the absolute cosine similarity: reddish values close to 1 correspond to highly aligned or misaligned directions, while bluish values close to 0 indicate near-orthogonal alignment. Diagonal blocks indicate intra-task similarity, while off-diagonal blocks indicate cross-task similarity.

To better understand how task information is organized in low-rank spaces and whether different tasks exhibit variations in this regard, we conduct an analysis that progresses from single-task behavior to cross-task interactions. Specifically, we (1) first analyze the role of different spectral rank bands in LoRA updates on downstream task performance; (2) then reveal structural interference between different task-specific LoRA modules; and (3) propose a parameter-efficient model merging algorithm with an adjustable rank budget based on these observations.

Task performance saturates rapidly with respect to spectral rank. As shown in Figure 2, reconstructing LoRA adapters using only a small number of leading singular components recovers nearly the full performance of the original model. This indicates that LoRA updates exhibit strong spectral concentration and substantial redundancy, where most task-relevant information is captured by a few dominant directions.

Marginal and non-independent role of low-energy components in LoRA. In contrast, adapters reconstructed from mid- or low-energy spectral components alone suffer a severe performance degradation across tasks. The results in Table 1 collectively indicate that these components lack sufficient expressive power. They primarily act as minor adjustments when combined with high-energy spectral components, rather than independent carriers of task knowledge. This suggests that uniformly retaining all rank directions is inefficient, especially when multiple task updates are merged under a shared rank constraint.

LoRA updates across tasks exhibit structured interference and are dominated by the most prominent singular vectors. Figure 3 provides a qualitative view of cross-task interference through similarity heatmaps computed along top-ranked spectral directions. A notable observation is that cross-task directions exhibit noticeably higher collinearity than task-internal ones. Within the task, the trained LoRA module inherently possesses orthogonality, but subspace overlap occurs when merged across tasks. This suggests that naive merging or rank concatenation may suffer from redundancy under a limited rank budget. More importantly, such cross-task interactions are not uniformly distributed across all components. Within each cross-task block, higher similarities concentrate near the top-left region, which correspond to the most dominant components ranked by singular value. This concentration indicates that cross-task interference or potential sharing primarily arises from the high-energy components. Consistent patterns are observed across both shallow and deep layers. More visualizations can be found in the Appendix D.

3.3 PRISM Framework

Based on the observations in Section 3.2, parameter-efficient model merging requires addressing two key challenges. First, task-specific LoRA updates exhibit strong spectral concentration, rendering uniform rank aggregation inefficient under a limited budget. Second, the most informative directions across different tasks are often not mutually orthogonal, leading to structured inter-task interference when merged naively. To address these issues, we propose PRISM, a two-stage framework operating within the spectral domain of LoRA-induced weight updates. As illustrated in Fig. 4, our approach consists of Spectral Pruning and Energy-Prioritized Orthogonal Reconstruction.

Spectral Pruning. We aim to identify the most informative components among task-specific updates while discarding redundant or low-impact directions under a shared rank budget. Rather than treating all LoRA rank dimensions as equally important, PRISM explicitly leverages the spectral structure of parameter updates to guide this selection. Given a set of task-specific LoRA adapters $\{(\mathbf{A}_i, \mathbf{B}_i)\}_{i=1}^{N}$, we first recover the corresponding weight increments $\Delta\mathbf{W}_i = \mathbf{B}_i \cdot \mathbf{A}_i$. For each task i, we perform singular value decomposition (SVD):

$$\Delta\mathbf{W}_i = U_i \Sigma_i V_i^\top, \tag{3}$$

where $U_i \in \mathbb{R}^{d_{\text{out}} \times r}$ and $V_i \in \mathbb{R}^{d_{\text{in}} \times r}$ contain the left and right singular vectors, respectively, and $\Sigma_i = \text{diag}(\sigma_{i,1}, \sigma_{i,2}, \ldots, \sigma_{i,r})$ contains the singular values sorted in descending order. Each rank-one component $(\sigma_{i,j},\ u_{i,j},\ v_{i,j})$ can be interpreted as an atomic spectral component in the update space, with its singular value serving as a natural measure of contribution strength.

To select which components should be preserved in the merged model, PRISM constructs a global spectral pool that aggregates all singular components across tasks:

$$\mathcal{P} = \{(\sigma_{i,j}, u_{i,j}, v_{i,j}) \mid i = 1, \ldots, N;\ j = 1, \ldots, r\}. \tag{4}$$

We then perform global energy-based selection by retaining the top-r_m components with the largest singular values, where r_m denotes the prescribed rank budget of the merged LoRA adapter:

$$\mathcal{P}_{\text{top}} = \text{TopK}_\sigma(\mathcal{P}, K = r_m) = \{(\sigma_k, u_k, v_k)\}_{k=1}^{K}, \tag{5}$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_K$. Here, r_m is determined by the desired parameter budget of the merged adapter rather than introduced as an additional tunable hyperparameter. When $r_m = r$, the merged adapter preserves the same rank as each single-task LoRA expert. Smaller or larger values of r_m naturally correspond to parameter compression or expansion, respectively.

This global pruning strategy is motivated by the empirical finding that task-relevant information in LoRA is highly concentrated in a limited subset of high-energy spectral directions, while lower-energy components play only a marginal or auxiliary role. By retaining only the most significant spectral

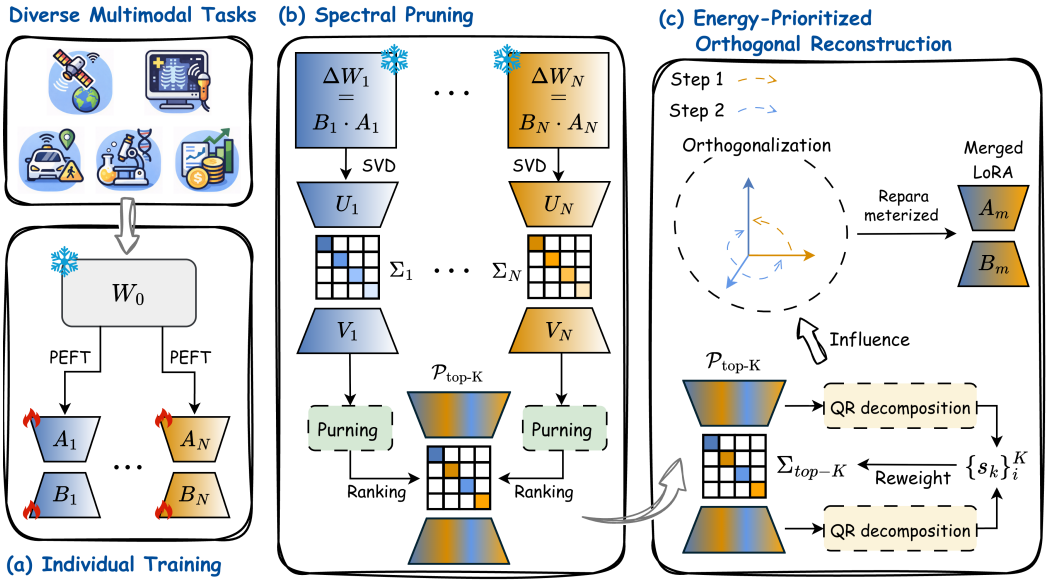

Figure 4: Overview of the PRISM framework. (a) Task-specific LoRA adapters are independently trained from a frozen backbone. (b) Spectral Pruning decomposes each LoRA adapter and globally selects the high-energy components across tasks. (c) Energy-Prioritized Orthogonal Reconstruction orthogonalizes the retained components in descending energy order via QR decomposition, suppressing cross-task interference and redundancy. The orthogonalized components are finally reparameterized into a single merged LoRA adapter.

components, Spectral Pruning effectively filters out task-irrelevant noise and redundancy before cross-task interaction.

Energy-Prioritized Orthogonal Reconstruction. While Spectral Pruning identifies the most informative components under a shared rank budget, the retained components may still exhibit significant geometric overlap across tasks. As observed in Section 3.2, high-energy directions from different tasks exhibit stronger collinearity. Therefore, directly concatenating or summing the pruned components can still lead to destructive interference. To address this issue, PRISM further enforces structured conflict resolution through Energy-Prioritized Orthogonal Reconstruction, which explicitly accounts for both the importance and the mutual compatibility of spectral components.

For the pruned spectral pool $\mathcal{P}_{\text{top}}$, we first construct concatenated subspaces for the left and right singular vectors:

$$\begin{aligned} U_{\text{cat}} &= [u_1, u_2, \ldots, u_K] \in \mathbb{R}^{d_{\text{out}} \times K}, \\ V_{\text{cat}} &= [v_1, v_2, \ldots, v_K] \in \mathbb{R}^{d_{\text{in}} \times K}. \end{aligned} \tag{6}$$

The ordering induced by spectral energy plays a crucial role here, as it defines the priority with which components are incorporated into the shared subspace.

To mitigate cross-task interference, we perform ordered orthogonalization on both concatenated matrices via QR decomposition:

$$U_{\text{cat}} = Q_U R_U, \quad V_{\text{cat}} = Q_V R_V, \tag{7}$$

where $Q_U \in \mathbb{R}^{d_{\text{out}} \times K}$ and $Q_V \in \mathbb{R}^{d_{\text{in}} \times K}$ form orthonormal bases, and $R_U, R_V \in \mathbb{R}^{K \times K}$ are upper triangular matrices. This procedure is equivalent to a Gram–Schmidt process applied in the order of decreasing spectral energy: the k-th component is projected onto the orthogonal complement of the subspace spanned by the preceding $k-1$ higher-priority components.

Importantly, the diagonal entries of the triangular factors, $(R_U)_{kk}$ and $(R_V)_{kk}$, quantify how much of the k-th spectral component remains after removing its overlap with previously selected directions. If a component is largely redundant with the existing subspace, these values approach zero, indicating

limited additional contribution. We leverage this property to perform energy-prioritized modulation, defining an effective scaling factor for each component as:

$$s_k = |(R_U)_{kk}| \cdot |(R_V)_{kk}| \,. \tag{8}$$

The coefficient s_k measures the fraction of the component's Frobenius norm that survives projection onto the orthogonal complements of both preceding subspaces. Components that introduce a new direction on only one side can therefore be suppressed. This is a conservative choice of the paired reconstruction, and does not imply that such one-sided couplings are mathematically redundant. Appendix

Finally, the orthogonal bases (Q_U, Q_V) together with the modulated singular values $\tilde{\Sigma}$ are reparameterized back into a standard LoRA form. Since these quantities jointly define a rank-K linear operator in the original weight space, the merged weight increment can be expressed as:

$$\Delta W_{\text{merge}} = Q_U \Sigma_{\text{new}} Q_V^\top. \tag{9}$$

To maintain compatibility with the standard LoRA parameterization, we re-factorize the merged update into low-rank form by symmetrically distributing the spectral energy:

$$B_{\text{merge}} = Q_U \Sigma_{\text{new}}^{1/2}, \quad A_{\text{merge}} = \Sigma_{\text{new}}^{1/2} Q_V^\top. \tag{10}$$

This reconstruction introduces no additional learnable parameters and requires no access to training data, yielding a LoRA adapter with comprehensive knowledge under a fixed rank budget.

4 Experiments

4.1 Experimental Setup

Implementation details. We conduct experiments using LLaVA-v1.5-7B [26] and Qwen2.5-VL-3B [3] as foundation models. For all tasks and models, we employ LoRA to fine-tune all linear layers within the large language model and directly perform full fine-tuning on the vision-language projectors. The visual encoder is frozen. Unless otherwise specified, all comparison methods are merged based on the same task-specific LoRA experts and employ identical inference configurations to ensure fair and controlled comparisons. All methods process vision-language projectors using the same protocol: directly averaging task-specific parameters. More details about training and evaluation can be found in Appendix B.

Datasets and baselines. For multimodal task merging, we adopt the benchmark established in [62], encompassing both multi-domain learning and multi-ability learning scenarios. Multi-domain learning aims to construct a comprehensive model across different domains including remote sensing (RS), medical (Med), autonomous driving (AD), science (Sci) and finance (Fin), where the training and test sets are IID. Multi-ability learning focuses on integrating foundational capabilities including OCR, mathematics and logic (Math), visual perception (VP), and GUI agents, which are evaluated on non-IID test sets. For comparison methods, we re-implement SOTA FFT model merging methods (Ties-Merging [50], DARE [57], PCB-Merging [6]) and LoRA-based model merging methods (LoRALEGO [63], KnOTS [41], RobustMerge [58], DC-Merge [59]) on the parameter-efficient modules of MLLMs to ensure a fair comparison. Details about these baselines can be found in Appendix C.

4.2 Main Results

Superior performance across diverse tasks. As shown in Table 1, under the same merged rank budget, PRISM consistently outperforms the state-of-the-art baseline in both experimental settings. In the multi-domain learning, PRISM achieves an average accuracy of 57.51%, surpassing the strongest baseline by a significant margin of 4.8%. This advantage is particularly evident in distinct domains with high heterogeneity, such as remote sensing and medical. For instance, while other methods suffer from severe performance collapse in RS, PRISM effectively recovers the expert capabilities, closely approaching the upper bound of individual fine-tuning. Meanwhile, in the multi-ability learning, **non-IID test** conditions pose a more rigorous challenge to merging algorithms. Under this stricter setting, PRISM still maintains a leading average performance. These results confirm that

Table 1: Quantitative comparison with SOTA merging methods on LLaVA-1.5-7B. The best results are highlighted in **bold**. 'Individual' reports the performance of task-specific LoRA experts evaluated on their corresponding tasks. r represents the rank of individual LoRAs, and r_m denotes the rank budget of the merged LoRA. PRISM achieves superior performance in both settings.

Method	r	r_m	Multi-Domain Tasks						Multi-Ability Tasks				
			RS	Med	AD	Sci	Fin	**Average**	OCR	Math	VP	GUI	**Average**
Individual	128	-	81.01	64.88	53.26	54.63	90.18	68.79	32.03	32.70	62.36	33.83	40.23
Zero-Shot	-	-	32.29	28.28	15.59	35.55	62.56	34.85	31.10	25.60	60.91	25.00	35.65
Weight Average	128	128	42.99	37.48	16.88	3.33	15.48	23.23	31.00	25.80	19.16	10.18	21.54
PCB-Merging	128	128	46.31	40.36	18.15	11.68	51.60	33.62	32.20	26.40	56.48	23.27	34.59
Ties-Merging	128	128	49.21	40.51	20.85	28.17	76.18	42.98	33.20	29.30	57.12	26.67	36.57
DARE	128	128	47.53	37.48	17.90	11.36	67.64	36.38	31.50	26.90	58.17	24.44	35.25
LoRALEGO	128	128	47.07	38.02	17.61	30.03	66.19	39.78	**34.40**	28.79	57.03	27.08	36.83
RobustMerge	128	128	59.75	41.66	26.48	43.84	83.28	51.00	33.10	30.50	57.23	31.39	38.06
KnOTS	128	128	61.79	44.83	25.47	**45.89**	82.06	52.01	34.10	29.80	58.23	30.24	38.09
DC-Merge	128	128	70.90	38.59	27.98	43.30	82.66	52.69	33.10	29.50	56.58	30.16	37.34
PRISM	128	128	**76.12**	**46.94**	**35.66**	42.79	**86.02**	**57.51**	33.60	**31.24**	57.87	**33.89**	**39.15**
PRISM	128	32	74.34	44.55	32.00	42.07	84.31	55.45	33.90	31.20	**58.48**	28.25	37.96

Table 2: Quantitative comparison with SOTA merging methods on Qwen2.5-VL-3B. The best results are highlighted in **bold**. 'Individual' reports the performance of task-specific LoRA experts evaluated on their corresponding tasks. r represents the rank of individual LoRAs, and r_m denotes the rank budget of the merged LoRA. PRISM achieves superior performance even under a restricted r_m.

Method	r	r_m	Multi-Domain Tasks					
			RS	Med	AD	Sci	Fin	**Average**
Individual	16	-	79.44	55.26	51.69	80.89	90.91	71.64
Zero-Shot	-	-	53.36	30.75	5.12	76.72	54.86	44.16
Average Merge	16	16	57.72	32.53	4.85	78.29	64.90	47.66
TIES-Merging	16	16	60.12	35.29	9.69	78.44	69.39	50.59
PCB-Merging	16	16	57.87	34.76	5.74	78.43	64.67	48.29
DARE	16	16	67.70	37.97	5.23	**79.17**	79.94	54.00
LoRALEGO	16	16	58.08	33.12	6.50	78.54	65.68	48.38
RobustMerge	16	16	62.19	36.46	8.46	79.12	73.43	51.93
KnOTS	16	16	68.71	43.20	19.64	78.87	80.14	58.11
DC-Merge	16	16	62.10	37.04	14.56	78.84	69.83	52.47
PRISM	16	16	**71.24**	**48.13**	**48.75**	79.08	**83.32**	**66.10**
PRISM	16	8	70.68	47.08	47.78	78.73	81.41	65.14

resolving spatial conflicts in the spectral domain is more effective than direct weight manipulation for integrating diverse expert knowledge.

Robustness and efficiency in integration. Beyond achieving high average scores, the robustness and efficiency of PRISM are also noteworthy. While baselines like KnOTS or LoRALEGO may marginally outperform our method in specific isolated tasks (e.g., Science or OCR), they often suffer from severe degradation in others. PRISM mitigates this phenomenon by orthogonally reconstructing conflicting components, ensuring a balanced performance without catastrophic failures in any domain. Furthermore, PRISM naturally supports different prescribed rank budgets for the merged LoRA adapter, demonstrating remarkable parameter efficiency under compressed settings. As shown in the last row of Table 1, even when the merged rank is compressed to $r_m = 32$ (only 25% of the standard budget), PRISM still attains 55.45% on multi-domain tasks, outperforming most full-rank baselines. This provides strong support for the motivation behind our Spectral Pruning: task-essential information is highly concentrated in dominant spectral components.

Generalization across model architectures. To evaluate the performance of PRISM on different architectural foundation models, we extended our evaluation to Qwen2.5-VL-3B, a state-of-the-art

Table 3: Ablation studies for each component of PRISM in both multi-domain and multi-ability learning scenarios. Red numbers indicate the improvement in average accuracy over the baseline.

SP	EPOR	Multi-Domain Tasks						Multi-Ability Tasks				
		RS	Med	AD	Sci	Fin	**Average**	OCR	Math	VP	GUI	**Average**
		51.59	35.40	21.64	37.49	71.88	43.60	31.60	28.28	55.44	31.54	36.72
✓		74.62	42.11	32.38	39.24	85.09	54.69 (+11.09)	**33.90**	29.99	56.92	33.01	38.46 (+1.74)
	✓	51.15	36.27	22.93	38.41	72.06	44.18 (+0.56)	32.07	28.65	55.76	31.93	37.10 (+0.38)
✓	✓	**76.12**	**46.94**	**35.66**	**42.79**	**86.02**	**57.51** (+13.91)	33.60	**31.20**	**57.87**	**33.89**	**39.14** (+2.42)

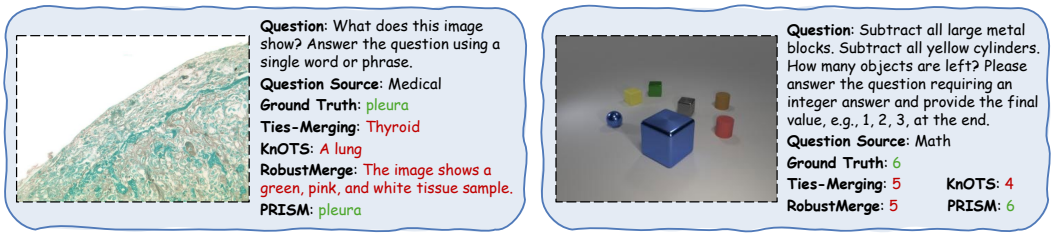

Figure 5: Visualization of PRISM and other baselines in multi-domain and multi-skill learning scenarios. The left panel shows medical tasks, and the right panel shows math tasks.

MLLM with a distinct visual-linguistic alignment mechanism. As detailed in Table 2, the superiority of PRISM is even more pronounced. PRISM achieves the highest average accuracy, surpassing the strongest baseline by a significant margin of 7.99%. In particular, all baselines experienced significant performance degradation in the autonomous driving (AD) domain. The performance of some methods drops to near zero-shot levels, indicating that the knowledge about AD within the merged LoRA was severely disrupted. Meanwhile, PRISM maintained robust performance at 48.75%, approaching that of the individual LoRA. This result highlights the advantage of our method in maintaining a multi-task equilibrium, preventing knowledge from certain tasks from being eclipsed during merging. Furthermore, consistent with the LLaVA results, the compressed PRISM ($r_m = 8$) continues to outperform all full-rank baselines. Experiments confirm that the capability of PRISM to integrate knowledge across multiple domains can be generalized across different MLLM backbones.

4.3 Ablation Study

To evaluate the contribution of each component in PRISM, we present the results of our ablation study in Table 3. We established a baseline that employs random pruning, where hierarchical components are randomly selected from the global pool and aggregated without orthogonality.

Validation of spectral concentration. The most significant performance improvement stems from replacing random selection with our proposed Spectral Pruning, boosting the average accuracy from 43.60% to 54.69%. This result corroborates our observation in Section 3.2 regarding the intrinsic redundancy of LoRA. Spectral Pruning leverages singular values as a proxy for information density, ensuring that the limited rank budget is allocated to the high-energy directions that truly define task expertise. This prevents noise accumulation from weakening expert capabilities during the merging.

Resolution of high-energy conflicts. The impact of Energy-Prioritized Orthogonal Reconstruction offers a nuanced insight into the nature of inter-task interference. When applied to randomly pruned components (Row 3), it provides a negligible gain. However, when applied after Spectral Pruning (Row 4), it delivers further substantial improvements. This phenomenon aligns with our analysis in Figure 3, which reveals that structural conflicts are more pronounced in the dominant spectral components. It is worth noting that the benefits of orthogonalization are non-uniform, favoring tasks that are more susceptible to interference. Comparing Row 2 and Row 4, our orthogonal reconstruction demonstrates moderate gains in already robust domains like RS and Fin. In contrast, tasks struggling in the baseline significantly benefit from conflict resolution such as Med, Sci, and AD. PRISM effectively isolates these sensitive features from interference, ensuring that the merged model retains competence in complex domains without sacrificing the integrity of robust ones.

4.4 Analysis

Visualization. To intuitively understand the performance gap, Figure 5 visualizes examples of different settings. The comparison reveals that while baseline methods suffer from knowledge distortion and logical collapse, PRISM preserves expert-level precision. As observed in the Medical case, baselines like Ties-Merging and KnOTS exhibit hallucinations (identifying pleura as "Thyroid" or "Lung"). Meanwhile, RobustMerge degrades into superficial visual descriptions. This indicates that fine-grained domain features are drowned out by noise during merging. Similarly, in Math tasks, the comparison method fails to sustain the logical reasoning required for conditional counting, whereas PRISM performs robust reasoning. This corroborates that our spectral approach effectively isolates and protects crucial semantics from inter-task interference.

Layer-wise adaptive rank allocation. To further investigate the mechanism of Spectral Pruning, we visualized the distribution of retained ranks across different tasks at specific layers under the multi-ability learning setting. Figure 6 illustrates the percentage of ranks retained from each task in layer 0 (shallow) and layer 31 (deep). A key observation is the semantic shift in distribution of rank allocation. The math task consistently occupies the largest proportion of the spectral energy, increasing from 38.50% in the shallow layer to 42.89% in the deep layer. This aligns with the intuition that mathematical reasoning relies heavily on the complex logical processing capabilities, which are typically located in the deeper layers of the MLLMs. Conversely, the VP task exhibits a notable decline. This suggests that visual perception requires retaining fine-grained visual features at shallow levels and embedding visual labels into the semantic space of LLM, while their importance diminishes as models transition toward abstract reasoning. These results qualitatively validate the effectiveness of our energy-based allocation. PRISM dynamically allocates the rank budget to where it is most needed by prioritizing perception-heavy tasks in shallow layers and reasoning-heavy tasks in deep layers.

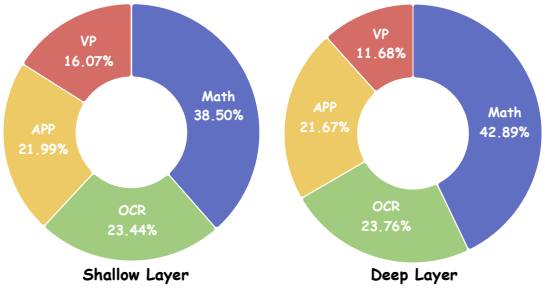

Figure 6: The percentage of components retained for each task in shallow and deep layer of the merged LoRA from PRISM. It automatically allocates more budget to inference-intensive tasks (e.g., Math) in deeper layers, while prioritizing vision-centric tasks (e.g., VP) in shallower layers.

5 Conclusion

In this work, we revisited parameter-efficient model merging of MLLMs from a spectral perspective. We reveal the concentrated distribution of task-specific knowledge in the dominant spectral components within LoRA, along with cross-task conflicts between high-energy directions. To address this, we proposed PRISM, a data-free framework that reformulates merging as a spectral signal reconstruction process. By synergizing Spectral Pruning to filter out task-irrelevant noise and Energy-Prioritized Orthogonal Reconstruction to resolve spatial conflicts, PRISM effectively harmonizes diverse task-specific experts into a unified model. Our approach not only ensures robust multi-task performance and high parameter efficiency but also paves the way for future research into modular and composable AI systems, enabling flexible knowledge integration without expensive retraining.

Acknowledgments

This work is funded by the National Key Research and Development Program of China (Grant No. 2026YFE0202100), the National Natural Science Foundation of China (Grant No. T254100759), the Independent Research Project of the National Key Laboratory of Big Data and Decision. This work was also supported by the WHU-Kingsoft Joint Lab. The supercomputing system at the Supercomputing Center of Wuhan University supported the numerical calculations in this paper.

References

[1] Manoj Acharya, Kushal Kafle, and Christopher Kanan. Tallyqa: Answering complex counting questions. In *AAAI*, volume 33, pages 8076–8084, 2019.

[2] Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *NeurIPS*, 35:23716–23736, 2022.

[3] Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. Qwen2.5-vl technical report. *arXiv preprint arXiv:2502.13923*, 2025.

[4] Shuaichen Chang, David Palzer, Jialin Li, Eric Fosler-Lussier, and Ningchuan Xiao. Mapqa: A dataset for question answering on choropleth maps. *arXiv preprint arXiv:2211.08545*, 2022.

[5] Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *ICML*, 2025.

[6] Guodong Du, Junlin Lee, Jing Li, Runhua Jiang, Yifei Guo, Shuyang Yu, Hanting Liu, Sim K Goh, Ho-Kin Tang, Daojing He, et al. Parameter competition balancing for model merging. *NeurIPS*, 37:84746–84776, 2024.

[7] Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodola. Task singular vectors: Reducing task interference in model merging. In *CVPR*, pages 18695–18705, 2025.

[8] Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*, pages 6904–6913, 2017.

[9] Ziyu Guo, Renrui Zhang, Hao Chen, Jialin Gao, Dongzhi Jiang, Jiaze Wang, and Pheng-Ann Heng. Sciverse: Unveiling the knowledge comprehension and visual reasoning of lmms on multi-modal scientific problems. In *Findings of ACL*, pages 19683–19704, 2025.

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

[11] Xuehai He, Yichen Zhang, Luntian Mou, Eric Xing, and Pengtao Xie. Pathvqa: 30000+ questions for medical visual question answering. *arXiv preprint arXiv:2003.10286*, 2020.

[12] Yu-Chung Hsiao, Fedir Zubach, Gilles Baechler, Srinivas Sunkara, Victor Cărbune, Jason Lin, Maria Wang, Yun Zhu, and Jindong Chen. Screenqa: Large-scale question-answer pairs over mobile app screenshots. In *NAACL*, pages 9427–9452, 2025.

[13] Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *ICLR*, 2022.

[14] Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. Lorahub: Efficient cross-task generalization via dynamic lora composition. *arXiv preprint arXiv:2307.13269*, 2023.

[15] Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. Emr-merging: Tuning-free high-performance model merging. *NeurIPS*, 37:122741–122769, 2024.

[16] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *ICLR*, 2023.

[17] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *ECCV*, pages 709–727. Springer, 2022.

[18] Justin Johnson, Bharath Hariharan, Laurens Van Der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. In *CVPR*, pages 2901–2910, 2017.

[19] Aniruddha Kembhavi, Mike Salvato, Eric Kolve, Minjoon Seo, Hannaneh Hajishirzi, and Ali Farhadi. A diagram is worth a dozen images. In *ECCV*, pages 235–251, 2016.

[20] Aniruddha Kembhavi, Minjoon Seo, Dustin Schwenk, Jonghyun Choi, Ali Farhadi, and Hannaneh Hajishirzi. Are you smarter than a sixth grader? textbook question answering for multimodal machine comprehension. In *CVPR*, pages 4999–5007, 2017.

[21] Muhammad Uzair Khattak, Hanoona Rasheed, Muhammad Maaz, Salman Khan, and Fahad Shahbaz Khan. Maple: Multi-modal prompt learning. In *CVPR*, pages 19113–19122, 2023.

[22] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *PNAS*, pages 3521–3526, 2017.

[23] Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *ICML*, pages 19730–19742, 2023.

[24] Zhang Li, Biao Yang, Qiang Liu, Zhiyin Ma, Shuo Zhang, Jingxu Yang, Yabo Sun, Yuliang Liu, and Xiang Bai. Monkey: Image resolution and text label are important things for large multi-modal models. In *CVPR*, pages 26763–26773, 2024.

[25] Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin A Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. *NeurIPS*, 35:1950–1965, 2022.

[26] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *NeurIPS*, 36: 34892–34916, 2023.

[27] Junpeng Liu, Tianyue Ou, Yifan Song, Yuxiao Qu, Wai Lam, Chenyan Xiong, Wenhu Chen, Graham Neubig, and Xiang Yue. Harnessing webpage uis for text-rich visual understanding. In *ICLR*, 2025.

[28] 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 *ICML*, 2024.

[29] Yuliang Liu, Zhang Li, Mingxin Huang, Biao Yang, Wenwen Yu, Chunyuan Li, Xu-Cheng Yin, Cheng-Lin Liu, Lianwen Jin, and Xiang Bai. Ocrbench: on the hidden mystery of ocr in large multimodal models. *SCIS*, 67(12):220102, 2024.

[30] Sylvain Lobry, Diego Marcos, Jesse Murray, and Devis Tuia. Rsvqa: Visual question answering for remote sensing data. *IEEE TGRS*, 58(12):8555–8566, 2020.

[31] Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. Mathvista: Evaluating mathematical reasoning of foundation models in visual contexts. *arXiv preprint arXiv:2310.02255*, 2023.

[32] Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D Bagdanov, and Joost van de Weijer. No task left behind: Isotropic model merging with common and task-specific subspaces. In *ICML*, 2025.

[33] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychol. Learn. Motiv.*, pages 109–165. 1989.

[34] 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 *NeurIPS*, 2025.

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

[36] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763, 2021.

[37] Viraj Shah, Nataniel Ruiz, Forrester Cole, Erika Lu, Svetlana Lazebnik, Yuanzhen Li, and Varun Jampani. Ziplora: Any subject in any style by effectively merging loras. In *ECCV*, pages 422–438, 2024.

[38] Wenhao Shi, Zhiqiang Hu, Yi Bin, Junhua Liu, Yang Yang, See Kiong Ng, Lidong Bing, and Roy Ka-Wei Lee. Math-llava: Bootstrapping mathematical reasoning for multimodal large language models. In *Findings of EMNLP*, pages 4663–4680, 2024.

[39] Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. Drivelm: Driving with graph visual question answering. In *ECCV*, pages 256–274, 2024.

[40] George Stoica, Daniel Bolya, Jakob Bjorner, Pratik Ramesh, Taylor Hearn, and Judy Hoffman. Zipit! merging models from different tasks without training. *arXiv preprint arXiv:2305.03053*, 2023.

[41] George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with svd to tie the knots. In *ICLR*, 2025.

[42] Anke Tang, Li Shen, Yong Luo, Yibing Zhan, Han Hu, Bo Du, Yixin Chen, and Dacheng Tao. Parameter-efficient multi-task model fusion with partial linearization. In *ICLR*, 2024.

[43] Qwen Team. Qwen2.5: A party of foundation models, September 2024.

[44] Peter Tong, Ellis Brown, Penghao Wu, Sanghyun Woo, Adithya Jairam Vedagiri IYER, Sai Charitha Akula, Shusheng Yang, Jihan Yang, Manoj Middepogu, Ziteng Wang, et al. Cambrian-1: A fully open, vision-centric exploration of multimodal llms. *NeurIPS*, 37:87310–87356, 2024.

[45] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[46] 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 *EMNLP Workshop*, pages 353–355, 2018.

[47] Bryan Wang, Gang Li, Xin Zhou, Zhourong Chen, Tovi Grossman, and Yang Li. Screen2words: Automatic mobile ui summarization with multimodal learning. In *UIST*, pages 498–510, 2021.

[48] Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jiménez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. In *ICML*, pages 50268–50287, 2024.

[49] Yuanyi Wang, Yanggan Gu, Yiming Zhang, Qi Zhou, Zhaoyi Yan, Congkai Xie, Xinyao Wang, Jianbo Yuan, and Hongxia Yang. Model merging scaling laws in large language models. *arXiv preprint arXiv:2509.24244*, 2025.

[50] Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *NeurIPS*, 36:7093–7115, 2023.

[51] Prateek Yadav, Colin Raffel, Mohammed Muqeeth, Lucas Caccia, Haokun Liu, Tianlong Chen, Mohit Bansal, Leshem Choshen, and Alessandro Sordoni. A survey on model moerging: Recycling and routing among specialized experts for collaborative learning. *TMLR*, 2025.

[52] Kunda Yan, Min Zhang, Sen Cui, Qu Zikun, Bo Jiang, Feng Liu, and Changshui Zhang. Calm: Consensus-aware localized merging for multi-task learning. In *ICML*, 2025.

[53] Zhaoyi Yan, Yiming Zhang, Baoyi He, Yuhao Fu, Qi Zhou, Zhijie Sang, Chunlin Ji, Shengyu Zhang, Fei Wu, and Hongxia Yang. Infifusion: A unified framework for enhanced cross-model reasoning via llm fusion. *arXiv preprint arXiv:2501.02795*, 2025.

[54] Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. Model merging in llms, mllms, and beyond: Methods, theories, applications, and opportunities. *ACM Computing Surveys*, 2024.

[55] Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. In *ICLR*, 2024.

[56] Kaining Ying, Fanqing Meng, Jin Wang, Zhiqian Li, Han Lin, Yue Yang, Hao Zhang, Wenbo Zhang, Yuqi Lin, Shuo Liu, et al. Mmt-bench: A comprehensive multimodal benchmark for evaluating large vision-language models towards multitask agi. *arXiv preprint arXiv:2404.16006*, 2024.

[57] Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *ICML*, 2024.

[58] Fanhu Zeng, Haiyang Guo, Fei Zhu, Li Shen, and Hao Tang. Robustmerge: Parameter-efficient model merging for mllms with direction robustness. In *NeurIPS*, 2025.

[59] Han-Chen Zhang, Zi-Hao Zhou, Mao-Lin Luo, Shimin Di, Min-Ling Zhang, and Tong Wei. Dc-merge: Improving model merging with directional consistency, 2026. URL `https://arxiv.org/abs/2603.06242`.

[60] Jinghan Zhang, Junteng Liu, Junxian He, et al. Composing parameter-efficient modules with arithmetic operation. *NeurIPS*, 36:12589–12610, 2023.

[61] Renrui Zhang, Xinyu Wei, Dongzhi Jiang, Ziyu Guo, Shicheng Li, Yichi Zhang, Chengzhuo Tong, Jiaming Liu, Aojun Zhou, Bin Wei, et al. Mavis: Mathematical visual instruction tuning with an automatic data engine. *arXiv preprint arXiv:2407.08739*, 2024.

[62] Hongbo Zhao, Fei Zhu, Haiyang Guo, Meng Wang, Rundong Wang, Gaofeng Meng, and Zhaoxiang Zhang. Mllm-cl: Continual learning for multimodal large language models. *arXiv preprint arXiv:2506.05453*, 2025.

[63] Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, and Fei Wu. Merging loras like playing lego: Pushing the modularity of lora to extremes through rank-wise clustering. In *ICLR*, 2025.

[64] Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *NeurIPS*, 36:46595–46623, 2023.

[65] Shenghe Zheng, Hongzhi Wang, Chenyu Huang, Xiaohui Wang, Tao Chen, Jiayuan Fan, Shuyue Hu, and Peng Ye. Decouple and orthogonalize: A data-free framework for lora merging. *arXiv preprint arXiv:2505.15875*, 2025.

[66] Didi Zhu, Yibing Song, Tao Shen, Ziyu Zhao, Jinluan Yang, Min Zhang, and Chao Wu. Remedy: Recipe merging dynamics in large vision-language models. In *ICLR*, 2025.

[67] Heming Zou, Yunliang Zang, Wutong Xu, Yao Zhu, and Xiangyang Ji. Flylora: Boosting task decoupling and parameter efficiency via implicit rank-wise mixture-of-experts. In *NeurIPS*, 2025.

A Dataset Details

To evaluate the efficacy of the merging algorithm in integrating diverse expert capabilities, we adopt the MLLM-CL benchmark introduced by [62]. While MLLM-CL was originally designed for sequential continual learning (where tasks arrive one by one), we adapt it for a Multi-Task Model Merging setting. In our experiments, we treat each dataset as an independent domain or capability. We first train individual LoRA experts on the training split of each task, simulating the creation of specialized experts, then merge them to evaluate the performance of the unified model on the corresponding test set. This setting rigorously tests the ability of the merging algorithm to handle both heterogeneous feature distributions and conflicting parameter updates. We provide the data sources and data volumes for different tasks in Table 4.

Table 4: Statistics of the training datasets and test datasets for multi-domain learning and multi-ability learning.

Setting	Task	Train Dataset	Test Dataset	Train Number	Test Number
Multi-Domain Learning	RS	RSVQA [30]	RSVQA	60k	10k
	Med	PathVQA [11]	PathVQA	22.8k	9.8k
	AD	DriveLM [39]	DriveLM	60k	10k
	Sci	AI2D [19] SciVerse [9] MapQA [4] TQA [20]	AI2D SciVerse MapQA TQA	33.4k	8.2k
	Fin	StockQA [62]	StockQA	60k	10k
Multi-Ability Learning	OCR	Monkey [24]	OCRBench [29]	128.1k	1k
	Math	MathV360K [38] MAVIS [61]	MathVista [31]	526.1k	1k
	VP	CLEVR [18] TallyQA [1]	CV-Bench [44]	119.9k	0.8k
	GUI	ScreenQA [12] MultiUI [27] Screen2Words [47]	MMTBench [56]	147.3k	0.8k

B More Implementation Details

We conducted experiments on both the LLaVA-v1.5 and Qwen2.5-VL models to validate the robustness of our approach across different foundational model architectures. Specifically, LLaVA-v1.5-7B employs vicuna-v1.5-7B [64] as its language backbone and CLIP-L-336 as its visual encoder [36]. Qwen2.5-VL-3B [3] is built upon Qwen2.5 [43] as its language model, utilizing a redesigned Vision Transformer (ViT) architecture as its visual encoder.

B.1 Training Details

Multi-domain learning. Table 5 shows the hyperparameters for training a LoRA expert on a single task in multi-domain learning. For most configurations, we follow the default setting of LLaVA-v1.5-7B. We set the same number of training steps for each task to ensure comparable training exposure across datasets of different sizes. For parameter efficient parameter fine-tuning, the LLaVA-v1.5-7B model employs a rank of 128. To evaluate the effectiveness of PRISM with lower ranks of LoRA, we set the rank to 16 for each task in our experiments on the Qwen2.5-VL-3B model. We initialize the down-projection matrix $\mathbf{A}_i$ using Kaiming uniform distribution and the up-projection matrix $\mathbf{B}_i$ with zeros. For all the experiments, we use 8 RTX 4090 GPUs.

Multi-ability learning. Table 6 shows the hyperparameters for training a LoRA on a single task in multi-ability learning. Qwen2.5-VL-3B is not trained under this configuration. Since Qwen2.5-

Table 5: Training configurations in multi-domain learning.

	LLaVA-1.5-7B	Qwen2.5VL-3B
batch size	32	64
base lr	2×10^{-5}	1×10^{-5}
training steps	2k	1k
LoRA rank	128	16
optimizer	AdamW	
lr warmup ratio	0.03	
lr schedule	cosine decay	
DeepSpeed stage	2	

Table 6: Training configurations in multi-ability learning.

	LLaVA-1.5-7B
batch size	32
base lr	2×10^{-5}
training steps	16k
LoRA rank	128
optimizer	AdamW
lr warmup ratio	0.03
lr schedule	cosine decay
DeepSpeed stage	2

VL-3B already possesses strong OCR and mathematical capabilities, fine-tuning would not yield significant improvements.

B.2 Evaluation Details

Remote sensing, medical, and autonomous driving tasks follow default evaluation criterion: if the label is numerical, the prediction must yield the same numerical value; if the label is Yes/No, the prediction must correspond to Yes/No; in other cases, the prediction is considered correct if it either perfectly matches the label or contains the label. For science tasks, some test samples are multiple-choice questions (MCQs), where predictions require exact matching of the letter corresponding to the answer option; enumeration-type questions require the predicted set of items to match the ground truth; all other issues are evaluated according to the default criterion. The finance section consists solely of multiple-choice and true/false questions, with the same evaluation criterion as similar questions mentioned above. In multi-ability learning, we follow the default setting of the corresponding benchmarks.

C Details of Comparison Methods

We compare PRISM against a wide range of state-of-the-art model merging techniques. These baselines fall into two categories: (1) merging methods originally designed for FFT which we re-implemented for PEFT scenarios, and (2) methods specifically tailored for parameter-efficient modules. For all comparisons, we merge the task-specific LoRA experts into a unified adapter and evaluate it under the same inference configuration. We briefly introduce the comparison methods in the following paragraphs. First is the merging methods for FFT:

PCB-Merging introduces a parameter competition balancing mechanism to resolve conflicts. It evaluates the significance of parameters through intra-task balancing and inter-task balancing. Concretely, it calculates a competition score for each parameter, prunes those with low importance scores, and rescales the remaining parameters to form the final merged model without requiring additional training data.

Ties-Merging mitigates parameter interference by addressing redundancy and sign conflicts. It first applies a trim operation to retain only the parameters with the top-$k\%$ magnitude, and then resolves sign disagreements by an elect sign operation, which keeps the direction of the cumulative sum. Finally, it computes the mean of the preserved parameters that align with the elected sign.

DARE operates on the premise that fine-tuned parameters are highly sparse. It randomly drops a large proportion of the delta parameters according to a specified drop rate p and subsequently rescales the surviving parameters by a factor of $1/(1-p)$. This approach aims to approximate the original expectation of the parameter updates while reducing redundancy.

These FFT-based methods are highly effective for full-parameter tuning by primarily aggregating parameters at the element level. Recognizing that element-wise manipulation may not fully preserve the intrinsic low-rank geometric structure of PEFT modules, several methods have been specifically tailored for merging LoRA modules:

LoRALEGO focuses on the composability of LoRA modules. It treats LoRA weights as building blocks and employs a stitching-based approach or linear interpolation optimized for modular composition. By ensuring that the low-rank matrices are structurally compatible, it enables the efficient combination of multiple task-specific adapters into a single cohesive unit.

RobustMerge emphasizes the preservation of direction robustness in the low-rank space. It prunes parameters with small magnitudes that contribute to direction instability and employs complementary parameter scaling and cross-task normalization to align the singular values, ensuring that the merged model retains the critical directional information of each task.

KnOTS leverages SVD to align the task-specific updates into a shared geometric space. It concatenates the LoRA updates from different tasks and performs decomposition to find a core space that best represents the collective knowledge. By aggregating the parameters directly within this common projection basis, it reduces the misalignment often observed in standard LoRA merging. Specifically, we use the KnOTS implementation based on Ties-Merging.

DC-Merge preserves the directional consistency of task vectors by applying energy smoothing to their singular values and aggregating the weights within a shared orthogonal cover space constructed via whitening.

D More Disscusion

D.1 Performance of Homogeneous Knowledge Aggregation

Beyond resolving conflicts across heterogeneous domains, a robust merging algorithm should also effectively aggregate homogeneous knowledge from the same domain. To evaluate this capability, we conducted experiments by redistributing data from remote sensing and medical tasks. For each domain, we randomly partitioned the training set into five disjoint subsets and trained an individual LoRA expert on each partition. These LoRA experts are then merged and evaluated on the complete test set. The results are presented in Table 7. The "Average of Individual LoRAs" column calculates the average performance of the five experts without any merging. Essentially, it reflects the expected capability of a model that has only seen a fraction $(1/5)$ of the data. As observed, most existing methods fail to effectively synthesize the distributed information, often performing worse than or comparable to the individual average. In contrast, PRISM achieves the highest performance across both domains. This indicates that despite employing an apparent orthogonalization strategy, PRISM does not compromise homogenous knowledge. Notably, our method surpasses the average performance of individual experts, demonstrating its ability to constructively integrate partial knowledge into a more powerful global model.

Table 7: Aggregating homogeneous experts trained on disjoint data partitions. The "Individual LoRAs" column serves as a baseline, representing the **mean performance** of single experts on the global test set. PRISM demonstrates superior capability in synthesizing partial knowledge compared to SOTA methods.

Method	Individual LoRAs	DARE	Ties-Merging	RobustMerge	KnOTS	PRISM
RS	77.87	76.89	71.84	78.06	78.06	**78.43**
Med	46.63	44.73	45.20	46.83	42.27	**47.66**

D.2 Additional Spectral Similarity Analysis across Layers

To verify that the spectral properties observed in the main text are not specific to intermediate layers, we extend our visualization to the shallow layer and deep layer of the model. As shown in Figure 7, the patterns of spectral concentration and cross-task interference remain largely consistent across different depths and attention components.

D.3 Selection of the merged-rank budget.

We evaluate the cumulative-energy rule on Qwen2.5-VL-3B using five multi-domain experts with rank $r = 16$. Table 8 reports the selected rank budgets and average accuracy at different approximate

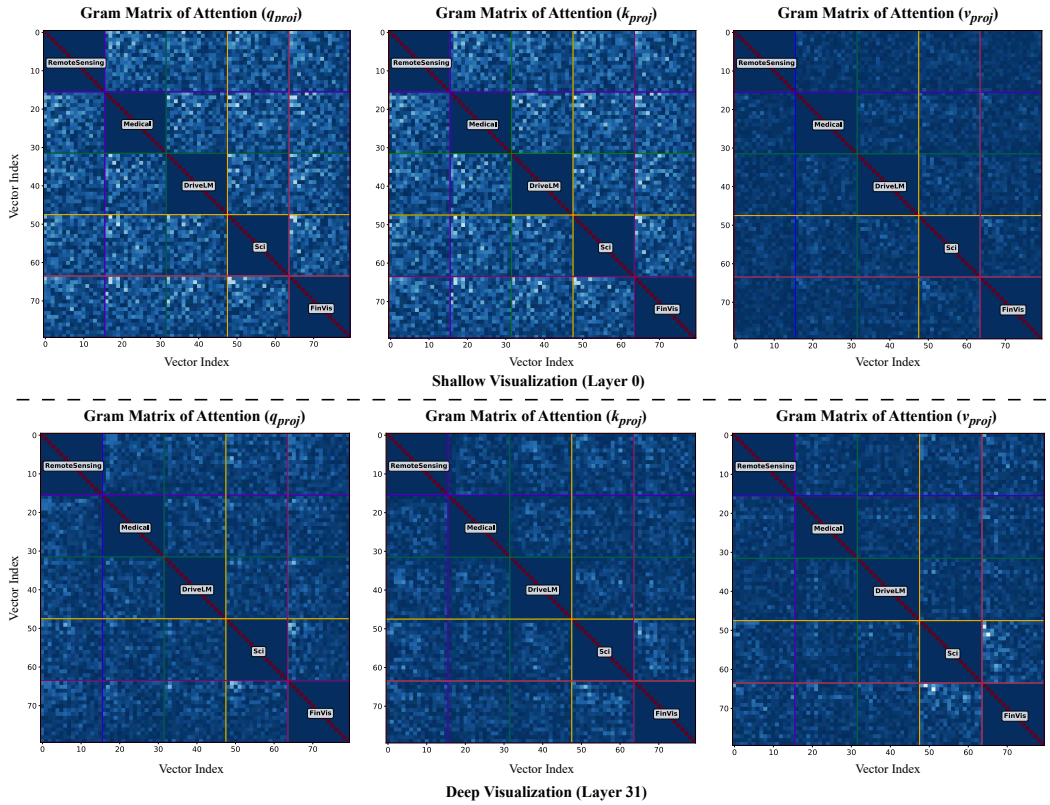

Figure 7: Cross-task spectral similarity of LoRA attention updates. Gram matrices of the top-k right singular vectors of LoRA-induced weights, computed across five tasks. Color intensity denotes the absolute cosine similarity: reddish values close to 1 correspond to highly aligned or misaligned directions, while bluish values close to 0 indicate near-orthogonal alignment. Diagonal blocks correspond to intra-task similarity, while off-diagonal blocks indicate cross-task similarity.

Table 8: Rank-budget selection on the Qwen2.5-VL-3B multi-domain benchmark. Each individual expert has rank $r = 16$.

Target energy	$\sim 75\%$	$\sim 80\%$	$\sim 85\%$	$\sim 90\%$	$\sim 95\%$
Selected rank	8	11	16	26	40
Average accuracy (%)	65.14	65.94	66.10	66.44	66.59

energy coverage levels. Rank 16 corresponds to the default setting $r_m = r$. At approximately 80% energy coverage, rank 11 achieves 99.0% of the average accuracy obtained at rank 40 while using 27.5% of its adapter parameters. Increasing the rank from 11 to 40 adds 0.65 accuracy points. These results support 80% to 85% energy coverage as a practical reference range in this setting, while the default configuration remains $r_m = r$.

D.4 Comparison of Merge Efficiency.

We further evaluate the merging efficiency of PRISM against representative PEFT merging baselines in the multi-domain setting. All methods are run on CPU to avoid confounding factors from GPU-specific kernels and implementation-level parallelization, and to enable a fair comparison within the same execution environment. and we report wall-clock runtime, peak memory usage, and average accuracy. PRISM achieves the best accuracy of 57.51% with the lowest memory usage among all compared methods. Its runtime is also competitive with KnOTS and LoRALEGO, while delivering substantially higher accuracy. Although RobustMerge runs faster, it requires more than twice the

memory of PRISM and obtains a lower average accuracy. These results show that PRISM improves merging performance without incurring prohibitive computational or memory overhead.

Table 9: Merging efficiency comparison under the *multi-domain setting*. All methods are run on CPU, and we report wall-clock runtime, peak memory usage during merging, and average accuracy.

Method	LoRALEGO	RobustMerge	KnOTS	PRISM (Ours)
Time (s) ↓	995	**291**	1008	834
Peak Memory (GB) ↓	5.2	10.4	6.6	**4.3**
Average Accuracy (%) ↑	39.78	51.00	52.01	**57.51**

E Limitation and Impact Statement

Although PRISM demonstrates strong effectiveness and efficiency for parameter-efficient merging of MLLMs, this work mainly focuses on LoRA-based experts that are fine-tuned from a shared backbone. PRISM performs its spectral operations on the reconstructed weight increment ΔW, so these operations do not depend on the factorization used to represent the same update. Extension to other PEFT variants requires compatible additive low-rank weight updates in the same parameter space. Methods such as adapter or prefix tuning do not directly fit the current formulation. Controlled evaluation beyond LoRA and across heterogeneous architectural configurations remains future work. In addition, PRISM currently adopts a prescribed merged-rank budget to ensure a controllable parameter cost and a fair comparison. While this design avoids introducing additional tuning hyperparameters, adaptive rank allocation across layers or tasks may further improve the efficiency–performance trade-off in more large-scale merging scenarios.

This work proposes a novel approach for merging parameter-efficient modules within MLLMs, aiming to advance the field of machine learning, particularly in multi-task learning. Our research seeks to enhance the efficiency and accessibility of developing versatile AI systems. The societal impact of this work is multifaceted. On the positive side, our approach may reduce energy consumption associated with large-scale model development. Such efficiency enables organizations with limited computational resources to more readily access advanced multimodal capabilities, potentially accelerating progress in healthcare, environmental science, and education.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We detail the proposed PRISM framework in Section 3 and comprehensively evaluate its efficacy across diverse benchmarks in Section 4, substantiating the claims made in the abstract and introduction.

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We explicitly discuss the limitations of our method in Appendix E.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [N/A]

Justification: Our work focuses on an empirical methodology for model merging and does not introduce new theoretical theorems or mathematical proofs.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide comprehensive implementation details, including foundation models, baseline configurations, compute resources, and hyperparameters in 4.1 and Appendices A and B.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: We state that our code and weights will be released in the Abstract.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Training configurations, batch sizes, optimizers, and learning rates are detailed in Appendix B. Data splits for all tasks are provided in Appendix A.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: All merging methods are evaluated under identical experimental conditions, utilizing the same task-specific LoRA experts, rank budgets, and evaluation protocols. The merging process is strictly deterministic and training-free, involving no stochastic optimization or random initialization. Consequently, the reported results provide a highly controlled comparison under identical settings, although they do not quantify the variance of LoRA experts independently trained with different random seeds.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We specify the use of 8 NVIDIA RTX 4090 GPUs for all our training and merging experiments in Appendix B.1.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics, focusing on efficient model merging using public datasets without malicious intent or ethical violations.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We explicitly discuss the impacts of our work in Appendix E.

    Guidelines:

- The answer [N/A] means that there is no societal impact of the work performed.
- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: We utilize existing, publicly released foundational models and academic benchmarks. Our method does not introduce new high-risk misuse scenarios or scrape new unvetted data.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We appropriately cite all foundation models (LLaVA-v1.5, Qwen2.5-VL) and public datasets (RSVQA, PathVQA, etc.) in 4.1 and Appendix A, respecting their academic terms of use.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: We do not curate or release new datasets; we adapt existing public benchmarks for our evaluation protocol.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our research relies purely on computational experiments with existing data and does not involve crowdsourcing or human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The research does not involve human subjects, so IRB approval is not applicable.

    Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: We do not use LLMs as an original component of the core method development.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.