Label-Free Cross-Task LoRA Merging with Null-Space Compression

Wonyoung Lee*
KAIST
wylee@kaist.ac.kr

Wooseong Jeong*
KAIST
stk14570@kaist.ac.kr

Kuk-Jin Yoon
KAIST
kjyoon@kaist.ac.kr

Abstract

Model merging combines independently fine-tuned checkpoints without joint multi-task training. In the era of foundation-model, fine-tuning with Low-Rank Adaptation (LoRA) is prevalent, making LoRA merging a promising target. Existing approaches can work in homogeneous settings where all target tasks are classification but often fail when tasks span classification and regression. Approaches using entropy-based surrogates do not apply to regression and are costly for large language models due to long token sequences. We introduce Null-Space Compression (NSC) Merging*, a label-free, output-agnostic method that sets merge weights from adapter geometry. Our key observation is that during LoRA finetuning the down-projection factor A in $\Delta W = BA$ compresses its null space, and the compression correlates with performance. NSC uses this as an optimization signal for merging that can generalize across classification, regression, and sequence generation. NSC achieves state-of-the-art performance across twenty heterogeneous vision tasks with balanced gains where prior methods overfit subsets of tasks. It also outperforms baselines on six NLI benchmarks and on vision–language evaluations for VQA and image captioning, demonstrating scalability and effectiveness. Our code is available at* https://github.com/wonyoung01/nsc_merging.

1. Introduction

Modern deep learning models, including vision transformers [16] and large foundation models such as LLMs [18] and VLMs [46, 47], achieve strong performance and exhibit broad generalization to unseen tasks. Despite their scale, these models still require task-specific fine-tuning when the target distribution or objective departs from what was covered during pretraining. Conventional full fine-tuning, however, is computationally and memory intensive, limiting its practicality in many real-world settings.

This challenge has motivated the development of parameter-efficient fine-tuning (PEFT) methods [27, 54]. Among them, Low-Rank Adaptation (LoRA) [28] has become a widely adopted strategy for adapting large models while keeping base weights frozen. LoRA is advantageous for training and deployment, as it substantially reduces the number of trainable parameters during fine-tuning and produces compact adapter modules that are easy to distribute and share through open-source model hubs.

Beyond single-task adaptation, modern systems are expected to handle multiple downstream tasks. The traditional solution is multi-task learning [31–35, 44, 45, 99], which requires a unified training pipeline and a joint dataset with aligned per-task labels. This is costly in curation and computation, and often infeasible when labels are proprietary or restricted by license. It also makes it difficult to leverage numerous task-specialized models that are scattered across open model hubs, because those checkpoints cannot be reused directly without centralized retraining. These constraints motivate model merging, where independently finetuned checkpoints are combined without revisiting labeled data. In the foundation-model era, model merging is especially effective with LoRA adapters. One can merge or compose adapters trained on diverse tasks, retrieved independently from model hubs, to synthesize a single model that inherits their strengths while preserving the benefits of lightweight storage and easy distribution.

Conventional model merging has largely focused on full model weights [30, 36, 90, 91, 98]. Task Arithmetic [30] constructs task vectors as differences between fine-tuned and pretrained checkpoints, then adds them with a scale chosen on a small validation split. In contrast, LoRA-aware merging [71, 106] operates directly on the low-rank adapters rather than the full weights. Working in the low-rank space is memory and compute efficient and it performs markedly better in LoRA settings, where conventional full-weight mergers can degrade due to strong subspace misalignment between adapters [71]. Given that LoRA fine-tuning is now standard for LLMs and VLMs, LoRA-targeted merging is a promising direction.

Another advantage of LoRA merging is that it can use gradient signals to set merge weights across checkpoints,

*Equal contribution to this work.

which is prohibitive at the scale of LLMs or VLMs when operating on full weights. AdaMerging [91] is a representative gradient-based method that estimates merge weights without labels by minimizing output entropy as a surrogate objective. This surrogate is standard in test-time adaptation [79] and has been adopted by AdaMerging and subsequent work [6, 74, 85], yielding strong gains on classification. However, entropy-based approaches face fundamental limitations in broader tasks and domains. First, entropy does not apply to regression problems such as depth estimation or surface normal prediction. Second, for LLMs and VLMs, entropy must be computed at each token prediction, so cost grows with the number of generated tokens and quickly becomes a bottleneck for *entropy-based* merging.

To address these issues, we introduce *Null-Space Compression (NSC) Merging*. Our key observation is that during LoRA fine-tuning the down-projection factor $\boldsymbol{A}$ in $\Delta \boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}$ defines a projection onto a low-dimensional subspace of the input features, and the corresponding null-space ratio, which is the proportion of activation discarded by this projection, decreases over training. This decrease is closely correlated with task performance. We use this null-space compression as a label-free signal to determine merging weights across adapters. Unlike entropy-based surrogates, NSC applies naturally to both classification and regression and remains efficient for long sequence generation. NSC Merging demonstrates superior performance across heterogeneous scenarios that mix classification and regression, and we validate it on 20 vision tasks, natural language inference for LLMs, and Vision Question Answering (VQA) and image captioning for VLMs.

Our main contribution are as follows:

- We observe LoRA finetuning systematically compresses the null space of the down-projection factor $\boldsymbol{A}$, and the size of null space is closely correlated with the peformance of LoRA adapted model, that can be used as learning signal.
- Based on the observations, we suggest Null-Space Compression (NSC) Merging using a effective gradient-based adaptation signal for model merging which can be used for various scenario where entropy-based methods cannot be applied or inefficient.
- We validate our methods with extensive experiments including various models and benchmarks including conventional vision encoder, LLM, VLM and shows state-of-the-art performance compared to previous model merging baselines.

2. Related Work

Model Merging. Model merging combines knowledge from models that were fine-tuned separately, avoiding joint multi-task training and reducing computational cost while aiming to preserve task performance [92]. Representative methods include *Task Arithmetic*, which adds or subtracts task vectors [30], *RegMean*, which uses inner-product statistics of layer inputs to enable data-free fusion [36], *TIES*, which resolves sign conflicts and prunes small updates [90], and *DARE*, which sparsifies parameter deltas [98]. *AdaMerging* estimates layer-wise merging coefficients by minimizing an entropy-based objective to improve multi-task utility [6, 74, 85, 91]. When models are trained on different data or initialized differently, permutation alignment helps maintain linear mode connectivity [1, 20]. *ZipIt* aligns intermediate features to re-basin networks and match neurons across models, enabling training-free layer alignment [70]. Uncertainty-aware strategies mitigate mismatch through Fisher-weighted averaging [58] and through uncertainty or gradient matching [12]. Pareto merging [6] motivate multi-objective treatments considering preferences.

LoRA Merging. Low-Rank Adaptation (LoRA) [28] is a standard fine-tuning mechanism for large networks, including foundation models such as LLMs [18]. Conventional full-parameter merging methods [30, 36, 90, 98] transfer poorly to LoRA adapters, which has motivated LoRA-specific approaches [71, 76, 106]. Stoica et al. [71] report that LoRA-updated models exhibit weaker cross-model representation alignment than full-rank fine-tunes and propose KNOTS, which concatenates adapter updates and applies SVD to align them in a shared subspace before merging principal components. Zhao et al. [106] introduce minimal semantic units and use clustering to assemble a merged adapter with adjustable effective rank. Compositions of multiple LoRAs have also been explored for image generation, including multi-LoRA composition and concept mixing [22, 108, 111].

Conventional model merging underperforms on LoRA-adapted models, whereas LoRA-targeted merging performs better. Given the prevalence of PEFT in foundation models, LoRA-targeted merging is a promising direction. In the LoRA regime, gradient-based methods such as AdaMerging are feasible in compute and memory, unlike merging full-rank fine-tuned weights. However, AdaMerging relies on entropy minimization, which does not apply to regression and scales poorly to LLMs and VLMs. In this paper, we propose a new gradient-based method that is efficient, handles both classification and regression, and scales to LLMs and VLMs where prior approaches are inefficient. Further discussion of prior work and our approach appears in Section B of the supplementary material.

3. Method

To extend model merging to broader tasks and domains, we propose a gradient-based merging algorithm, termed *Null-Space Compression (NSC) merging*. The optimization ob-

jective stems from our key observation about the *dynamics of LoRA layers during fine-tuning*, which we refer to as *null-space compression*. Our method is designed to be task-agnostic: it optimizes layer-wise merging coefficients using only *unlabeled data* and *structural information* available from the LoRA adapter. Before delving into this phenomenon, we outline the goal of our approach.

3.1. Preliminaries

Consider a pretrained base model with L layers and parameters $\{\boldsymbol{W}_0^\ell\}_{\ell=1}^L$, where $\boldsymbol{W}_0^\ell$ denotes the parameter tensor at layer ℓ. Let there be K downstream tasks indexed by $k \in \{1, \dots, K\}$. For each task k, let $\{\boldsymbol{W}_k^\ell\}_{\ell=1}^L$ be the parameters of the model obtained by fine-tuning on task k starting from the base model. The difference $\Delta\boldsymbol{W}_k^\ell = \boldsymbol{W}_k^\ell - \boldsymbol{W}_0^\ell$ is referred to as a *task vector* [30]. Model merging [92] aims to combine these task vectors into a single model, allowing efficient multi-task adaptation and knowledge integration.

With the rise of large-scale foundation models [18, 46], full fine-tuning is often infeasible, and parameter-efficient fine-tuning with LoRA [28] has been widely adopted for its simplicity and efficiency. Consequently, principled merging strategies tailored to LoRA are essential. In this paper, we propose a merging strategy for LoRA-fine-tuned models and validate it across diverse tasks and models. We denote the task-specific LoRA as $\{\Delta\boldsymbol{W}_k^\ell\}_{\ell\in\mathcal{J}}$, where $\mathcal{J} \subseteq \{1, \dots, L\}$ is the subset of layers equipped with adapters. Each task vector $\Delta\boldsymbol{W}_k^\ell$ is expressed as

$$\Delta\boldsymbol{W}_k^\ell = \boldsymbol{B}_k^\ell \boldsymbol{A}_k^\ell, \quad \boldsymbol{B}_k^\ell \in \mathbb{R}^{d_{\text{out}}^\ell \times r_k}, \ \boldsymbol{A}_k^\ell \in \mathbb{R}^{d_{\text{in}}^\ell \times r_k} \tag{1}$$

where r_k denotes the adapter rank for the k-th task and typically satisfies $r_k \ll \min(d_{\text{in}}^\ell, d_{\text{out}}^\ell)$.

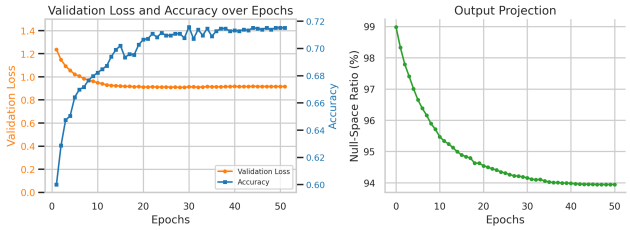
Given layer-wise coefficients λ_k^ℓ for task k, the merged update is defined as

$$\Delta\boldsymbol{W}_{\text{merge}}^\ell = \begin{cases} \sum_{k=1}^K \lambda_k^\ell \, \Delta\boldsymbol{W}_k^\ell, & \ell \in \mathcal{J}, \\ \boldsymbol{0}, & \ell \notin \mathcal{J}. \end{cases} \tag{2}$$

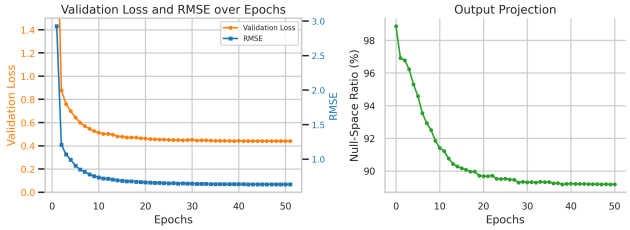
Our goal is to learn optimal coefficients $\{\lambda_k^\ell\}$ such that the merged model $\boldsymbol{\Theta}_{\text{merge}} = \{\boldsymbol{W}_0^\ell + \Delta\boldsymbol{W}_{\text{merge}}^\ell\}_{\ell=1}^L$ maximizes overall performance across tasks, balancing shared representations and task-specific specialization.

3.2. Null-space compression in LoRA fine-tuning

Considering the structure of LoRA, we denote $\boldsymbol{A}_k^\ell$ in Eq. (1) as down-projection matrix. This down-projection matrix inherently project incoming activations into a reduced subspace, and hence discarding components that fall into the *null space*. We refer to the proportion of activation suppressed by this projection as the *null-space ratio*. Specifically, we define the null-space ratio of an adapter weight



(a) Image Classification (Classification Task)



(b) Depth Estimation (Regression Task)

Figure 1. Visualization of validation loss, task performance, and the null-space ratio for an output projection layer during LoRA fine-tuning on (a) image classification and (b) depth estimation.

$\Delta\boldsymbol{W}_k^\ell = \boldsymbol{B}_k^\ell \boldsymbol{A}_k^\ell$ with input feature $\boldsymbol{z}$ as

$$\omega_k^\ell(\boldsymbol{z}) = \frac{\left\| \text{Proj}_{\mathcal{N}(\boldsymbol{A}_k^\ell)}(\boldsymbol{z}) \right\|_2}{\|\boldsymbol{z}\|_2}, \tag{3}$$

where $\text{Proj}_{\mathcal{N}(\boldsymbol{A}_k^\ell)}$ denotes the orthogonal projection operator onto the null space of $\boldsymbol{A}_k^\ell$. This ratio measures the extent to which the input feature is discarded by the adapter: a higher $\omega_k^\ell(\boldsymbol{z})$ indicates that a larger portion of the activation lies in the null space, implying that less information is propagated to subsequent layers.

Compression dynamics during fine-tuning. To quantify layer-wise null-space ratio, we compute the expectation $\mathbb{E}_{\boldsymbol{x}}[\omega_k^\ell(\boldsymbol{z}^\ell(\boldsymbol{x}; \boldsymbol{\Theta}))]$ over unlabeled samples, where $\boldsymbol{z}^\ell(\boldsymbol{x}; \boldsymbol{\Theta})$ represents the *incoming* activation to the ℓ-th layer of a model parameterized by $\boldsymbol{\Theta}$ for an input batch $\boldsymbol{x}$. Figure 1 tracks how the null-space ratio evolves during LoRA fine-tuning. For *classification* (Fig. 1a), we fine-tune CLIP [66] with a ViT-B/32 backbone and report validation loss, accuracy, and the layer-wise null-space ratio. For *regression* (Fig. 1b), we fine-tune a ViT-B/16 [16] backbone with a lightweight decoder on depth estimation and report validation loss, RMSE, and the same ratio. In both settings, LoRA adapters of rank 16 are attached to the query, key, value, and output projections in each self-attention block, and training runs for 50 epochs.

As training progresses, the null-space ratio consistently decreases, indicating that a smaller fraction of the activation lies in the adapter's null space. Both models use a latent dimension of 768, while the LoRA rank is 16, so the adapter subspace spans about 2.1 percent of the feature

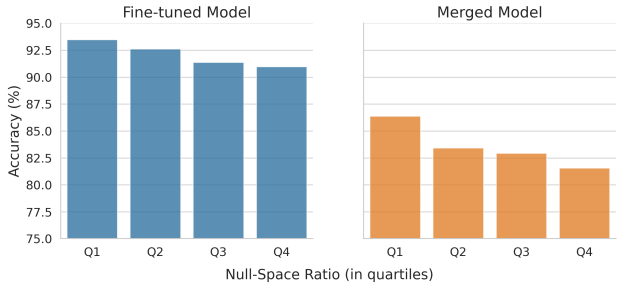


Figure 2. Classification accuracy versus the *null-space ratio*. Accuracy is averaged within quartiles of the ratio across tasks. Left shows fine-tuned experts, right shows the merged model.

space. Given this small coverage, the observed decrease is substantial, showing that more of the representation is being captured within the adapter subspace. Empirically, a lower null-space ratio is associated with higher task performance, exhibiting a strong inverse correlation across datasets and layers. The same pattern holds for both classification and regression despite differences in objectives and data domains. This motivates us to use the ratio as a task-agnostic signal for merging that generalizes across settings. Similar behavior appears in other LoRA layers and transformer blocks (see Sec. C.2 in the supplementary material).

Performance correlation after fine-tuning. The null-space ratio is closely related to model performance after fine-tuning. Figure 2 shows its relationship to classification accuracy. For each task, we partition test samples into four quantiles by their ratio and report the average accuracy within each quantile across tasks. In the *fine-tuned model* (left), samples with lower ratios achieve higher accuracy. A similar pattern holds in the *merged model* (right), where performance remains inversely correlated with the ratio in a merged model with task arithmetic [30]. Even after merging, samples that interact more strongly with their adapters (lower null-space ratios) show the same inverse correlation.

3.3. Null-Space Compression Merging

Motivated by the observed inverse correlation between null-space compression and task performance, we use the null-space ratio as a label-free learning signal to estimate merge coefficients and propose *Null-Space Compression (NSC) Merging*. Unlike applying gradient-based signals when merging full-rank fine-tuned networks, which is memory prohibitive, LoRA merging updates only lightweight adapters, so the extra memory and compute are modest, making the approach feasible in practice.

We compute the mean null-space ratio across all LoRA-equipped layers during inference as

$$\Omega_k(\boldsymbol{x};\boldsymbol{\Theta}) = \frac{1}{|\mathcal{J}|}\sum_{\ell\in\mathcal{J}} \omega_k^\ell\big(\boldsymbol{z}^\ell(\boldsymbol{x};\boldsymbol{\Theta})\big)\,, \tag{4}$$

and estimate its expectation $\mathbb{E}_{\boldsymbol{x}}[\Omega_k(\boldsymbol{x};\boldsymbol{\Theta})]$ over unlabeled samples. Finally, we propose objectives for learning the merging coefficients.

Optimization Objective. We minimize the null-space ratio as a task-agnostic surrogate loss for optimizing model merging coefficients. Precisely, the NSC objective is formulated as

$$\begin{aligned} &\min_{\{\lambda_k^\ell\}} \frac{1}{K}\sum_{k=1}^{K}\left(\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{D}_k}\Big[\Omega_k\big(\boldsymbol{x};\boldsymbol{\Theta}_{\text{merge}}\big)\Big]\right) \\ &\text{s.t.}\quad \boldsymbol{\Theta}_{\text{merge}} = \big\{\boldsymbol{W}_0^\ell + \textstyle\sum_k \lambda_k^\ell \boldsymbol{B}_k^\ell \boldsymbol{A}_k^\ell\big\}_{\ell=1}^{L} \end{aligned} \tag{5}$$

where $\mathcal{D}_k$ denotes an unlabeled validation set for task k.

Fast NSC: Caching the Adapter Gram-Inverse. Before optimizing the merging coefficients in the NSC objective in Eq. (5), we pre-compute the parameters required for evaluating the null-space ratio at each LoRA-equipped layer. For a LoRA update $\Delta\boldsymbol{W}_k = \boldsymbol{B}_k\boldsymbol{A}_k$, the null-space ratio of a feature $\boldsymbol{z}$ can be expressed as

$$\omega_k(\boldsymbol{z}) = \sqrt{1 - \frac{\boldsymbol{z}^\top \boldsymbol{A}_k^\top (\boldsymbol{A}_k\boldsymbol{A}_k^\top)^{-1}\boldsymbol{A}_k\boldsymbol{z}}{\|\boldsymbol{z}\|_2^2}}. \tag{6}$$

This expression is equivalent to Eq. (3). For clarity we write it using the down-projection matrix, and for brevity we omit the layer index ℓ. Since $\boldsymbol{z}$ and $\boldsymbol{A}_k\boldsymbol{z}$ are computed during inference, the only additional term needed is $(\boldsymbol{A}_k\boldsymbol{A}_k^\top)^{-1}$, a small square matrix of dimension equal to the LoRA rank. This avoids explicit construction of the full null-space projection matrix, greatly reducing both memory usage and computational overhead. As a result, the proposed computation enables short preparation time and efficient optimization of the merging coefficients. A detailed derivation of Eq. (6) is in Sec. C.1 of supplementary material, and efficiency is analyzed in Sec. 4.3.

Target Layers for Objective Computation. Computing the null-space ratio at every LoRA-equipped layer during inference incurs computational and memory overhead. Meanwhile, optimization based on the null-space ratio from later transformer blocks and projection layers still influences the merging coefficients of earlier layers. Hence, we analyze effective target layers for computing the NSC objective. Formally, rather than computing Ω_k across the entire set $\mathcal{J}$, restricting computation to a smaller subset $\mathcal{J}_{\text{tgt}} \subseteq \mathcal{J}$ offers a better trade-off between efficiency and performance. In practice, we use only the last quarter of transformer blocks for computing the objective. Experiments supporting this choice are provided in Sec. 4.3.

Advantages over Entropy Minimization. NSC merging extends gradient-based model merging by relying solely on the structural information of LoRA parameters, making it label-free and suitable for heterogeneous tasks where

Algorithm 1 NSC Merging

Require: Pretrained $\{\boldsymbol{W}_0^\ell\}_{\ell=1}^L$; LoRA $\{(\boldsymbol{B}_k^\ell, \boldsymbol{A}_k^\ell)\}$ for k=1:K, unlabeled validation set $\{\mathcal{D}_k\}$; target layers $\mathcal{J}_{\text{tgt}} \subseteq \mathcal{J}$; steps T; batch size b; learning rate η
Ensure: Coefficients $\{\lambda_k^\ell\}$, merged model $\boldsymbol{\Theta}_{\text{merge}}$

Step 1: Prepare adapter Gram-inverse
for $k = 1$ to K **do**
  for $\ell \in \mathcal{J}_{\text{tgt}}$ **do**
    Compute and cache $\left(\boldsymbol{A}_k^\ell \boldsymbol{A}_k^{\ell\top}\right)^{-1}$
  end for
end for
Step 2: Optimize coefficients
Initialize $\lambda_k^\ell \quad \forall\, k, \ell \in \mathcal{J}$
for $t = 1$ to T **do**
  $\boldsymbol{\Theta}_{\text{merge}} \leftarrow \left\{\boldsymbol{W}_0^\ell + \sum_k \lambda_k^\ell \boldsymbol{B}_k^\ell \boldsymbol{A}_k^\ell\right\}_{\ell=1}^L$
  for $k = 1$ to K **do**
    Sample $\{\boldsymbol{x}_i\}_{i=1}^b \sim \mathcal{D}_k$ ▷ Unlabeled mini-batch
    $\widehat{\mathcal{L}}_k \leftarrow \frac{1}{b} \sum_{i=1}^b \Omega_k(\boldsymbol{x}_i; \boldsymbol{\Theta}_{\text{merge}})$
  end for
  $\widehat{\mathcal{L}} \leftarrow \frac{1}{K} \sum_{k=1}^K \widehat{\mathcal{L}}_k$
  $\lambda_k^\ell \leftarrow \lambda_k^\ell - \eta \frac{\partial \widehat{\mathcal{L}}}{\partial \lambda_k^\ell} \quad \forall\, k, \ell \in \mathcal{J}$
end for
return $\{\lambda_k^\ell\}, \boldsymbol{\Theta}_{\text{merge}}$

entropy-based objectives are ill-defined. For LLMs and VLMs performing next-token prediction, entropy objectives must be evaluated at every generated token, causing cost to scale with sequence length and makes the algorithm not scalable. In contrast, NSC is *input*-oriented: it estimates merge weights from initial input prompts or image tokens without relying on *output* logits, keeping its cost independent of generated sequence length. Its ability to adapt using only input IDs is demonstrated in Sec. 4.2.3, and a detailed cost analysis in Sec. 4.3. Overall process of NSC merging is summarized in Algorithm 1.

4. Experiments

4.1. Experimental Setup

Datasets and Tasks. We evaluate across three settings: a heterogeneous vision setting with ViT-based multi-task learning, an LLM sequence classification setting, and a VLM vision–language setting. For vision, we use three multi-task datasets spanning indoor and outdoor scenes. NYUD-v2 [68] provides four tasks: depth estimation, semantic segmentation, surface normal prediction, and edge detection. PASCAL-Context [60] includes five tasks: semantic segmentation, human parts estimation, saliency estimation, surface normal prediction, and edge detection. Taskonomy [100] covers eleven tasks: Depth Euclidean (DE), Depth Z-buffer (DZ), Edge Texture (ET), Keypoints 2D (K2), Keypoints 3D (K3), Normal (N), Principal Curvature (C), Reshading (R), Segment Unsup2D (S2), and Segment Unsup2.5D (S2.5).

For large language models (LLMs), we use LLaMA-3-8B [18] on six natural language inference (NLI) datasets: MNLI, QNLI, SNLI, RTE, SICK, and SciTail. For vision-language models (VLMs), we employ LLaVA-1.5-7B [46, 47] across a diverse set of multimodal tasks, ranging from word-level visual question answering to sentence-level generation. Specifically, we evaluate on VizWiz, IconQA, ChartQA, DocVQA, COCO, and Flickr30k. Further details on the datasets and data splits are provided in supple Sec. A.1.

Baselines. For simplicity, we omit layer-wise index ℓ for this section. Let $\boldsymbol{W}_0$ denote the pretrained parameters, $\Delta\boldsymbol{W}_k$ the task vector for task k, and $\boldsymbol{W}_{\text{merge}}$ the merged parameters. For LoRA adapters we write $\Delta\boldsymbol{W}_k = \boldsymbol{B}_k\boldsymbol{A}_k$, and λ denotes a global scaling coefficient. We evaluate two families of methods. *Vanilla merging* operates in full parameter space. (i) **Task Arithmetic (TA)** [30] adds a scaled sum of task vectors to the pretrained weights, i.e., $\boldsymbol{W}_{\text{merge}} = \boldsymbol{W}_0 + \lambda \sum_k \Delta\boldsymbol{W}_k$. (ii) **TIES** [90] prunes low-magnitude coordinates, enforces sign consensus across models, then averages only the retained coordinates with scale λ. (iii) **DARE-TIES** [98] applies Bernoulli sparsification with probability p and rescales by $1/(1-p)$ to preserve the expectation. (iv) **AdaMerging** [91] learns merging coefficients by minimizing an output-entropy surrogate in the spirit of test-time adaptation [79]. One line of works exploit the low-rank structure of adapters. (v) **SVD** [76] aggregates LoRA task vectors $\Delta\boldsymbol{W}_{\text{merge}} = \lambda \sum_k \boldsymbol{B}_k\boldsymbol{A}_k$, applies truncated SVD to the sum, and refactors the result back into LoRA form to recover the desired rank. (vi) **Linear** [54] performs TA directly on the factors by linearly combining $\{\boldsymbol{A}_k\}$ and $\{\boldsymbol{B}_k\}$ instead of whole task vectors. (vii) **KnOTS** [71] computes an SVD over the concatenation of task vectors and merges by assigning the right-singular components to tasks; applying TIES or DARE-TIES to these components yields **KnOTS-TIES** and **KnOTS-DARE-TIES**. (viii) **LoRA-LEGO** [106] decomposes adapters into semantic units, clusters them rank-wise, and assembles a new adapter from cluster centroids. Additionally, we evaluate (ix) **EMR-Merging** [29], (x) **FR-Merging** [107], and (xi) **RobustMerge** [101]. Implementation details for all baselines appear in Sec. A.3 of supplementary material.

Evaluation Metrics. We report *normalized performance* relative to a finetuned reference model. For higher-is-better metrics, a method's score is divided by the finetuned model's score. For lower-is-better metrics, the finetuned score is divided by the method's score. Each task uses its conventional metric from the literature. See supple A.2 for detailed information.

Table 1. Merging results on NYUD-v2, PASCAL-Context, and Taskonomy using ViT-B. The fine-tuned rows report per-task absolute performance. For each merging baseline, we report the normalized performance (%) relative to the corresponding fine-tuned score. The rightmost *Avg* column is the mean of the normalized scores across tasks.

Method	NYUD-v2 (4)				PASCAL-Context (5)					Taskonomy (11)											Avg
	Depth RMSE ↓	Semseg mIoU ↑	Normal Mean ↓	Edge L1 ↑	Semseg mIoU ↑	Parts mIoU ↑	Saliency mIoU ↑	Normal Mean ↓	Edge L1 ↑	DE L1 ↓	DZ L1 ↓	EO L1 ↓	ET L1 ↓	K2 L1 ↓	K3 L1 ↓	N L1 ↓	C RMSE ↓	R L1 ↓	S2 L1 ↓	S2.5 L1 ↓	
Finetuned performance																					
Finetuned	0.657	37.66	25.98	0.051	70.07	54.77	80.00	18.36	0.046	0.016	0.016	0.101	0.171	0.162	0.082	0.217	0.710	0.136	0.170	0.144	–
Merged models, normalized to finetuned (%)																					
TA [30]	24.0	1.3	54.9	105.2	4.7	20.6	36.4	62.5	104.3	100.1	99.8	101.7	104.6	106.4	102.2	100.2	103.6	106.2	107.4	98.8	77.2
TIES [90]	28.5	1.3	55.2	105.1	4.7	20.6	34.6	66.1	104.3	102.5	102.0	100.6	103.7	105.4	99.5	100.1	102.4	105.2	104.3	99.5	77.3
DARE-TIES [98]	24.3	1.3	54.6	105.2	4.7	20.6	32.6	62.2	104.3	98.1	97.7	103.3	104.6	106.4	104.5	100.1	103.7	103.9	104.6	100.6	76.9
SVD [76]	6.8	1.2	40.6	105.4	4.8	20.6	31.2	49.4	104.3	98.0	94.4	103.7	123.4	146.9	117.8	101.5	122.2	143.4	146.7	96.9	83.0
Linear [54]	2.6	0.5	30.7	105.2	0.1	20.5	28.5	21.4	102.2	108.3	105.9	95.9	120.5	122.1	117.1	100.4	98.1	126.4	75.0	104.9	74.3
KnOTS-TIES [71]	19.4	1.5	54.5	105.2	4.7	20.6	38.3	61.2	104.3	100.1	99.6	101.3	102.8	105.2	102.4	100.1	102.7	104.9	104.8	97.9	76.6
KnOTS-DARE-TIES [71]	18.2	1.5	55.4	105.2	4.7	20.6	42.5	59.9	104.3	103.9	102.8	97.7	107.8	107.3	96.1	100.2	102.4	103.1	101.7	100.2	76.8
LoRA-LEGO [106]	9.3	1.4	52.8	104.3	4.7	20.6	36.3	55.4	104.0	103.9	104.1	96.7	106.7	108.7	99.1	100.2	103.6	110.4	102.2	104.5	76.4
EMR-Merging [29]	21.6	1.5	55.3	103.9	4.7	20.6	44.9	58.9	104.0	98.2	98.2	102.4	103.0	104.8	103.2	100.1	103.4	105.4	108.1	98.5	77.0
FR-Merging [107]	6.3	1.2	37.9	105.4	4.8	20.6	31.0	48.2	104.3	99.7	95.6	101.5	125.8	150.6	117.0	101.7	122.7	146.6	149.7	97.3	83.4
RobustMerge [101]	37.2	76.5	63.4	100.1	83.1	73.7	85.7	73.7	100.6	100.0	101.0	102.2	100.6	100.5	100.6	100.0	100.2	99.7	99.1	99.8	89.9
NSC (Ours)	45.9	85.1	69.6	100.4	86.7	76.7	88.7	80.9	101.7	100.1	101.0	102.2	100.8	100.8	100.7	100.0	100.4	100.0	99.6	99.6	**92.0**

Table 2. Per-task results on six NLI benchmarks. We merge six LLAMA-3 8B checkpoints, each fine-tuned with LoRA (rank 16). The top panel reports absolute accuracies of the fine-tuned models. The bottom panel reports accuracies of the merged models, normalized to their corresponding fine-tuned baselines (%).

Method	MNLI	QNLI	SNLI	RTE	SICK	SciTail	Avg
Per-task absolute accuracy (%)							
Finetuned	90.8	94.9	91.8	87.0	90.9	94.8	91.7
Merged models (normalized to finetuned baselines, %)							
TA [30]	92.8	86.8	93.3	93.6	83.8	95.0	90.9
TIES [90]	94.3	88.8	90.8	89.8	86.6	94.4	90.8
DARE-TIES [98]	94.3	88.8	90.8	89.8	86.6	94.4	90.8
SVD [76]	95.4	88.6	92.8	93.7	80.0	92.4	90.5
Linear [54]	96.0	86.1	88.1	94.4	83.8	96.1	90.8
KnOTS-TIES [71]	92.0	82.0	94.9	92.1	80.2	95.3	89.4
KnOTS-DARE-TIES [71]	91.8	82.3	94.9	91.3	80.1	95.4	89.3
LoRA-LEGO [106]	87.7	85.1	90.9	92.1	86.1	95.6	89.6
EMR-Merging [29]	96.2	88.0	94.2	92.9	76.6	94.5	90.4
FR-Merging [107]	95.5	88.7	93.1	93.6	80.3	92.6	90.6
RobustMerge [101]	94.3	88.1	93.7	93.6	83.0	94.5	91.2
AdaMerging [91]	94.3	84.8	92.5	92.1	89.2	84.8	89.6
NSC (Ours)	94.9	88.3	92.8	91.3	91.2	95.1	**92.3**

Implementation Details. We use AdamW [51] with a learning rate of 0.001. The merging weights λ are initialized to 0.4. For heterogeneous vision tasks, we optimize merging weights λ for 100 iterations with a batch size of 32. For LLM and VLM experiments, we use a batch size of 2 and 1, respectively, for 500 iterations with a learning rate of 0.0003. We compare against ten baselines and follow the experimental protocols reported in their original papers. For TA [30], TIES [90], and KnOTS [71], we set a global merge scale λ and tune it on a small validation split using validation loss. For AdaMerging [91], all merging coefficients $\{\lambda_k^\ell\}_{k=1,\ell=1}^{K,L}$ are learned with the entropy surrogate, where N is the number of tasks and L is the number of layers. Further details about the algorithm and fine-tuning details are provided in Sec. A.4 of supplementary material.

4.2. Experimental Results

4.2.1. Results across 20 heterogeneous vision tasks

To test that our method works well on both classification and regression, we test on 20 tasks from NYUD-v2 (4), PASCAL-Context (5), and Taskonomy (11), encompassing diverse dense predictions like semantic segmentation, depth, and normals. Table 1 reports normalized per-task scores for a ViT-B backbone. Gradient-free methods collapse on dense prediction in NYUD-v2 and PASCAL-Context. For example, TA, TIES, and DARE-TIES achieve only about 1–2% on NYUD-v2 semantic segmentation, whereas NSC reaches 85.1%. On PASCAL-Context, vanilla methods stay near 20–21% on Parts and about 33–38% on Saliency, while NSC attains 76.7% and 88.7% respectively. SVD and Linear improve several Taskonomy metrics but suffer pronounced drops on others (e.g., NYUD-v2 Depth), lowering their overall averages. This indicates that prior gradient-free approaches have clear limitations when merging models for heterogeneous tasks. NSC provides the most balanced outcome. It stays near parity on all eleven Taskonomy tasks (about 99.6–102.2%) and markedly outperforms alternatives on NYUD-v2 and PASCAL-Context. This balance yields the best average normalized score at 92.0% across twenty tasks, indicating strong retention of task-specific performance in heterogeneous settings.

4.2.2. Per-task evaluation across six NLI tasks

To assess the scalability of merging methods, we report per-task normalized accuracies on six NLI benchmarks. In Tab. 2, we fine-tune LLAMA-3 8B with LoRA adapters of rank 16 applied to the query and value projections. Gradient-free baselines show limited ability to preserve task-specific information at this scale. AdaMerging uses a pseudo-entropy surrogate by treating token-level probabilities as a label-free signal, yet it underperforms both

Table 3. Per-task performance of merged LLaVA-1.5-7B models across six multi-modal benchmarks. Each model fine-tunes the Vicuna-7B language backbone with LoRA [28] (rank = 16), while fully fine-tuning the multi-modal projector. The upper block reports absolute accuracies of fine-tuned baselines; the lower block presents merged models' results normalized to their corresponding baselines (%).

Method	IconQA	VizWiz$_{val}$	ChartQA	DocVQA$_{val}$	COCO	Flickr30k	Avg
Per-task absolute score							
Zero-Shot	17.9	55.2	18.2	24.3	109.5	79.2	–
Finetuned	67.8	69.3	39.0	40.7	130.5	91.3	–
Avg. generated tokens	2.1	2.8	4.7	6.7	12.5	14.2	7.17
Merged models (normalized to fine-tuned baselines, %)							
TA [30]	56.8	83.2	73.0	85.2	92.9	97.6	81.4
TIES [90]	61.2	81.6	76.1	82.3	92.9	96.7	81.8
DARE-TIES [98]	57.4	84.4	72.8	84.4	92.5	95.8	81.2
SVD [76]	56.2	82.7	72.9	85.5	92.6	96.6	81.1
Linear [54]	52.6	89.8	70.0	79.2	87.0	95.9	79.1
KnOTS-TIES [71]	49.8	84.1	70.4	85.5	92.3	97.2	79.9
KnOTS-DARE-TIES [71]	52.4	86.1	72.6	85.3	93.0	96.3	80.9
LoRA-LEGO [106]	53.1	86.5	68.4	82.5	91.5	96.6	79.8
EMR-Merging [29]	84.0	49.2	65.6	84.3	92.0	97.1	78.7
FR-Merging [107]	87.2	46.3	61.5	76.2	90.2	94.8	76.0
RobustMerge [101]	82.3	58.2	71.1	82.1	93.6	96.9	80.7
AdaMerging [91] (Single Token)	64.7	76.0	76.4	82.1	87.5	98.4	80.9
AdaMerging [91] (Full Token)	68.7	76.2	77.9	85.8	91.1	94.4	82.4
NSC (Ours)	59.7	82.9	78.1	87.1	91.7	96.8	**82.7**

gradient-free baselines and our method, which indicates that entropy-based objectives are not uniformly effective for LLMs. In contrast, NSC achieves consistent gains across all tasks and attains the best average normalized accuracy of 92.3%, demonstrating the effectiveness of our approach on language benchmarks.

4.2.3. Per-task evaluation across six VLM tasks

We evaluate the scalability of NSC merging for sequence generation on six vision–language benchmarks in Tab. 3. VLMs such as LLaVA [46, 47] decode outputs token by token. Classification prompts often terminate after a single token, whereas captioning tasks need long sequences. AdaMerging must compute a entropy loss at each generated token, so cost grows with sequence length. To reflect this, we consider two AdaMerging settings as baselines: *Single Token*, which updates with the entropy of only the first generated token, and *Full Token*, which updates with sequence-level entropy over all generated tokens. NSC leverages the LoRA projection matrix's null-space ratio, a structural signal independent of token logits, so even a single generated token provides a sufficient gradient signal. Its computational cost therefore matches the *Single Token* setting. NSC consistently outperforms AdaMerging (*Single Token*) and matches or slightly exceeds *Full Token* across tasks. Given the reported sequence lengths, *Full Token* requires on average about seven times more gradient updates than NSC to reach similar accuracy. A detailed analysis of computational cost is provided in Sec. 4.3.

4.2.4. Generalization to unseen tasks

We evaluate generalization in a seen–unseen protocol inspired by AdaMerging [91]. Prior work considered eight classification tasks. We expand the setting to twenty heterogeneous vision tasks that include regression. We select ten seen tasks and ten unseen tasks distributed across NYUD-v2, PASCAL-Context, and Taskonomy, and report per-task scores along with averages over seen, unseen, and all tasks.

Table 4 summarizes the results. Gradient-free methods struggle to preserve performance on unseen objectives and dense prediction. Their averages remain modest on both splits (e.g., TA 82.4% seen and 76.8% unseen, TIES 80.5% seen and 76.3% unseen). LoRA-aware methods can post strong performances on subsets of Taskonomy but remain unstable when moving to NYUD-v2 and PASCAL-Context. Linear shows pronounced volatility and the lowest unseen average among LoRA-aware baselines (68.9%), and SVD improves some Taskonomy tasks yet drops on NYUD-v2 Depth and PASCAL-Context Normal, which pulls down its averages (78.1% seen and 75.1% unseen). NSC delivers consistently balanced gains. It achieves the best averages on both splits, with 95.1% on seen tasks and 87.1% on unseen tasks, and the highest overall average of 91.1%.

4.3. Ablation Study

Robustness to target module and block count. To ensure the NSC objective Eq. (5) influences all LoRA adapters, it is intuitive to place it toward the back of the network so that gradients flow through every upstream adapter at least once. Tab. 5 ablates the attention module targeted by NSC (Q, K, V, O and their subsets) and the number of activated blocks. Targeting the output projection O gives a slight edge across depths, but the spread across QKVO, KVO, and VO is small, indicating that NSC is robust to the module choice. Under a coverage constraint that always includes the last portion with a LoRA adapter, activating a small set of mid to late blocks is sufficient. Performance peaks at

Table 4. Experiments on generalization to unseen tasks, including additional baselines. Merging results on NYUD-v2, PASCAL-Context, and Taskonomy with a ViT-B backbone. Fine-tuned rows report per-task absolute performance. For each baseline, we report performance normalized (%) to the corresponding fine-tuned score, and we also report averages over seen tasks, unseen tasks, and all tasks.

Method	Seen										Avg (Seen)	Unseen										Avg (Unseen)	Avg
	NYUD-v2		PASCAL-Context		Taskonomy							NYUD-v2		PASCAL-Context			Taskonomy						
	Semseg mIoU ↑	Edge L1 ↑	Parts mIoU ↑	Sal mIoU ↑	DZ L1 ↓	ET L1 ↓	K3 L1 ↓	N L1 ↓	R L1 ↓	S2.5 L1 ↓		Depth RMSE ↓	Normal Mean ↓	Semseg mIoU ↑	Normal Mean ↓	Edge L1 ↑	DE L1 ↓	EO L1 ↓	K2 L1 ↓	C RMSE ↓	S2 L1 ↓		
Finetuned performance (absolute)																							
Finetuned	37.66	0.051	54.77	80.00	0.0160	0.1713	0.0820	0.2169	0.1357	0.1435	–	0.657	25.98	70.07	18.36	0.046	0.016	0.101	0.162	0.710	0.170	–	–
Merged models, normalized to finetuned (%)																							
TA [30]	1.3	102.0	20.6	30.8	91.9	120.9	115.0	100.8	134.7	106.1	82.4	6.6	43.5	4.7	47.0	104.3	95.1	103.7	132.3	118.6	111.7	76.8	79.6
TIES [90]	1.3	104.2	20.6	28.7	99.6	111.4	105.2	100.6	127.9	105.7	80.5	9.6	49.0	4.7	51.4	104.3	103.5	98.9	123.2	113.2	105.4	76.3	78.4
DARE-TIES [98]	1.2	105.3	20.6	33.6	89.4	111.4	118.3	100.5	117.9	100.9	79.9	7.3	49.8	4.7	53.5	104.3	88.1	106.4	128.6	122.0	105.9	77.1	78.5
SVD [76]	1.4	105.1	20.6	32.6	96.8	107.2	106.2	100.2	110.4	100.0	78.1	14.3	49.7	4.7	52.9	104.3	98.2	103.3	110.8	107.1	105.3	75.1	76.6
Linear [54]	0.4	73.4	20.3	29.0	117.7	226.8	95.6	99.6	124.3	25.9	81.3	1.5	29.1	0.1	23.5	102.0	130.3	87.9	190.4	95.6	28.9	68.9	75.1
KnOTS-TIES [71]	1.3	80.3	20.6	28.5	96.9	119.9	108.7	100.4	116.0	99.1	77.2	5.2	43.5	4.7	44.5	104.3	99.5	104.3	121.7	110.0	105.7	74.3	75.8
KnOTS-DARE-TIES [71]	1.2	104.2	20.6	28.5	101.6	115.9	118.3	100.4	103.0	85.8	77.9	4.8	47.9	4.7	48.2	104.3	97.5	105.5	117.8	106.2	122.8	76.0	77.0
LoRA-LEGO [106]	1.2	104.7	20.6	33.5	97.2	102.1	105.5	100.2	109.7	98.5	77.3	9.0	48.9	4.7	53.0	104.3	98.8	102.2	106.9	104.8	102.8	73.5	75.4
EMR-Merging [29]	1.3	88.3	20.6	36.4	94.3	104.4	106.9	100.2	108.9	98.6	76.0	15.9	53.1	4.7	56.9	104.1	95.2	103.5	109.2	107.3	108.2	75.8	75.9
FR-Merging [107]	1.3	104.0	20.6	31.4	92.9	121.0	113.0	100.8	134.9	106.2	82.6	6.8	45.1	4.7	47.3	104.3	96.3	101.4	130.5	117.2	110.5	76.4	79.5
RobustMerge [101]	76.5	100.1	73.6	85.7	101.0	100.6	100.6	100.0	99.7	99.8	93.8	37.2	63.4	83.1	73.7	100.6	100.0	102.2	100.5	100.2	99.1	86.0	89.9
NSC (Ours)	84.6	100.3	76.8	87.3	100.9	100.6	100.8	100.0	99.7	99.7	**95.1**	40.8	65.0	85.4	76.0	101.2	100.0	102.3	100.6	100.3	99.3	**87.1**	**91.1**

Table 5. Ablation on where the NSC objective is applied: target module and number of blocks. Evaluated on CLIP (ViT-B/32) across eight image classification tasks.

Targeted Projection Matrix	Number of Activated Transformer Blocks			
	12	6	3	1
QKVO	83.7	84.4	83.9	83.0
KVO	84.0	84.6	84.0	83.0
VO	84.3	84.7	84.1	82.6
O	84.5	84.9	84.6	82.6

84.9 with O on six blocks and remains strong at 84.6 with three blocks, which we adopt as the default for a better accuracy–compute trade off. Using only one block degrades accuracy, while activating all twelve is unnecessary. In practice, applying NSC to O on a few back half blocks is both effective and efficient.

Computational cost and memory. In Tab. 6, we compare wall-clock time and peak memory of NSC and baselines on LLAVA across six tasks using 500 validation samples. We include gradient-free methods (TA, TIES, KnOTS–TIES) and the gradient-based AdaMerging. Gradient-free methods merge parameters and tune scales on validation splits, with TIES and KnOTS–TIES add preprocessing for sign-conflict or SVD of task vectors. In contrast, both NSC merging and AdaMerging optimizes merge coefficients via a label-free objective without repeated evaluations. While gradient-free methods consume less GPU memory, they require longer validation due to multiple next-token predictions. Gradient-based approaches use slightly more memory, but with LoRA adapters the difference is minor compared to full fine-tuned model merging, keeping gradient-based optimization practical. NSC matches AdaMerging (Single Token) in cost, with only negligible overhead from projection formation in Eq. (6). Applying entropy minimization to all tokens causes AdaMerging's optimization time to scale with sequence length. Overall, NSC achieves a favorable performance–efficiency balance and surpasses prior baselines, as shown in Tab. 3.

Table 6. Runtime and memory efficiency. Comparison of preparation, optimization, and validation time (minutes) and memory usage across gradient-free and gradient-based methods. Validation time *excludes* grid-search overhead for gradient-free methods.

Method	Requirements			Prep. (min)	Opt. (min)	Val. (min)	*Total* (min)	*GPU Mem.* (GB)
	Prep.	Opt.	Val.					
Gradient-Free								
TA [30]	✗	✗	✓	–	–	16.6	16.6	**14.4**
TIES [90]	✓	✗	✓	4.5	–	16.6	20.1	**14.4**
KnOTS-TIES [71]	✓	✗	✓	6.8	–	16.6	23.4	**14.4**
Gradient-Based								
AdaMerging [91] (Single Token)	✗	✓	✗	–	13.3	–	**13.3**	18.0
AdaMerging (Full Token)	✗	✓	✗	–	104.0	–	104.0	18.0
NSC (Ours)	✓	✓	✗	≈0.0	13.3	–	**13.3**	17.9

Additional analyses including computational cost and memory are presented in Sec. D of the supplement.

5. Conclusion

We studied LoRA merging, where multiple adapters are combined to build a single model. During LoRA fine-tuning, the down-projection factor $\boldsymbol{A}$ in $\Delta \boldsymbol{W} = \boldsymbol{B}\boldsymbol{A}$ defines an adapter subspace. We define the proportion of the activation that is suppressed by this projection as the null-space ratio. Empirically, this ratio decreases as task performance improves, providing a simple label-free signal for learning merge weights. Building on this insight, we introduce Null-Space Compression (NSC) Merging, a computationally efficient approach for combining multiple LoRA adapters. Across twenty heterogeneous vision tasks, six NLI tasks, and six vision–language tasks, NSC achieves balanced, state-of-the-art performance.

Acknowledgements This research was supported by the Challengeable Future Defense Technology Research and Development Program through the Agency For Defense Development(ADD) funded by the Defense Acquisition Program Administration(DAPA) in 2026(No.915102201)

References

[1] Samuel K Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries. *arXiv preprint arXiv:2209.04836*, 2022. 2

[2] Roy Bar-Haim, Ido Dagan, Bill Dolan, L. Ferro, Danilo Giampiccolo, B. Magnini, and Idan Szpektor. The Second PASCAL Recognising Textual Entailment Challenge . 2006. 1

[3] Luisa Bentivogli, Ido Dagan, Hoa Trang Dang, Danilo Giampiccolo, and Bernardo Magnini. The Fifth PASCAL Recognizing Textual Entailment Challenge. 1

[4] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 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, 2015. Association for Computational Linguistics. 1

[5] Shiqi Chen, Jinghan Zhang, Tongyao Zhu, Wei Liu, Siyang Gao, Miao Xiong, Manling Li, and Junxian He. Bring reason to vision: Understanding perception and reasoning through model merging. In *Forty-second International Conference on Machine Learning*, 2025. 2

[6] Weiyu Chen and James Kwok. Pareto merging: Multi-objective optimization for preference-aware model merging. *arXiv preprint arXiv:2408.12105*, 2024. 2

[7] Zitian Chen, Yikang Shen, Mingyu Ding, Zhenfang Chen, Hengshuang Zhao, Erik G Learned-Miller, and Chuang Gan. Mod-squad: Designing mixtures of experts as modular multi-task learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11828–11837, 2023. 2

[8] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote Sensing Image Scene Classification: Benchmark and State of the Art. *Proceedings of the IEEE*, 105(10):1865–1883, 2017. 7

[9] 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. *arXiv preprint arXiv:2503.08099*, 2025. 2

[10] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing Textures in the Wild. pages 3606–3613, 2014. 7

[11] Ido Dagan, Oren Glickman, and Bernardo Magnini. The PASCAL Recognising Textual Entailment Challenge. In *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*, pages 177–190, Berlin, Heidelberg, 2006. Springer. 1

[12] Nico Daheim, Thomas Möllenhoff, Edoardo Maria Ponti, Iryna Gurevych, and Mohammad Emtiyaz Khan. Model merging by uncertainty-based gradient matching. *arXiv preprint arXiv:2310.12808*, 2023. 2

[13] Hoa T. Dang, Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, Elena Cabrio, and Bill Dolan. The Fourth PASCAL Recognizing Textual Entailment Challenge . *NIST*, 2009. Last Modified: 2017-02-19T20:02-05:00 Publisher: Hoa T. Dang, Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, Elena Cabrio, Bill Dolan. 1

[14] MohammadReza Davari and Eugene Belilovsky. Model breadcrumbs: Scaling multi-task model merging with sparse masks. In *European Conference on Computer Vision*, pages 270–287. Springer, 2024. 2

[15] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 2

[16] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 1, 3

[17] Yiyang Du, Xiaochen Wang, Chi Chen, Jiabo Ye, Yiru Wang, Peng Li, Ming Yan, Ji Zhang, Fei Huang, Zhifang Sui, et al. Adamms: Model merging for heterogeneous multimodal large language models with unsupervised coefficient optimization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 9413–9422, 2025. 2

[18] Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024. 1, 2, 3, 5

[19] Sebastian Dziadzio, Vishaal Udandarao, Karsten Roth, Ameya Prabhu, Zeynep Akata, Samuel Albanie, and Matthias Bethge. How to merge your multimodal models over time? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 20479–20491, 2025. 3

[20] Rahim Entezari, Hanie Sedghi, Olga Saukh, and Behnam Neyshabur. The role of permutation invariance in linear mode connectivity of neural networks. *arXiv preprint arXiv:2110.06296*, 2021. 2

[21] Chris Fifty, Ehsan Amid, Zhe Zhao, Tianhe Yu, Rohan Anil, and Chelsea Finn. Efficiently identifying task groupings for multi-task learning. *Advances in Neural Information Processing Systems*, 34:27503–27516, 2021. 5

[22] Rohit Gandikota, Joanna Materzyńska, Tingrui Zhou, Antonio Torralba, and David Bau. Concept sliders: Lora adaptors for precise control in diffusion models. In *European Conference on Computer Vision*, pages 172–188. Springer, 2024. 2

[23] 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 *Proceedings of the Com-*

puter Vision and Pattern Recognition Conference, pages 18695–18705, 2025. 2

[24] Danilo Giampiccolo, Bernardo Magnini, Ido Dagan, and Bill Dolan. The Third PASCAL Recognizing Textual Entailment Challenge. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 1–9, Prague, 2007. Association for Computational Linguistics. 1

[25] Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. Vizwiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3608–3617, 2018. 1

[26] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. EuroSAT: A Novel Dataset and Deep Learning Benchmark for Land Use and Land Cover Classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019. 7

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

[28] 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. *ICLR*, 1(2):3, 2022. 1, 2, 3, 7

[29] Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. Emr-merging: Tuning-free high-performance model merging. *Advances in Neural Information Processing Systems*, 37:122741–122769, 2024. 5, 6, 7, 8, 2, 3

[30] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022. 1, 2, 3, 4, 5, 6, 7, 8

[31] Wooseong Jeong and Kuk-Jin Yoon. Quantifying task priority for multi-task optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 363–372, 2024. 1

[32] Wooseong Jeong and Kuk-Jin Yoon. Resolving token-space gradient conflicts: Token space manipulation for transformer-based multi-task learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2887–2897, 2025.

[33] Wooseong Jeong and Kuk-Jin Yoon. Selective task group updates for multi-task optimization. In *The Thirteenth International Conference on Learning Representations*, 2025.

[34] Wooseong Jeong, Jegyeong Cho, Youngho Yoon, and Kuk-Jin Yoon. Synchronizing task behavior: Aligning multiple tasks during test-time training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 24340–24350, 2025.

[35] Wooseong Jeong, Jegyeong Cho, Youngho Yoon, Jaeyoung Lee, and Kuk-Jin Yoon. Stabilizing multi-task latent spaces: Recursive refinement with coordinators in partially labeled learning. *IEEE Access*, 2026. 1

[36] Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. *arXiv preprint arXiv:2212.09849*, 2022. 1, 2

[37] Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[38] Tushar Khot, Ashish Sabharwal, and Peter Clark. SciTaiL: A Textual Entailment Dataset from Science Question Answering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2018. 1

[39] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 2

[40] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013. 7

[41] Yann LeCun and Corinna Cortes. The mnist database of handwritten digits. 2005. 7

[42] Giwon Lee, Wooseong Jeong, Daehee Park, Jaewoo Jeong, and Kuk-Jin Yoon. Interaction-merged motion planning: Effectively leveraging diverse motion datasets for robust planning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 28610–28621, 2025. 3

[43] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014. 2

[44] Bo Liu, Xingchao Liu, Xiaojie Jin, Peter Stone, and Qiang Liu. Conflict-averse gradient descent for multi-task learning. *Advances in Neural Information Processing Systems*, 34:18878–18890, 2021. 1

[45] Bo Liu, Yihao Feng, Peter Stone, and Qiang Liu. Famo: Fast adaptive multitask optimization. *Advances in Neural Information Processing Systems*, 36:57226–57243, 2023. 1

[46] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916, 2023. 1, 3, 5, 7, 2

[47] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 26296–26306, 2024. 1, 5, 7, 2

[48] Shuqi Liu, Han Wu, Bowei He, Xiongwei Han, Mingxuan Yuan, and Linqi Song. Sens-merging: Sensitivity-guided parameter balancing for merging large language models. *arXiv preprint arXiv:2502.12420*, 2025. 3

[49] Yun Liu, Peng-Tao Jiang, Vahan Petrosyan, Shi-Jie Li, Jiawang Bian, Le Zhang 0001, and Ming-Ming Cheng. Del: Deep embedding learning for efficient image segmentation. In *IJCAI*, page 870, 2018. 1

[50] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017. 2

[51] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 6, 2

[52] Pan Lu, Liang Qiu, Jiaqi Chen, Tony Xia, Yizhou Zhao, Wei Zhang, Zhou Yu, Xiaodan Liang, and Song-Chun Zhu. Iconqa: A new benchmark for abstract diagram understanding and visual language reasoning. In *The 35th Conference on Neural Information Processing Systems (NeurIPS) Track on Datasets and Benchmarks*, 2021. 1

[53] Zhenyi Lu, Chenghao Fan, Wei Wei, Xiaoye Qu, Dangyang Chen, and Yu Cheng. Twin-merging: Dynamic integration of modular expertise in model merging. *Advances in Neural Information Processing Systems*, 37:78905–78935, 2024. 2

[54] Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. PEFT: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`, 2022. 1, 5, 6, 7, 8

[55] 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. *arXiv preprint arXiv:2502.04959*, 2025. 2

[56] Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 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, 2014. European Language Resources Association (ELRA). 1

[57] Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. Chartqa: A benchmark for question answering about charts with visual and logical reasoning. *arXiv preprint arXiv:2203.10244*, 2022. 1

[58] Michael S Matena and Colin A Raffel. Merging models with fisher-weighted averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716, 2022. 2

[59] Minesh Mathew, Dimosthenis Karatzas, and CV Jawahar. Docvqa: A dataset for vqa on document images. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 2200–2209, 2021. 1

[60] Roozbeh Mottaghi, Xianjie Chen, Xiaobai Liu, Nam-Gyu Cho, Seong-Whan Lee, Sanja Fidler, Raquel Urtasun, and Alan Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 891–898, 2014. 5, 1

[61] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading Digits in Natural Images with Unsupervised Feature Learning. 7

[62] Amin Heyrani Nobari, Kaveh Alimohammadi, Ali ArjomandBigdeli, Akash Srivastava, Faez Ahmed, and Navid Azizan. Activation-informed merging of large language models. *arXiv preprint arXiv:2502.02421*, 2025. 3

[63] 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. *arXiv preprint arXiv:2509.17786*, 2025. 2

[64] Akshara Prabhakar, Yuanzhi Li, Karthik Narasimhan, Sham Kakade, Eran Malach, and Samy Jelassi. Lora soups: Merging loras for practical skill composition tasks. *arXiv preprint arXiv:2410.13025*, 2024. 2

[65] Zihuan Qiu, Lei Wang, Yang Cao, Runtong Zhang, Bing Su, Yi Xu, Fanman Meng, Linfeng Xu, Qingbo Wu, and Hongliang Li. Null-space filtering for data-free continual model merging: Preserving transparency, promoting fidelity. *arXiv preprint arXiv:2509.21413*, 2025. 3

[66] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021. 3, 7

[67] Li Shen, Anke Tang, Enneng Yang, Guibing Guo, Yong Luo, Lefei Zhang, Xiaochun Cao, Bo Du, and Dacheng Tao. Efficient and effective weight-ensembling mixture of experts for multi-task model merging. *arXiv preprint arXiv:2410.21804*, 2024. 2

[68] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012. 5, 1

[69] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In *The 2011 International Joint Conference on Neural Networks*, pages 1453–1460, 2011. ISSN: 2161-4407. 7

[70] 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. 2

[71] George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with svd to tie the knots. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025. 1, 2, 5, 6, 7, 8

[72] Derek Tam, Mohit Bansal, and Colin Raffel. Merging by matching models in task parameter subspaces. *arXiv preprint arXiv:2312.04339*, 2023. 2

[73] 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. *arXiv preprint arXiv:2310.04742*, 2023. 3

[74] Anke Tang, Li Shen, Yong Luo, Nan Yin, Lefei Zhang, and Dacheng Tao. Merging Multi-Task Models via Weight-Ensembling Mixture of Experts. In *Proceedings of the 41st International Conference on Machine Learning*, pages 47778–47799. PMLR, 2024. ISSN: 2640-3498. 2

[75] Anke Tang, Enneng Yang, Li Shen, Yong Luo, Han Hu, Bo Du, and Dacheng Tao. Merging models on the fly without retraining: A sequential approach to scalable continual model merging. *arXiv preprint arXiv:2501.09522*, 2025. 3

[76] Dennis Tang, Prateek Yadav, Yi-Lin Sung, Jaehong Yoon, and Mohit Bansal. Lora merging with svd: Understanding interference and preserving performance. In *ICML 2025 Workshop on Reliable and Responsible Foundation Models*, 2025. 2, 5, 6, 7, 8

[77] Jianqiang Wan, Yang Liu, Donglai Wei, Xiang Bai, and Yongchao Xu. Super-bpd: Super boundary-to-pixel direction for fast image segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9253–9262, 2020. 1

[78] Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. 2018. 1

[79] Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully Test-Time Adaptation by Entropy Minimization. 2020. 2, 5

[80] Ke Wang, Nikolaos Dimitriadis, Alessandro Favero, Guillermo Ortiz-Jimenez, Francois Fleuret, and Pascal Frossard. Lines: Post-training layer scaling prevents forgetting and enhances model merging. *arXiv preprint arXiv:2410.17146*, 2024. 3

[81] Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jimenez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. *arXiv preprint arXiv:2405.07813*, 2024. 2

[82] Ke Wang, Yiming Qin, Nikolaos Dimitriadis, Alessandro Favero, and Pascal Frossard. Memoir: Lifelong model editing with minimal overwrite and informed retention for llms. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025. 3

[83] Yongxian Wei, Runxi Cheng, Weike Jin, Enneng Yang, Li Shen, Lu Hou, Sinan Du, Chun Yuan, Xiaochun Cao, and Dacheng Tao. Unifying multimodal large language model capabilities and modalities via model merging. *arXiv preprint arXiv:2505.19892*, 2025. 2

[84] Yongxian Wei, Anke Tang, Li Shen, Zixuan Hu, Chun Yuan, and Xiaochun Cao. Modeling multi-task model merging as adaptive projective gradient descent. In *Forty-second International Conference on Machine Learning*, 2025. 2

[85] Yongxian Wei, Anke Tang, Li Shen, Zixuan Hu, Chun Yuan, and Xiaochun Cao. Modeling Multi-Task Model Merging as Adaptive Projective Gradient Descent. 2025. 2

[86] Adina Williams, Nikita Nangia, and Samuel Bowman. 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, 2018. Association for Computational Linguistics. 1

[87] Han Wu, Yuxuan Yao, Shuqi Liu, Zehua Liu, Xiaojin Fu, Xiongwei Han, Xing Li, Hui-Ling Zhen, Tao Zhong, and Mingxuan Yuan. Unlocking efficient long-to-short llm reasoning with model merging. *arXiv preprint arXiv:2503.20641*, 2025. 3

[88] Jianxiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. SUN database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492, 2010. ISSN: 1063-6919. 7

[89] Zhengqi Xu, Ke Yuan, Huiqiong Wang, Yong Wang, Mingli Song, and Jie Song. Training-free pretrained model merging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5915–5925, 2024. 2

[90] Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115, 2023. 1, 2, 5, 6, 7, 8

[91] Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning. *arXiv preprint arXiv:2310.02575*, 2023. 1, 2, 5, 6, 7, 8

[92] 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. *arXiv preprint arXiv:2408.07666*, 2024. 2, 3

[93] Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xiaojun Chen, Xingwei Wang, and Dacheng Tao. Representation surgery for multi-task model merging. *arXiv preprint arXiv:2402.02705*, 2024. 2

[94] Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xingwei Wang, Xiaocun Cao, Jie Zhang, and Dacheng Tao. Surgeryv2: Bridging the gap between model merging and multi-task learning with deep representation surgery. *arXiv preprint arXiv:2410.14389*, 2024. 2

[95] Enneng Yang, Anke Tang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, and Jie Zhang. Continual model merging without data: Dual projections for balancing stability and plasticity. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. 3

[96] Yuxuan Yao, Shuqi Liu, Zehua Liu, Qintong Li, Mingyang Liu, Xiongwei Han, Zhijiang Guo, Han Wu, and Linqi Song. Activation-guided consensus merging for large language models. *arXiv preprint arXiv:2505.14009*, 2025. 3

[97] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2:67–78, 2014. 2

[98] 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 *Forty-first International Conference on Machine Learning*, 2024. 1, 2, 5, 6, 7, 8

[99] Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33:5824–5836, 2020. 1

[100] Amir R Zamir, Alexander Sax, William Shen, Leonidas J Guibas, Jitendra Malik, and Silvio Savarese. Taskonomy:

Disentangling task transfer learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3712–3722, 2018. 5, 1

[101] Fanhu Zeng, Haiyang Guo, Fei Zhu, Li Shen, and Hao Tang. Parameter efficient merging for multimodal large language models with complementary parameter adaptation. *arXiv preprint arXiv:2502.17159*, 2025. 5, 6, 7, 8, 2, 3

[102] Binchi Zhang, Zaiyi Zheng, Zhengzhang Chen, and Jundong Li. Beyond the permutation symmetry of transformers: The role of rotation for model fusion. *arXiv preprint arXiv:2502.00264*, 2025. 2

[103] Haobo Zhang and Jiayu Zhou. Unraveling lora interference: Orthogonal subspaces for robust model merging. *arXiv preprint arXiv:2505.22934*, 2025. 3

[104] Juzheng Zhang, Jiacheng You, Ashwinee Panda, and Tom Goldstein. Lori: Reducing cross-task interference in multi-task low-rank adaptation. *arXiv preprint arXiv:2504.07448*, 2025. 3

[105] Kaichen Zhang, Bo Li, Peiyuan Zhang, Fanyi Pu, Joshua Adrian Cahyono, Kairui Hu, Shuai Liu, Yuanhan Zhang, Jingkang Yang, Chunyuan Li, and Ziwei Liu. Lmms-eval: Reality check on the evaluation of large multimodal models, 2024. 1, 2

[106] Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, Kun Kuang, and Fei Wu. Merging loras like playing lego: Pushing the modularity of lora to extremes through rank-wise clustering. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2025. 1, 2, 5, 6, 7, 8

[107] Shenghe Zheng and Hongzhi Wang. Free-merging: Fourier transform for efficient model merging. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3863–3873, 2025. 5, 6, 7, 8, 2, 3

[108] Ming Zhong, Yelong Shen, Shuohang Wang, Yadong Lu, Yizhu Jiao, Siru Ouyang, Donghan Yu, Jiawei Han, and Weizhu Chen. Multi-lora composition for image generation. *arXiv preprint arXiv:2402.16843*, 2024. 2

[109] Yuyan Zhou, Liang Song, Bingning Wang, and Weipeng Chen. Metagpt: Merging large language models using model exclusive task arithmetic. *arXiv preprint arXiv:2406.11385*, 2024. 2

[110] 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 *The Thirteenth International Conference on Learning Representations*, 2025. 2

[111] Shaobin Zhuang, Yiwei Guo, Yanbo Ding, Kunchang Li, Xinyuan Chen, Yaohui Wang, Fangyikang Wang, Ying Zhang, Chen Li, and Yali Wang. Timestep master: Asymmetrical mixture of timestep lora experts for versatile and efficient diffusion models in vision. *arXiv preprint arXiv:2503.07416*, 2025. 2