Model Merging in the Essential Subspace

Longhua Li[1,2] Lei Qi[1,2*] Qi Tian[3] Xin Geng[1,2*]
[1]School of Computer Science and Engineering, Southeast University, Nanjing, China
[2]Key Laboratory of New Generation Artificial Intelligence Technology and Its Interdisciplinary Applications (Southeast University), Ministry of Education, China
[3]Huawei Technologies, Shanghai, China
lhli@seu.edu.cn, qilei@seu.edu.cn, tian.qi1@huawei.com, xgeng@seu.edu.cn

Abstract

*Model merging aims to integrate multiple task-specific fine-tuned models derived from a shared pre-trained checkpoint into a single multi-task model without additional training. Despite extensive research, task interference remains a major obstacle that often undermines the performance of merged models. In this paper, we propose ESM (**E**ssential **S**ubspace **M**erging), a robust framework for effective model merging. We begin by performing Principal Component Analysis (PCA) on feature shifts induced by parameter updates. The resulting principal directions span an essential subspace that dominantly influences feature representations. Each task's parameter update matrix is projected onto its respective essential subspace for low-rank decomposition before merging. This methodology mitigates inter-task interference while preserving core task-specific functionality. Furthermore, we introduce a multi-level polarized scaling strategy that amplifies parameters containing critical knowledge and suppresses redundant ones, preventing essential knowledge from being overwhelmed during fusion. Extensive experiments across multiple task sets and model scales demonstrate that our method achieves state-of-the-art performance in multi-task model merging.*

1. Introduction

In recent years, the pre-training–fine-tuning paradigm has revolutionized performance across numerous downstream tasks, enabling models to excel by adapting to task-specific data. This success has spurred the creation of many specialized expert models. The growth of public model repositories is now fueling demand for techniques that reuse and integrate these models. Among them, model merging [18, 39, 46, 53] has emerged as a promising, parameter-efficient approach to combine multiple experts into a single versatile model, thus marking a pivotal advance toward the development of general-purpose artificial intelligence.

However, integrating the capabilities of multiple models remains challenging. A key difficulty is inter-task interference, as fine-tuning on different tasks often drives parameters in conflicting directions. Simple averaging methods such as Model Soup [46] mix these updates, diluting task-specific knowledge and leading to suboptimal performance. To alleviate this problem, subsequent studies analyzed task vectors, defined as the parameter differences between fine-tuned and pre-trained models [18, 19, 29, 52]. More recently, advanced methods have employed Singular Value Decomposition (SVD) on task vectors to identify low-rank subspaces that compactly represent task knowledge and reduce redundancy during merging [11, 28, 38, 57]. However, these SVD-based subspaces are not well aligned with task feature spaces and exhibit limited knowledge concentration.

In this paper, we first theoretically analyze the limitations of SVD-based low-rank decomposition for task-specific parameter updates. Although these methods truncate the smallest singular values, they overlook the feature distribution. Consequently, when input tokens are aligned with the truncated singular vectors, significant truncation errors may still occur, as illustrated in Equation 4. Moreover, when merging a large number of task-specific update parameters, the core knowledge of each task may be overwhelmed. As shown in Figure 2 and 3, we observe that the magnitude (norm) of task parameter updates is closely correlated with their directional importance: large-magnitude updates often correspond to critical adjustments essential for capturing task-specific or shared knowledge across tasks, whereas small-magnitude updates, though potentially negligible for individual task performance, may obscure important task knowledge after model merging.

Building on these insights, our method consists of two components. Instead of directly decomposing the task matrix (i.e., the parameter update matrix), we first perform Principal Component Analysis (PCA) on the proxy acti-

*Corresponding authors.

vation shifts induced by the task matrix. The matrix is then decomposed along the resulting principal directions, forming the task's Essential Subspace, which is directly connected to the output feature space and ensures minimal feature loss compared to other low-rank subspaces of the same rank. Each task matrix is subsequently projected onto its Essential Subspace before fusion. Experiments demonstrate that this subspace aligns more closely with the task's feature representation, exhibits stronger energy concentration, and isolates task-specific knowledge more effectively than conventional SVD-based subspaces. Next, we introduce a simple yet effective Polarized Scaling mechanism. It adaptively rescales task matrices at three levels: across layers, tasks, and dimensions, amplifying high-norm (strong-signal) components while suppressing low-norm (weak-noise) ones. This polarization effectively reduces inter-task interference during model fusion and ensures the preservation of essential task knowledge in the merged model.

Our main contributions are summarized as follows:

- We propose a decomposition method that builds an essential subspace aware of activation shift distributions, aligning closely with feature distribution and concentrating energy more effectively, thereby better preserving task-specific knowledge and reducing cross-task interference.
- We show that the pre-fusion and post-fusion task matrix norms reflect gradient confidence and inter-task consensus, respectively. Based on this, we introduce a multi-level polarized scaling mechanism that prevents crucial parameters from being overshadowed during merging.
- Extensive experiments show that our method outperforms existing model merging methods and effectively narrows the gap between the merged model and the expert models.

2. Related Work

2.1. Model Merging

Model merging aims to combine multiple task-specific fine-tuned models into a unified multi-task model without retraining. To ensure merging stability, recent studies typically merge models that are fine-tuned from the same pre-trained checkpoint. Model Soup [46] averages fine-tuned weights to improve generalization, while Task Arithmetic [18] introduces task vectors, defined as the parameter differences between fine-tuned and pre-trained models, to enable vector-based knowledge composition. However, direct averaging of task vectors often causes severe task interference due to conflicting updates. To address this, TIES-Merging [52] trims redundant parameters before averaging salient ones, AdaMerging [54] learns adaptive task-wise coefficients, and DARE [55] resets redundant updates while rescaling the rest. Information-weighted methods such as Fisher Merging [29] and RegMean [19] use Fisher information or input similarity for weighted averaging. Other works refine merging through parameter- or layer-wise strategies [10, 43, 56], or leverage implicit or modular representations to enhance flexibility [3, 16, 58].

Recent advances move beyond raw parameter space to the spectral domain. TSV-M [11] perform Singular Value Decomposition (SVD) on task matrices and merge along the top singular directions that capture dominant functional subspaces. Iso-CTS [28] constructs an isotropic common subspace through singular value normalization followed by task-specific refinements, achieving state-of-the-art performance. However, singular values reflect only the parameter energy rather than their functional impact. To overcome this limitation, we decompose each task matrix within an essential subspace derived from its effect on output activations, which better captures task-specific features. This leads to lower truncation error and improved preservation of task knowledge during merging. Moreover, we introduce a polarized scaling mechanism that amplifies strong signals and suppresses noise, further enhancing fusion performance.

2.2. Model Weight Transformation

Transforming model weights has been extensively studied in areas such as model compression [17, 25, 45] and efficient knowledge transfer [24, 47, 48, 51]. One of the most popular approaches to model weight transformation is based on the low-rank assumption of weight matrices. The LoRA family of methods [8, 9, 15, 27] assumes that fine-tuning updates are inherently low-rank and learns compact matrices to parameterize these updates. Other methods use SVD-based decompositions for parameter-efficient fine-tuning [13, 40] or model compression [26, 35, 44, 45]. More recently, low-rank decompositions have been applied to model merging [11, 28, 38], combining task updates in reduced subspaces to mitigate inter-task interference.

In contrast, our proposed Essential Subspace Decomposition (ESD) constructs the decomposition space not from the weight updates themselves but from the activation shifts induced by these updates. By capturing task-specific principal directions in the activation space, ESD produces sparse yet expressive task representations, reducing cross-task interference while preserving high task fidelity.

3. Methodology

3.1. Preliminaries on Model Merging

Model merging aims to integrate a collection of task-specific models, each fine-tuned from a common pre-trained checkpoint, into a single unified model without additional retraining. Formally, let θ_0 denote the parameters of the pre-trained model, and θ_t be the parameters of the expert model fine-tuned on task t, where $t = 1, \ldots, T$. The fundamental object of interest in model merging is the task update, which captures how fine-tuning shifts the model away from

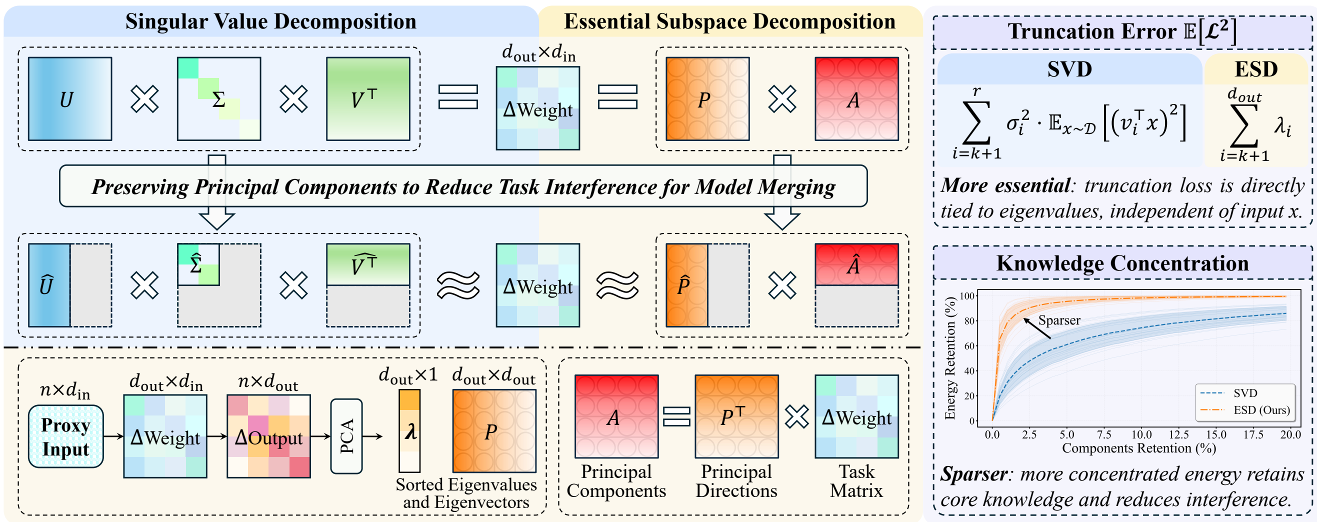


Figure 1. Essential Subspace Decomposition (ESD) versus Singular Value Decomposition (SVD). Unlike SVD, which decomposes the task matrix solely based on weights, ESD decomposes them based on feature shift distributions. When truncating components for merging, ESD's expected truncation error is directly related to the magnitude of the discarded eigenvalues and yields higher knowledge retention.

the pre-trained weights. Following Task Arithmetic [18], the task vector for task t is defined as:

$$\tau_t = \mathrm{Flatten}(\theta_t - \theta_0). \tag{1}$$

Given the structured nature of models, it is preferable to retain the matrix form of the update rather than flattening it into a vector. The task matrix of layer ℓ is defined as:

$$\Delta_{W_t}^{(\ell)} = \theta_t^{(\ell)} - \theta_0^{(\ell)}. \tag{2}$$

Each $\Delta_{W_t}^{(\ell)}$ represents the task-specific parameter update at layer ℓ, preserving the row–column structure essential for spectral analysis and subspace alignment. The goal of model merging is to construct merged weights θ_M that support all tasks, typically in the form:

$$\theta_M = \theta_0 + f\left(\Delta_{W_1}, \cdots, \Delta_{W_T}\right), \tag{3}$$

where $f(\cdot)$ is a merging function, which is the main focus of current model merging research [11, 18, 28, 38].

3.2. Model Merging in Essential Subspace

Unlike previous methods that merge models in the truncated singular vector subspace obtained via SVD, we propose to decompose and merge task matrices within a more essential subspace that is better aligned with the task's output feature space, as illustrated in Figure 1. For simplicity, unless otherwise specified, we omit the layer index ℓ and task identifier t in the task matrix and denote it simply as Δ_W.

The Limitation of Feature Distribution-Agnostic SVD. Recent model merging methods often apply SVD to the task matrix $\Delta_W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, keeping only the top-$k$ singular components to retain essential knowledge and reduce task interference, yielding the truncated approximation $\widehat{\Delta_W}$. However, this parameter-centric view ignores the input feature distribution. While it minimizes the Frobenius norm reconstruction error of Δ_W, it provides no guarantee about preserving the functional output. For an input $x \sim \mathcal{D}$, the expected output error after discarding the smallest $r - k$ singular components is (see Appendix A.1 for derivation):

$$\mathbb{E}_{x\sim\mathcal{D}}\left[\|\Delta_W x - \widehat{\Delta_W} x\|_2^2\right] = \sum_{i=k+1}^{r} \sigma_i^2 \cdot \mathbb{E}_{x\sim\mathcal{D}}\left[(v_i^\top x)^2\right], \tag{4}$$

where r denotes the number of non-zero singular values. As shown, the error depends not only on the discarded singular values σ_i, but also on the alignment between the input distribution and the right singular vectors v_i. A direction with small σ_i may be functionally critical if inputs project strongly onto v_i. By ignoring the input distribution, SVD risks discarding functionally essential information.

Essential Subspace Decomposition (ESD). To address this limitation, we introduce Essential Subspace Decomposition (ESD), an input distribution-aware method that directly connects task matrix decomposition to its functional influence. Instead of relying on the parameter energy of the task matrix, ESD constructs a basis from the principal directions of activation shifts induced by the task matrix. For each task t, we sample a lightweight proxy dataset (32 unlabeled examples in practice). By performing a forward pass through the task-specific fine-tuned model and recording the intermediate activations, we obtain the activation shift. Specifically, given n input tokens of dimension d_{in}, forming $X_{\text{proxy}} \in \mathbb{R}^{n \times d_{\text{in}}}$, the shift is computed as:

$$\Delta_O = X_{\text{proxy}} \Delta_W^\top \in \mathbb{R}^{n \times d_{\text{out}}}, \tag{5}$$

which captures the functional footprint of Δ_W on a representative set of inputs. By performing PCA on Δ_O, we obtain a set of eigenvectors $\boldsymbol{p}_i$ and their corresponding eigenvalues λ_i, sorted by explained variance. These eigenvectors form an orthonormal basis $P = [\boldsymbol{p}_1, \boldsymbol{p}_2, \ldots, \boldsymbol{p}_{d_{\text{out}}}] \in \mathbb{R}^{d_{\text{out}} \times d_{\text{out}}}$ for the output space that is inherently aligned with the task matrix's functional behavior. The original task matrix Δ_W is projected onto the space defined by P, which yields the coordinate matrix $A = P^\top \Delta_W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, where A contains the coordinates of Δ_W with respect to the basis P. Thus, Δ_W can be factorized as:

$$\Delta_W = PA = P(P^\top \Delta_W). \tag{6}$$

We truncate to the top-k principal components to form the essential basis $\hat{P} = [p_1, \ldots, p_k] \in \mathbb{R}^{d_{\text{out}} \times k}$. The corresponding coordinate matrix is $\hat{A} = \hat{P}^\top \Delta_W \in \mathbb{R}^{k \times d_{\text{in}}}$, leading to the low-rank approximation:

$$\widehat{\Delta_W} = \hat{P}\hat{A} = \hat{P}(\hat{P}^\top \Delta_W). \tag{7}$$

We can theoretically show (see Appendix A.2 for derivation) that the expected output error under this decomposition is now decoupled from the input direction:

$$\mathbb{E}_{x \sim \mathcal{D}} \left[\|\Delta_W x - \widehat{\Delta_W} x\|_2^2 \right] = \sum_{i=k+1}^{d_{\text{out}}} \lambda_i. \tag{8}$$

Unlike SVD (Equation 4), the ESD truncation error (Equation 8) depends only on the sum of discarded eigenvalues. Removing small-eigenvalue components thus discards the least functionally relevant directions, making ESD truly "essential" by optimally preserving task expressiveness and enabling sparser, higher-fidelity model merging.

Essential Subspace Merging (ESM). Building upon the proposed ESD, we propose Essential Subspace Merging (ESM), a strategy that robustly aggregates task matrices by only considering the functionally important components. This method is analogous to TSV-M [11], but is adapted to leverage our distribution-aware low-rank factors. The ESM process follows three sequential steps:

1. **Factorization and Truncation.** For each task $t \in \{1, \ldots, T\}$, we first factorize the task matrix Δ_{W_t} into its essential basis P_t and coordinate matrix A_t, such that $\Delta_{W_t} = P_t A_t$, where $P_t \in \mathbb{R}^{d_{\text{out}} \times d_{\text{out}}}$ and $A_t \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$. Given T tasks, we allocate a rank budget $k = \lfloor d_{\text{out}}/T \rfloor$ to each one. We truncate the task-specific factors to their top-k components, resulting in the sparse factors $\hat{P}_t \in \mathbb{R}^{d_{\text{out}} \times k}$ and $\hat{A}_t \in \mathbb{R}^{k \times d_{\text{in}}}$.
2. **Concatenation.** Next, we form the merged basis and coordinate matrices by horizontally and vertically concatenating the truncated factors across all tasks, respectively:

$$P_{\text{cat}} = [\hat{P}_1 | \hat{P}_2 | \ldots | \hat{P}_T] \in \mathbb{R}^{d_{\text{out}} \times (k \cdot T)}, \tag{9}$$

$$A_{\text{cat}} = \begin{bmatrix} \hat{A}_1 \\ \hat{A}_2 \\ \vdots \\ \hat{A}_T \end{bmatrix} \in \mathbb{R}^{(k \cdot T) \times d_{\text{in}}}. \tag{10}$$

The initial merged matrix is simply the product $P_{\text{cat}} A_{\text{cat}}$.
3. **Orthogonalization to Minimize Interference.** The concatenated factors P_{cat} and A_{cat} consist of components derived from different task subspaces, which may not be mutually orthogonal, leading to significant interference in the merged model. To reconstruct the merged matrix with minimized correlation, we must orthogonalize these concatenated factors. Following TSV-M [11], we first compute the SVDs for each concatenated matrix:

$$P_{\text{cat}} = U_P \Sigma_P V_P^\top, \; A_{\text{cat}} = U_A \Sigma_A V_A^\top. \tag{11}$$

To ensure that more important parameter directions are preferentially preserved during orthogonalization, we apply eigenvalue-based weighting to both the directional vectors of P_{cat} and the coordinate vectors of A_{cat} prior to performing the above SVD. We then perform a whitening operation on the matrices by retaining only the orthogonal components U and V. This transformation effectively rotating the basis to be maximally decorrelated:

$$\widetilde{P}_{\text{cat}} = U_P V_P^\top, \; \widetilde{A}_{\text{cat}} = U_A V_A^\top. \tag{12}$$

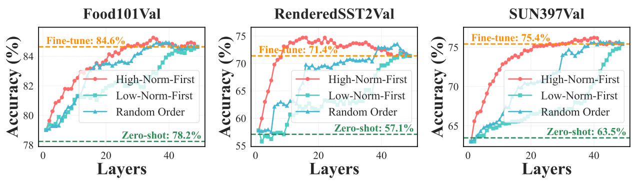
The final merged task matrix $\Delta_{W_{\text{merged}}}$ is then constructed by multiplying these orthogonalized factors:

$$\Delta_{W_{\text{merged}}} = \widetilde{P}_{\text{cat}} \widetilde{A}_{\text{cat}}. \tag{13}$$

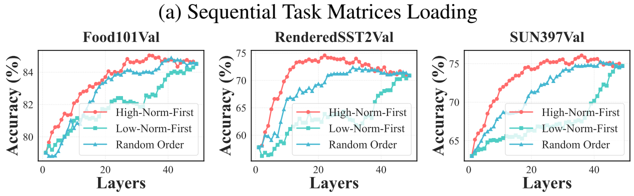
3.3. Polarized Scaling for Knowledge Enhancement

Although ESD preserves task matrix functionality, intense weight competition during fusion can still overwhelm critical knowledge. We note that the norm of a weight block update reflects its directional confidence and inter-task consensus, serving as an effective scaling factor. We therefore introduce Polarized Scaling (Figure 4) to amplify signals and suppress noise for improved merging performance.

Empirical Evidence: Single-Task Merging. As shown in Figure 2a, we sequentially add task matrices from a fine-tuned model to the zero-shot base model, comparing three orders: high-norm-first, low-norm-first, and random. The results consistently indicate that adding high-norm matrices first yields the best performance. Moreover, incorporating low-norm matrices in later stages leads to clear performance degradation. To isolate the effect of direction rather

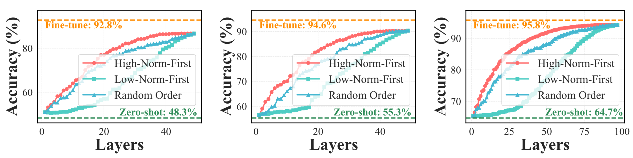


(a) Sequential Task Matrices Loading



(b) Sequential *Averaged-Norm* Task Matrices Loading

Figure 2. Performance evaluation of layer-wise task matrix loading under different ordering strategies on a pre-trained ViT-B/32 backbone. (a) Direct loading of task matrices. (b) Loading with layer-wise norms averaged to reflect directional importance.



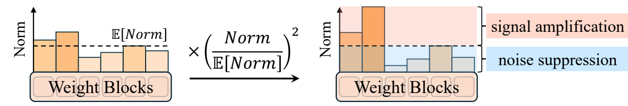
(a) ViT-B/32 (b) ViT-B/16 (c) ViT-L/14

Figure 3. Performance evaluation on the 8-task benchmark when loading merged task matrices layer-by-layer into a pre-trained backbone under different ordering strategies.

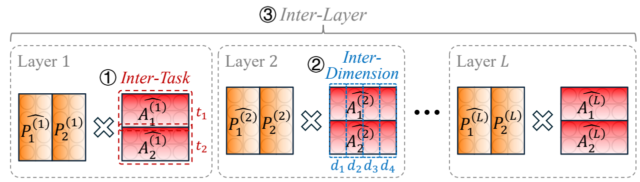
than scale, we repeat the experiment after normalizing each layer's task matrix to the same average norm (Figure 2b). The high-norm-first order still performs best, revealing a key insight: *high-norm task matrices matter because they capture consistent, task-aligned directions in the optimization landscape, encoding the task's essential knowledge*.

Empirical Evidence: Multi-Task Merging. As shown in Figure 3, we load the aggregated task matrices, obtained after Essential Subspace Merging (ESM), into the pre-trained model, again following the three loading orders based on the merged matrix norm. The results confirm that prioritizing high-norm components yields superior merged performance. This suggests that *the high-norm components in the merged matrix are likely the inter-task consensus, representing shared, important knowledge directions*.

Proposed Polarized Scaling (PS). Based on these observations, we introduce a simple yet effective Polarized Scaling (PS) scheme designed to proactively suppress the influence of noisy, low-confidence parameters and enhance the strength of essential, high-confidence ones. The core idea is to scale a weight block by the square of its relative mag-



(a) Polarized Scaling



(b) Scaling Hierarchy

Figure 4. (a) The proposed Polarized Scaling uses the norm of parameter updates as scaling factors to amplify essential parameters submerged by redundant ones. (b) This scaling is applied at three distinct levels: across tasks, dimensions, and layers.

nitude. We apply this scaling at three hierarchical levels:

1. **Inter-Task Scaling.** Within the same layer ℓ, we scale each individual truncated coordinate matrix $\hat{A}_t^{(\ell)}$ relative to the average norm across all tasks T. This prevents important task signals from being masked by the accumulation of noise from numerous competing tasks. The scaling coefficient $s_t^{(\ell)}$ for task t at layer ℓ is defined as:

$$s_t^{(\ell)} = \left(\frac{\left| \hat{A}_t^{(\ell)} \right|_F}{\mathbb{E}_{i \in \{1,\dots,T\}} \left[\left| \hat{A}_i^{(\ell)} \right|_F \right]} \right)^2. \tag{14}$$

   The scaled coordinate matrix, $s_t^{(\ell)} \cdot \hat{A}_t^{(\ell)}$, is used in place of the original $\hat{A}_t^{(\ell)}$ to construct $A_{\text{cat}}^{(\ell)}$ in Equation 10.

2. **Inter-Dimension Scaling.** After task-wise scaling and concatenation, we further scale the columns of the coordinate matrix $A_{\text{cat}}^{(\ell)}$. This highlights the specific dimensions that exhibit strong consensus across all tasks. Let $\mathbf{a}_j$ be the j-th column vector of $A_{\text{cat}}^{(\ell)}$. The scaling coefficient $c_j^{(\ell)}$ for column j is calculated as follows:

$$c_j^{(\ell)} = \left(\frac{\left| \mathbf{a}_j^{(\ell)} \right|_2}{\mathbb{E}_{i \in \{1,\dots,d_{\text{in}}\}} \left[\left| \mathbf{a}_i^{(\ell)} \right|_2 \right]} \right)^2. \tag{15}$$

   After applying weighting to each column of $A_{\text{cat}}^{(\ell)}$, the representation becomes more focused on the input dimensions that are effective for every task.

3. **Inter-Layer Scaling.** Finally, we apply scaling across the different layers $\ell \in \mathcal{L}$ of the merged task matrix $\Delta_{W_{\text{merged}}}^{(\ell)}$. This is crucial because residual connections introduce inter-layer competition, where less important layers can mask vital information from important layers. We only compare layers of the same nature (e.g., all attention QKV projection layers, or all MLP up-projection

Table 1. Average absolute accuracy on model merging benchmarks, with normalized average accuracy shown as subscripts in parentheses. "*Zero-shot*" (pre-trained model) and "*Fine-tuned*" (fine-tuned models) results are presented as the lower and upper bounds, respectively.

Method	ViT-B/32			ViT-B/16			ViT-L/14		
	8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks
Zero-shot	48.3	57.2	56.1	55.3	61.3	59.7	64.7	68.2	65.2
Fine-tuned	92.8	90.9	91.3	94.6	92.8	93.2	95.8	94.3	94.7
Weight Averaging [46]	$66.3_{(72.1)}$	$64.3_{(71.1)}$	$61.0_{(67.5)}$	$72.2_{(76.6)}$	$69.5_{(74.8)}$	$65.3_{(70.4)}$	$79.6_{(83.2)}$	$76.7_{(81.1)}$	$71.6_{(75.6)}$
Task Arithmetic [18]	$70.8_{(76.5)}$	$65.3_{(72.1)}$	$60.5_{(66.8)}$	$75.4_{(79.6)}$	$70.5_{(75.9)}$	$65.8_{(70.8)}$	$84.9_{(88.7)}$	$79.4_{(84.0)}$	$74.0_{(78.1)}$
TIES-Merging [52]	$75.1_{(81.0)}$	$68.0_{(74.8)}$	$63.4_{(69.9)}$	$79.7_{(84.3)}$	$73.2_{(78.7)}$	$68.2_{(73.3)}$	$86.9_{(90.7)}$	$79.5_{(84.1)}$	$75.7_{(79.8)}$
Consensus TA [42]	$75.0_{(80.8)}$	$70.4_{(77.4)}$	$65.4_{(72.0)}$	$79.4_{(83.9)}$	$74.4_{(79.9)}$	$69.8_{(74.9)}$	$86.3_{(90.1)}$	$82.2_{(86.9)}$	$79.0_{(83.2)}$
TSV-M [11]	$85.9_{(92.3)}$	$80.1_{(87.9)}$	$77.1_{(84.3)}$	$89.0_{(93.9)}$	$84.6_{(91.0)}$	$80.6_{(86.5)}$	$93.0_{(97.0)}$	$89.2_{(94.4)}$	$87.7_{(92.5)}$
Iso-C [28]	$86.3_{(92.9)}$	$80.3_{(88.1)}$	$75.5_{(82.5)}$	$90.6_{(95.6)}$	$84.8_{(91.1)}$	$79.6_{(85.4)}$	$94.2_{(98.3)}$	$89.3_{(94.5)}$	$87.6_{(92.2)}$
Iso-CTS [28]	$86.2_{(92.8)}$	$81.7_{(89.7)}$	$78.1_{(85.5)}$	$91.1_{(96.1)}$	$86.4_{(92.8)}$	$82.4_{(88.4)}$	$94.7_{(98.8)}$	$91.0_{(96.3)}$	$90.1_{(94.9)}$
ESM (Ours)	$\mathbf{88.4_{(95.3)}}$	$\mathbf{83.7_{(92.0)}}$	$\mathbf{81.3_{(88.9)}}$	$\mathbf{91.8_{(97.0)}}$	$\mathbf{87.4_{(94.1)}}$	$\mathbf{84.9_{(91.1)}}$	$\mathbf{94.8_{(98.9)}}$	$\mathbf{91.3_{(96.8)}}$	$\mathbf{90.4_{(95.3)}}$

layers) as comparing norms across different layer types is meaningless. Let $\mathcal{L}_{\text{type}}$ be the set of layers of the same nature. The scaling factor β_ℓ for layer $\ell \in \mathcal{L}_{\text{type}}$ is:

$$\beta_\ell = \left(\frac{\left| \Delta^{(\ell)}_{W_{\text{merged}}} \right|_F}{\mathbb{E}_{i\in\mathcal{L}_{\text{type}}} \left[\left| \Delta^{(i)}_{W_{\text{merged}}} \right|_F \right]} \right)^2 . \tag{16}$$

The scaled merged task matrix is $\beta_\ell \cdot \Delta^{(\ell)}_{W_{\text{merged}}}$. This scaling ensures that layers exhibiting stronger directional consensus (higher norm) retain their relative importance.

The parameter for the ℓ-th layer of the final merged multi-task model is given by:

$$\theta^{(\ell)}_M = \theta^{(\ell)}_0 + \alpha \cdot \beta_\ell \cdot \Delta^{(\ell)}_{W_{\text{merged}}}, \tag{17}$$

where α is chosen on a held-out validation set as in [28].

4. Experiments

4.1. Experimental Setup

Following [42], we evaluate multi-task merging on benchmarks of 8, 14, and 20 tasks. The 8-task set includes: Cars [21], DTD [4], EuroSAT [14], GTSRB [37], MNIST [23], RESISC45 [2], SUN397 [50], and SVHN [30]. The 14-task benchmark extends this with CIFAR100 [22], STL10 [6], Flowers102 [31], OxfordIIITPet [32], PCAM [41], and FER2013 [12]. The 20-task set further incorporates EMNIST [7], CIFAR10 [22], Food101 [1], FashionMNIST [49], RenderedSST2 [36], and KMNIST [5]. We employ three variants of the CLIP model [33] with ViT-B/32, ViT-B/16, and ViT-L/14 visual encoders as our pre-trained base models. For the task-specific parameters, we utilize the fine-tuned checkpoints provided in the TALL-masks [42] repository, corresponding to the tasks listed above. We report performance using both absolute and normalized accuracy, adhering to the standard evaluation practices [42].

4.2. Main Results

As presented in Table 1, we compare our proposed method, ESM, against a comprehensive suite of established model merging methods: Weight Averaging [46], Task Arithmetic [18], TIES-Merging [52], Consensus TA [42], TSV-M [11], and Iso-CTS [28]. We also include the performance of the zero-shot base model and the average performance of single-task fine-tuned expert models as the lower and upper bounds, respectively. Our results demonstrate that ESM achieves state-of-the-art performance across all benchmark settings and base model variants. Notably, the performance gains are particularly pronounced in challenging scenarios with a large number of tasks or when using models with smaller capacity (e.g., ViT-B/32 vs. ViT-L/14). This strongly validates the effectiveness of our method in preserving essential task-specific knowledge while significantly mitigating inter-task interference, leading to a superior and more robust merged model.

4.3. Ablation and Analysis

Ablation of Proposed ESD and PS. We conduct an ablation study of the proposed ESD and PS modules, as shown in Table 2. The results demonstrate that decomposing and merging in the essential subspace leads to a substantial improvement over performing the same operations in the SVD subspace, which is consistent with our theoretical and experimental analyses. Furthermore, the proposed PS module helps reduce task interference while preserving essential task-specific knowledge, leading to additional performance gains. Notably, PS proves beneficial not only when combined with ESD-based merging, but also significantly im-

Table 2. Ablation study of the proposed Essential Subspace Decomposition (ESD)-based merging and Polarized Scaling. Results are reported in terms of average absolute accuracy, with normalized average accuracy shown as subscripts in parentheses.

Method	Decomposition		Polarized Scaling			ViT-B/16			ViT-L/14		
	SVD	ESD	β_ℓ	$s_t^{(\ell)}$	$c_j^{(\ell)}$	8 tasks	14 tasks	20 tasks	8 tasks	14 tasks	20 tasks
Baseline	✓	✗	✗	✗	✗	$89.0_{(93.9)}$	$84.6_{(91.0)}$	$80.6_{(86.5)}$	$93.0_{(97.0)}$	$89.2_{(94.4)}$	$87.7_{(92.5)}$
	✓	✗	✓	✓	✓	$89.6_{(94.6)}$	$85.4_{(91.9)}$	$82.1_{(88.1)}$	$93.4_{(97.4)}$	$89.6_{(95.0)}$	$88.1_{(93.0)}$
	✗	✓	✗	✗	✗	$90.9_{(95.9)}$	$85.8_{(92.2)}$	$82.8_{(88.7)}$	$94.5_{(98.6)}$	$90.7_{(95.9)}$	$89.5_{(94.2)}$
	✗	✓	✓	✗	✗	$91.4_{(96.4)}$	$86.6_{(93.1)}$	$83.7_{(89.6)}$	$94.6_{(98.7)}$	$90.9_{(96.2)}$	$90.0_{(94.8)}$
ESM (Ours)	✗	✓	✓	✓	✓	$\mathbf{91.8}_{(\mathbf{97.0})}$	$\mathbf{87.4}_{(\mathbf{94.1})}$	$\mathbf{84.9}_{(\mathbf{91.1})}$	$\mathbf{94.8}_{(\mathbf{98.9})}$	$\mathbf{91.3}_{(\mathbf{96.8})}$	$\mathbf{90.4}_{(\mathbf{95.3})}$

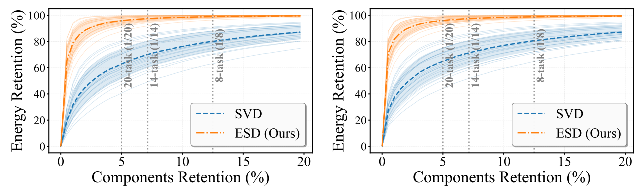
Table 3. Time cost of major operations during model merging. "*Forward*": forward pass of 32 proxy samples to obtain Δ_O. "*PCA*": principal component analysis performed on Δ_O. "*Orthogonalization*": orthogonalization of the matrices P_{cat} and A_{cat}.

Model	*Forward*	*PCA*	*Orthogonalization*
ViT-B/32	0.03 s/task	1.31 s/task	13.74 s (once)
ViT-B/16	0.04 s/task	1.39 s/task	13.89 s (once)
ViT-L/14	0.06 s/task	5.20 s/task	70.83 s (once)

proves the performance of the baseline method operating in the SVD subspace, highlighting its general applicability.

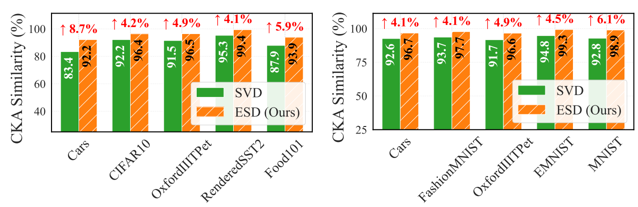
Computational Overhead. We report the major computational overhead introduced by our method, which includes the forward pass of a 32-sample proxy set to compute the activation shift Δ_O, the PCA on Δ_O to obtain the principal directions for the essential subspace, and the orthogonalization of the matrices P_{cat} and A_{cat}. As shown in Table 3, the experiments conducted on an RTX 4090 GPU demonstrate that the additional overhead of our method is minimal.

Comparison of ESD and SVD. Figure 5 shows the cumulative energy retained as different proportions of components are preserved, defined using the singular values (SVD) or eigenvalues (ESD) as detailed in Appendix C.7. Our proposed ESD exhibits a highly concentrated energy distribution, indicating its ability to preserve essential task-specific knowledge with fewer components. Figure 6 evaluates feature preservation when retaining only 5% of components per layer, using Centered Kernel Alignment (CKA) similarity [20]. We measure the similarity between the merged model and the fine-tuned expert using the class token from the final layer, based on the feature difference relative to the zero-shot pre-trained model. ESD more effectively preserves task-specific features than SVD, further confirming its advantage in retaining critical knowledge.



(a) ViT-B/16 (b) ViT-L/14

Figure 5. Energy retention as a function of the fraction of principal components retained from the task matrices. Results are reported for each of the 20 tasks as well as the average.



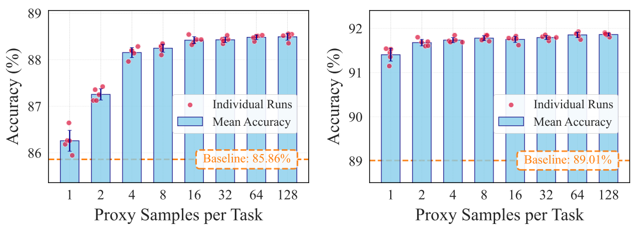
(a) ViT-B/16 (b) ViT-L/14

Figure 6. CKA similarity with the fine-tuned model when retaining only 5% of principal components using SVD and ESD.

Impact of Proxy Dataset Composition. We analyze how the composition of the proxy dataset affects Essential Subspace Merging (ESM). Table 4 shows the average performance and variance over five runs, with a fixed proxy set size of 32 samples. Our default setup uses unlabeled samples randomly selected from the respective task's dataset, denoted as "*Random (ID)*". We also experiment with extreme scenarios: using samples from only a single class within the task data ("*Class Imbalance*") and using samples randomly drawn from an out-of-distribution dataset (ImageNet-1k [34]), denoted as "*Random (OOD)*". Results indicate stable merging performance even with large distribution shifts, which we attribute to the consistent sparse patterns in the features extracted by each task-specific model, regardless of the input data distribution. This finding enhances the generality of our approach, demonstrating that it provides stable performance gains over SVD-based merg-

Table 4. Impact of proxy dataset composition on merging performance, evaluated on the 8-task benchmark. "*Random (ID)*": random sampling from in-distribution task data. "*Class Imbalance*": sampling only a single class per task. "*Random (OOD)*": random sampling from out-of-distribution ImageNet-1k [34].

Method	Sampling	ViT-B/32	ViT-B/16	ViT-L/14
SVD	-	86.6	89.6	93.4
ESD	*Random (ID)*	$88.4_{\pm 0.06}$	$91.8_{\pm 0.04}$	$94.8_{\pm 0.07}$
ESD	*Class Imbalance*	$88.4_{\pm 0.10}$	$91.8_{\pm 0.06}$	$94.8_{\pm 0.04}$
ESD	*Random (OOD)*	$88.3_{\pm 0.04}$	$91.8_{\pm 0.04}$	$94.8_{\pm 0.01}$



(a) ViT-B/32 (b) ViT-B/16

Figure 7. Ablation study on the number of proxy samples used in ESM, reporting the average accuracy on the 8-task benchmark. Each configuration was evaluated over five random runs.

ing methods even when task-specific data is unavailable.

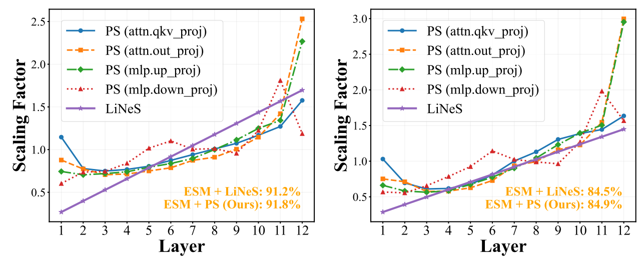
Ablation of Proxy Dataset Size. We perform an ablation study on the size of the proxy dataset used in Essential Subspace Merging (ESM), as illustrated in Figure 7. The results show that using only four unlabeled samples per task is sufficient to achieve robust performance, while even a single sample can consistently outperform the baseline.

Detailed Ablation Study of Polarized Scaling. We perform a detailed ablation of Polarized Scaling (PS) in Table 5. We compare three alternatives: (i) "*Reverse*", which applies the reciprocal of the scaling factors; (ii) "*Noise*$--$", which retains only factors < 1 to suppress noisy parameters; and (iii) "*Signal*$++$", which retains only factors $>$ 1 to enhance important parameters. Experimental results show that, compared with "*None*" (i.e., without scaling), the "*Reverse*" operation significantly degrades performance because important parameters are overwhelmed by redundant ones. Both "*Noise*$--$" and "*Signal*$++$" improve over "*None*" by raising the signal-to-noise ratio of important parameters. Our full PS method, which combines both suppression and amplification, achieves the best performance.

Analysis of Scaling Coefficients. Figure 8 compares the layer-wise coefficients of our Polarized Scaling (PS) with LiNeS [43]. Based on the functional roles of shallow and

Table 5. Detailed ablation study of the proposed Polarized Scaling. The symbol γ denotes the scaling factor at three different levels: s, c, or β. The following variants are compared: (i) "*Reverse*": taking the reciprocal of the scaling factors; (ii) "*Noise*$--$": retaining only factors < 1 to suppress noisy parameters; (iii) "*Signal*$++$": retaining only factors > 1 to enhance important parameters.

Method	Scaling	ViT-B/32		
		8 tasks	14 tasks	20 tasks
None	-	$86.7_{(93.4)}$	$81.1_{(89.0)}$	$78.1_{(85.5)}$
Reverse	$1/\gamma$	$82.9_{(89.2)}$	$76.3_{(83.7)}$	$72.6_{(79.4)}$
Noise$--$	$\min(\gamma, 1)$	$87.4_{(94.1)}$	$83.0_{(91.2)}$	$80.5_{(88.1)}$
Signal$++$	$\max(\gamma, 1)$	$87.7_{(94.6)}$	$82.8_{(91.0)}$	$80.2_{(87.8)}$
PS (Ours)	γ	$\mathbf{88.4}_{(\mathbf{95.3})}$	$\mathbf{83.7}_{(\mathbf{92.0})}$	$\mathbf{81.3}_{(\mathbf{88.9})}$



(a) 8-task (b) 20-task

Figure 8. Layer-wise scaling coefficients for merged ViT-B/16.

deep blocks, LiNeS applies coefficients that increase linearly with depth, requiring a validation set to determine the linear multiplier. In contrast, our PS method provides a more refined measure of parameter importance by leveraging layer norms. It performs separate scaling for four key layer types in a ViT model: the QKV and Output projections in the attention module, and the Up and Down projections in the MLP. This approach not only achieves superior merging performance but also removes the need for a validation set by computing coefficients directly from parameter norms.

5. Conclusion

In this paper, we propose Essential Subspace Merging (ESM), a novel model merging framework that narrows the gap to expert models. It introduces the ESD, a feature shift distribution-aware decomposition method that identifies a sparser and more functionally critical subspace than SVD, thereby reducing inter-task interference. Observing that high-norm updates indicate consensus-driven directions, we further propose Polarized Scaling (PS), a three-level mechanism that amplifies high-confidence signals while suppressing noise. ESM achieves state-of-the-art performance and provides a robust and effective solution for model merging.

Acknowledgment

This research was supported by the Jiangsu Science Foundation (BG2024036, BK20243012), the National Science Foundation of China (62125602, U24A20324, 92464301), the New Cornerstone Science Foundation through the XPLORER PRIZE, the Fundamental Research Funds for the Central Universities (2242025K30024), and the Big Data Computing Center of Southeast University.

References

[1] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *ECCV*, 2014. 6

[2] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 2017. 6

[3] 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. 2

[4] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 6

[5] Tarin Clanuwat, Mikel Bober-Irizar, Asanobu Kitamoto, Alex Lamb, Kazuaki Yamamoto, and David Ha. Deep learning for classical japanese literature. *arXiv preprint arXiv:1812.01718*, 2018. 6

[6] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011. 6

[7] Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In *IJCNN*, 2017. 6

[8] Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. Qlora: Efficient finetuning of quantized llms. In *NeurIPS*, 2023. 2

[9] Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, et al. Parameter-efficient fine-tuning of large-scale pre-trained language models. *NMI*, 2023. 2

[10] 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. In *NeurIPS*, 2024. 2

[11] 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*, 2025. 1, 2, 3, 4, 6

[12] Ian J Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *ICONIP*, 2013. 6

[13] Ligong Han, Yinxiao Li, Han Zhang, Peyman Milanfar, Dimitris Metaxas, and Feng Yang. Svdiff: Compact parameter space for diffusion fine-tuning. In *ICCV*, 2023. 2

[14] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *JSTARS*, 2019. 6

[15] Edward J Hu, 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. 2

[16] Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. Emr-merging: Tuning-free high-performance model merging. In *NeurIPS*, 2024. 2

[17] Xin Huang, Longhua Li, Lei Qi, and Xin Geng. Few-shot network thinning: A unified dense compression framework for cnns and vits. *TMC*, 2026. 2

[18] Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *ICLR*, 2023. 1, 2, 3, 6

[19] Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models. In *ICLR*, 2023. 1, 2

[20] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *ICML*, 2019. 7

[21] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCV workshops*, 2013. 6

[22] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. Technical report, Toronto, ON, Canada, 2009. 6

[23] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 2002. 6

[24] Longhua Li, Lei Qi, and Xin Geng. One-shot knowledge transfer for scalable person re-identification. In *ICCV*, 2025. 2

[25] Longhua Li, Lei Qi, and Xin Geng. Stratified knowledge-density super-network for scalable vision transformers. *arXiv preprint arXiv:2511.11683*, 2025. 2

[26] Yixiao Li, Yifan Yu, Qingru Zhang, Chen Liang, Pengcheng He, Weizhu Chen, and Tuo Zhao. Losparse: Structured compression of large language models based on low-rank and sparse approximation. In *ICML*, 2023. 2

[27] Xin Lin, Yajiao Wang, Zhixiong Zhang, and Mengting Zhang. Scientific claim recognition via staged fine-tuning with lora. *DATA INTELLIGENCE*, 2025. 2

[28] 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. 1, 2, 3, 6

[29] Michael S Matena and Colin A Raffel. Merging models with fisher-weighted averaging. In *NeurIPS*, 2022. 1, 2

[30] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Baolin Wu, Andrew Y Ng, et al. Reading digits in nat-

ural images with unsupervised feature learning. In *NeurIPS workshops*, 2011. 6

[31] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008. 6

[32] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *CVPR*, 2012. 6

[33] 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*, 2021. 6

[34] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 2015. 7, 8

[35] Rajarshi Saha, Varun Srivastava, and Mert Pilanci. Matrix compression via randomized low rank and low precision factorization. In *NeurIPS*, 2023. 2

[36] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, 2013. 6

[37] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *IJCNN*, 2011. 6

[38] George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with svd to tie the knots. In *ICLR*, 2025. 1, 2, 3

[39] Wenju Sun, Qingyong Li, Wen Wang, Yang Liu, Yangliao Geng, and Boyang Li. Towards minimizing feature drift in model merging: Layer-wise task vector fusion for adaptive knowledge integration. In *NeurIPS*, 2025. 1

[40] Yanpeng Sun, Qiang Chen, Xiangyu He, Jian Wang, Haocheng Feng, Junyu Han, Errui Ding, Jian Cheng, Zechao Li, and Jingdong Wang. Singular value fine-tuning: Few-shot segmentation requires few-parameters fine-tuning. In *NeurIPS*, 2022. 2

[41] Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *MICCAI*, 2018. 6

[42] Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jimenez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. In *ICML*, 2024. 6

[43] Ke Wang, Nikolaos Dimitriadis, Alessandro Favero, Guillermo Ortiz-Jimenez, François Fleuret, and Pascal Frossard. Lines: Post-training layer scaling prevents forgetting and enhances model merging. In *ICLR*, 2025. 2, 8

[44] Xin Wang, Samiul Alam, Zhongwei Wan, Hui Shen, and Mi Zhang. Svd-llm v2: Optimizing singular value truncation for large language model compression. In *NAACL*, 2025. 2

[45] Xin Wang, Yu Zheng, Zhongwei Wan, and Mi Zhang. Svd-llm: Truncation-aware singular value decomposition for large language model compression. In *ICLR*, 2025. 2

[46] Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, et al. Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *ICML*, 2022. 1, 2, 6

[47] Shiyu Xia, Miaosen Zhang, Xu Yang, Ruiming Chen, Haokun Chen, and Xin Geng. Transformer as linear expansion of learngene. In *AAAI*, 2024. 2

[48] Shi-Yu Xia, Wenxuan Zhu, Xu Yang, and Xin Geng. Exploring learngene via stage-wise weight sharing for initializing variable-sized models. In *IJCAI*, 2024. 2

[49] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017. 6

[50] Jianxiong Xiao, Krista A Ehinger, James Hays, Antonio Torralba, and Aude Oliva. Sun database: Exploring a large collection of scene categories. *IJCV*, 2016. 6

[51] Jiaze Xu, Shiyu Xia, Xu Yang, Jiaqi Lv, and Xin Geng. Learngene tells you how to customize: Task-aware parameter initialization at flexible scales. In *ICML*, 2025. 2

[52] Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. In *NeurIPS*, 2023. 1, 2, 6

[53] 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. 1

[54] 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. 2

[55] 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. 2

[56] Frederic Z Zhang, Paul Albert, Cristian Rodriguez-Opazo, Anton van den Hengel, and Ehsan Abbasnejad. Knowledge composition using task vectors with learned anisotropic scaling. In *NeurIPS*, 2024. 2

[57] 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. *arXiv preprint arXiv:2603.06242*, 2026. 1

[58] Shenghe Zheng and Hongzhi Wang. Free-merging: Fourier transform for efficient model merging. In *ICCV*, 2025. 2