When Shared Knowledge Hurts: Spectral Over-Accumulation in Model Merging

Anonymous Authors[1]

Abstract

Model merging combines multiple fine-tuned models into a single model by *adding* their weight updates, providing a lightweight alternative to retraining. Existing methods primarily target resolving conflicts between task updates, leaving the failure mode of over-counting shared knowledge unaddressed. We show that when tasks share aligned spectral directions (*i.e.*, overlapping singular vectors), a simple linear combination repeatedly accumulates these directions, inflating the singular values and biasing the merged model toward shared subspaces. To mitigate this issue, we propose Singular Value Calibration (SVC), a training-free and data-free post-processing method that quantifies subspace overlap and rescales inflated singular values to restore a balanced spectrum. Across vision and language benchmarks, SVC consistently improves strong merging baselines and achieves state-of-the-art performance. Furthermore, by modifying only the singular values, SVC improves the performance of Task Arithmetic by 13.0%.

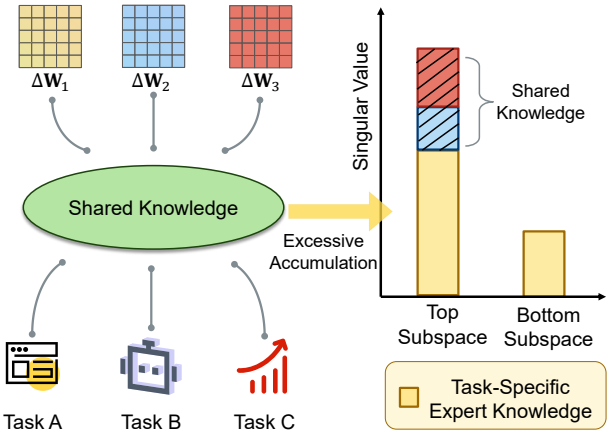

Figure 1. **Shared knowledge accumulation in model merging.** When merging task matrices ($\Delta \mathbf{W}_i$) from multiple tasks, shared knowledge that aligns across tasks can be over-counted, resulting in singular-value inflation in the merged model's spectrum. This inflation is concentrated in a few top spectral subspaces, causing the merged model to be dominated by shared directions, while task-specific components in the remaining subspaces are suppressed.

1. Introduction

Model merging combines trained models into a single model by operating directly in weight space (Ruan et al., 2025). Compared with retraining from scratch or classical ensembling, it enables direct manipulation of weight updates to integrate capabilities (Shah et al., 2025; Ortiz-Jimenez et al., 2023), forget undesirable knowledge (Ilharco et al., 2022; Ni et al., 2024), and accelerate iteration of large-scale models (including LLMs) (Goddard et al., 2024; Wan et al., 2024). Current research focuses on merging models fine-tuned on different tasks using the same pre-trained backbone, yielding a single model with enhanced multi-task capabilities. A central tool in this process is the use of task vectors (Ilharco et al., 2022), which capture differences between pre-trained and fine-tuned weights. In this context, the matrix-valued weight difference at each layer is referred to as the task matrix (TM) (Marczak et al., 2025a). Task matrices play a crucial role in analyzing interference during merging.

Prior work mainly improves merging by mitigating task-matrix conflicts. However, conflict is not the only source of interference. Recent studies report that cross-task alignment in spectral directions, manifested as overlap among singular vectors of different task matrices, is also associated with degradation after merging (Gargiulo et al., 2025). At first glance, this is unexpected. Components shared across tasks, which we call shared knowledge, are typically viewed as useful signals that should transfer rather than harm.

This observation points to a complementary mechanism that governs merged performance. A task is influenced not only by its task-specific update, but also by shared knowledge introduced by other tasks through the merge. When several task matrices align in the same spectral subspaces, a simple linear combination repeatedly aggregates the same shared spectral components, leading to spectral over-counting in

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

those subspaces. As a consequence, the merged update exhibits singular-value inflation and an imbalanced spectrum. Crucially, this inflation concentrates in a few top spectral subspaces with the largest singular values, so the merged model overemphasizes these dominant directions while underrepresenting the remaining subspaces, reducing downstream performance. Fig. 1 visualizes this pattern: shared knowledge is widespread across tasks, yet its over-counting leads to inflation in a few top spectral subspaces. These findings motivate measuring subspace-wise over-counting and correcting inflated singular values during merging.

To address this issue, we propose Singular Value Calibration (SVC), a training-free and data-free method for calibrating a merged update in spectral space. SVC targets the singular-value inflation caused by spectral over-counting, so that the merged model is less dominated by a few shared subspaces.

To make this correction possible, we decompose the merged task matrix into spectral subspaces and use its column-space basis as a shared coordinate system. In this basis, each task matrix can be expressed as subspace-wise responses, which makes different tasks comparable within the same subspace. With tasks aligned to the same basis, we evaluate whether the merged update preserves each task-specific response. Specifically, we project the merged response onto the task-specific response and use the resulting projection coefficient to quantify how much the merged update amplifies that direction. A coefficient larger than 1 means the merged update has accumulated extra mass along that direction, indicating spectral over-counting from cross-task alignment.

Based on these projection coefficients, SVC measures subspace-wise overlap and converts it into a calibration strength for each spectral subspace. It then rescales the corresponding singular values while keeping the spectral directions unchanged. As a result, SVC restores a more balanced spectrum and consistently improves merging performance across vision and language benchmarks. We summarize our contributions as follows:

- We identify spectral over-counting as a key failure mode in model merging, where redundant aggregation of shared knowledge induces singular-value inflation in a small set of dominant spectral subspaces and suppresses other components.
- We propose Singular Value Calibration (SVC), a training-free and data-free method that quantifies column-space overlap in the merged spectral basis and calibrates singular values to restore spectral balance.
- We provide theoretical analysis and empirical evidence validating SVC, which improves Task Arithmetic by 13.0%, and enables targeted improvements for specific tasks such as preference optimization.

2. Related Work

Dynamic Model Merging. Unlike model ensembling, which combines the outputs or predictions of multiple independent models to improve generalization (Dong et al., 2020), model merging operates directly at the weight level. In its common training-free (Zhang et al., 2025; Li et al., 2025; Yuan et al., 2025) form, it integrates the knowledge encoded in the parameters of several trained models into a single unified model (Yang et al., 2024a). This approach addresses challenges such as catastrophic forgetting (Chitale et al., 2023; Zhu et al., 2024; Marczak et al., 2025b), domain shift (Izmailov et al., 2019; Wortsman et al., 2022), and the efficient construction of LLMs (Dekoninck et al., 2024; Aiello et al., 2023). To reduce conflicts among models, a line of work studies dynamic merging, where the behavior of the merged model depends on the input. For example, DaWin (Oh et al., 2024) performs input-conditioned interpolation, EMR-Merging (Huang et al., 2024) and TALL-Mask (Wang et al., 2024) learn task-specific masks, and Twin-Merging (Lu et al., 2024) introduces task experts in the spirit of mixture-of-experts. These approaches can be effective, but they require task labels or routing decisions at inference time; in contrast, SVC is a static, training- and data-free post-hoc calibration that improves merged models without any additional inference-time information.

Static Model Merging. Early research in model merging primarily focused on weight averaging and traditional interpolation strategies (Wortsman et al., 2022; Ilharco et al., 2022; Matena & Raffel, 2022; Jin et al., 2023). These methods allowed rapid assembly of models with expertise from multiple tasks but often resulted in sub-optimal performance due to unresolved conflicts or redundancies among weights. Subsequent work has attempted to address these issues by mitigating inter-model conflicts under either data-dependent or data-free settings. Data-dependent methods typically require auxiliary data, such as validation sets or unlabeled test inputs. For example, PCB-Merging (Du et al., 2024) uses a validation set, NPS-Pruning (Du et al., 2025) relies on a calibration set, and AdaMerging (Yang et al., 2024c) and Trust Region Arithmetic (Sun et al., 2025b) perform test-time adaptation. However, these methods are less practical in scenarios where auxiliary data is unavailable. Data-free approaches can be further categorized into weight-space and singular vector-based methods. Weight-space methods (Yu et al., 2024; Yadav et al., 2023; He et al., 2024) focus on localizing and pruning conflicting parameters to reduce incompatibilities. Recent SVD-based works have reported that cross-task alignment in spectral directions can correlate with degraded merging performance (Gargiulo et al., 2025; Skorobogat et al., 2025). Our contribution is to explain the mechanism behind this observation in a broad class of merging methods. We show that aligned shared compo-

nents are repeatedly aggregated within dominant spectral subspaces, leading to spectral over-counting and singular-value inflation in the merged spectrum. This links alignment to performance degradation, motivating subspace-wise singular-value calibration as a direct remedy.

3. Spectral View of Inter-Model Interference

This section answers a single question: why does merging multiple task updates hurt performance even when tasks appear aligned? Our goal is to isolate an interference source that is not explained by weight conflicts. To do so, we analyze merged updates in spectral space and track how shared components accumulate across tasks. We show that repeated accumulation can over-count shared directions and inflate singular values in a few dominant subspaces.

3.1. Preliminary

Given a set of K fine-tuned model parameter sets $\{\mathbf{W}_i\}_{i=1}^K$, each obtained by fine-tuning the pre-trained parameter $\mathbf{W}_{\text{pre}}$. Model merging aims to construct a merged model $\mathbf{W}_{\text{merge}}$ that effectively inherits task-specific knowledge from all $\{\mathbf{W}_i\}$. Traditionally, a simple weight averaging is adopted: $\mathbf{W}_{\text{merge}} = \frac{1}{K}\sum_{i=1}^K \mathbf{W}_i$. Building on this, Task Arithmetic (TA) (Ilharco et al., 2022) rewrites merging in terms of task updates. For a given layer, the task matrix for task i is $\Delta\mathbf{W}_i = \mathbf{W}_i - \mathbf{W}_{\text{pre}}$. A merging method then combines $\{\Delta\mathbf{W}_i\}_{i=1}^K$ to produce a merged task matrix $\Delta\mathbf{W}_{\text{merge}}$. The final merged model is obtained by adding the merged update back to the pre-trained weights, with a global scaling λ:

$$\mathbf{W}_{\text{merge}} = \mathbf{W}_{\text{pre}} + \lambda\Delta\mathbf{W}_{\text{merge}}. \tag{1}$$

3.2. Projections Reveal Spectral Over-Counting

To investigate how different tasks interact after merging, we test whether the merged update preserves each task's contribution in a balanced way by projecting each task matrix onto a shared space induced by the merged matrix.

Projecting tasks onto shared space. Consider a merged task matrix $\Delta\mathbf{W}_{\text{merge}} = \sum_{i=1}^K \Delta\mathbf{W}_i$ and its Singular Value Decomposition (SVD)

$$\Delta\mathbf{W}_{\text{merge}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top, \qquad \mathbf{\Sigma} = \text{diag}(\sigma^1, \ldots, \sigma^R), \tag{2}$$

where $\boldsymbol{u}^r$ and $\boldsymbol{v}^r$ are the r-th left and right singular vectors. We use the index r to label a spectral subspace in matrix space, *i.e.*, the subspace spanned by the $\boldsymbol{u}^r(\boldsymbol{v}^r)^\top$, and use $\boldsymbol{u}^r$ as its column-space direction. Each task matrix $\Delta\mathbf{W}_i \in \mathbb{R}^{m\times n}$ can be written in row form:

$$\Delta\mathbf{W}_i = \begin{bmatrix} \boldsymbol{w}_1^\top \\ \boldsymbol{w}_2^\top \\ \vdots \\ \boldsymbol{w}_m^\top \end{bmatrix}, \qquad \boldsymbol{w}_k \in \mathbb{R}^n, \tag{3}$$

and its effect on input $\boldsymbol{x} \in \mathbb{R}^n$ is captured by the inner products $\boldsymbol{w}_k^\top\boldsymbol{x}$. To analyze their interactions along column-space directions, we project each task matrix onto the left singular vector $\boldsymbol{u}^r$ using left-multiplication:

$$\boldsymbol{a}_i^r := (\boldsymbol{u}^r)^\top\Delta\mathbf{W}_i = \sum_{k=1}^m u_k^r\boldsymbol{w}_k^\top \in \mathbb{R}^n. \tag{4}$$

This produces a weighted combination of the rows of $\Delta\mathbf{W}_i$, yielding a **subspace response $\boldsymbol{a}_i^r$ that captures the task's effect along the shared column-space direction $\boldsymbol{u}^r$**. Thus, cross-task interactions in subspace r can be reduced to simple inner products among $\{\boldsymbol{a}_i^r\}$.

In contrast, when we multiply $\Delta\mathbf{W}_i$ by a vector like $\boldsymbol{v}$ from the right, we collapse the input space to just one direction:

$$\Delta\mathbf{W}_i\boldsymbol{v} = \begin{bmatrix} \boldsymbol{w}_1^\top\boldsymbol{v} \\ \vdots \\ \boldsymbol{w}_m^\top\boldsymbol{v} \end{bmatrix} \in \mathbb{R}^m. \tag{5}$$

This operation only tells us how the task behaves in that specific input direction, losing information about how the task interacts with all possible inputs. On the other hand, when we multiply by the left singular vector $\boldsymbol{u}^r$, we maintain a more comprehensive view of the task's behavior across all input directions, independent of the data.

Remark 3.1 (Layer-wise linear view). Our analysis treats the task matrix as a local linear operator within a layer (or block) and studies how task matrices mix along column-space directions.

Interference as directional projection mismatch. Having projected task updates onto the shared column-space basis, we can now quantify inter-task interference within each spectral subspace. Specifically, we form the merged response in subspace r by summing the corresponding subspace responses $\boldsymbol{a}_i^r$:

$$\boldsymbol{a}_{\text{merge}}^r := (\boldsymbol{u}^r)^\top\Delta\mathbf{W}_{\text{merge}} = \sum_{i=1}^K \boldsymbol{a}_i^r. \tag{6}$$

Ideally, if the merged task matrix fully preserved task i's capability in subspace r, then the component of the merged response $\boldsymbol{a}_{\text{merge}}^r$ along $\boldsymbol{a}_i^r$ should match $\boldsymbol{a}_i^r$ in magnitude. We therefore measure the interference $\mathcal{I}_i^r$ by the mismatch between task i's subspace response $\boldsymbol{a}_i^r$ and the projection of the merged response $\boldsymbol{a}_{\text{merge}}^r$ onto $\boldsymbol{a}_i^r$:

$$\mathcal{I}_i^r := \left\|\text{Proj}_{\boldsymbol{a}_i^r}(\boldsymbol{a}_{\text{merge}}^r) - \boldsymbol{a}_i^r\right\|_2^2 \geq 0, \tag{7}$$

where

$$\text{Proj}_{\boldsymbol{a}_i^r}(\boldsymbol{a}_{\text{merge}}^r) \triangleq s_i^r\boldsymbol{a}_i^r = \frac{\langle\boldsymbol{a}_{\text{merge}}^r, \boldsymbol{a}_i^r\rangle}{\|\boldsymbol{a}_i^r\|_2^2}\boldsymbol{a}_i^r. \tag{8}$$

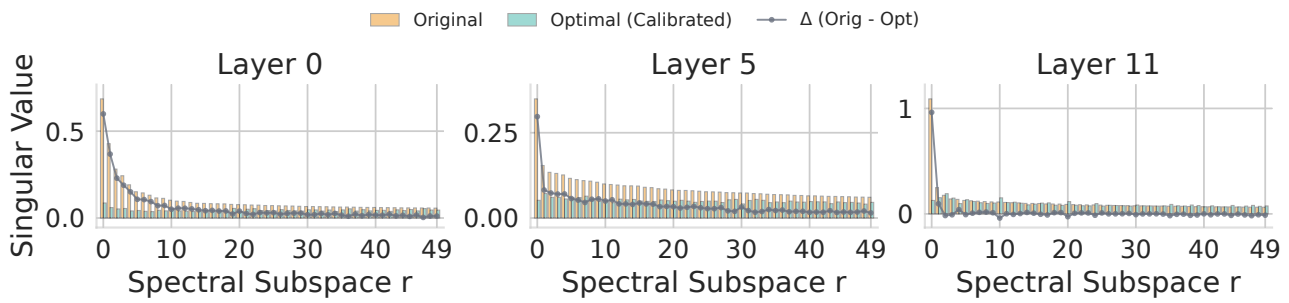

Figure 2. **Discrepancy between original and calibrated singular values.** For weight-space addition, we compare the original singular values σ from $\mathrm{SVD}(\Delta\mathbf{W}_{\mathrm{merge}})$ with the calibrated values $\sigma^\star$, where $\sigma^\star$ is obtained by first computing the task-wise optimal scalings $(\gamma_i^r)^\star$ from Eq. (13) and then averaging them across tasks within each subspace. A clear gap $\Delta = \sigma - \sigma^\star$ appears in top spectral subspaces, indicating systematic spectral over-counting and singular-value inflation.

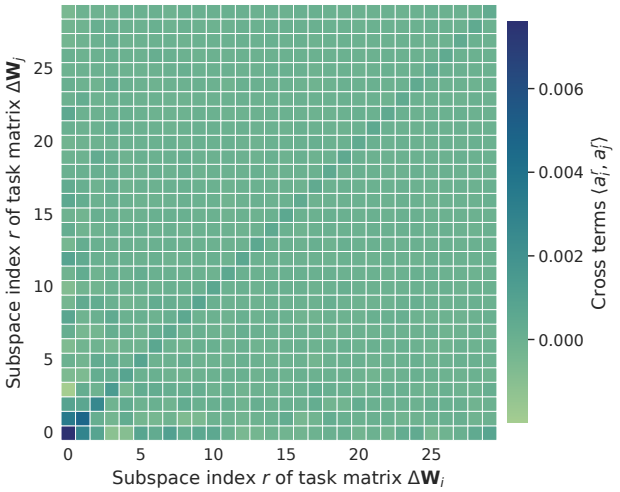

Figure 3. **Cross terms concentrate in top spectral subspaces.** We visualize $\langle \boldsymbol{a}_i^r, \boldsymbol{a}_j^r \rangle$ across tasks for small r, showing predominantly positive overlap that induces over-counting.

The projection coefficient s_i^r quantifies how the merged response scales task i along $\boldsymbol{a}_i^r$ in subspace r: $s_i^r > 1$ indicates amplification, while $s_i^r < 1$ indicates attenuation (or conflict). Consequently, $\mathcal{I}_i^r = 0$ holds if and only if $\mathrm{Proj}_{\boldsymbol{a}_i^r}(\boldsymbol{a}_{\mathrm{merge}}^r) = \boldsymbol{a}_i^r$, equivalently $s_i^r = 1$. Building on this projection coefficient, we derive the following lemma:

Lemma 3.2 (Cross-term form of projection interference)**.** *Assume $\Delta\mathbf{W}_{\mathrm{merge}} = \sum_{k=1}^K \Delta\mathbf{W}_k$. Fix any task i and subspace r, and assume $\|\boldsymbol{a}_i^r\|_2^2 > 0$. Then*

$$s_i^r = 1 + \sum_{j \neq i} \frac{\langle \boldsymbol{a}_j^r, \boldsymbol{a}_i^r \rangle}{\|\boldsymbol{a}_i^r\|_2^2}, \qquad \mathcal{I}_i^r = (s_i^r - 1)^2 \, \|\boldsymbol{a}_i^r\|_2^2. \quad (9)$$

Lemma 3.2 makes the source of projection mismatch explicit. **Projection mismatch is governed by the cross-task inner products $\langle \boldsymbol{a}_j^r, \boldsymbol{a}_i^r \rangle$, which quantifies how strongly other tasks contribute along task i's response direction in subspace r.** When many $\langle \boldsymbol{a}_j^r, \boldsymbol{a}_i^r \rangle$ terms are positive, we obtain $s_i^r > 1$ and thus $\mathcal{I}_i^r > 0$, meaning that multiple tasks jointly accumulate along the same direction $\boldsymbol{a}_i^r$ and the merged response over-counts this shared component. This concentration of cross-task overlap is visible in Fig. 3, where large overlaps cluster in the top spectral subspaces.

From interference to singular-value inflation. The projection mismatch above is a behavioral symptom; to correct it, we next trace its impact back to the parameters.

By Eq. (2), the merged response in subspace r satisfies $\boldsymbol{a}_{\mathrm{merge}}^r = \sigma^r (\boldsymbol{v}^r)^\top$, hence the singular value is exactly the response magnitude:

$$\sigma^r \;=\; \left\|\boldsymbol{a}_{\mathrm{merge}}^r\right\|_2. \quad (10)$$

Thus, once we characterize how projection mismatch changes the magnitude of $\boldsymbol{a}_{\mathrm{merge}}^r$, we can directly translate it into a statement about the singular value σ^r. This connection yields the following theorem.

Theorem 3.3 (Projection-optimal calibration and singular–value inflation)**.** *Fix a target task i and a subspace r, and assume $\|\boldsymbol{a}_i^r\|_2^2 > 0$. Let $\boldsymbol{a}_{\mathrm{merge}}^r$ denote the merged response in subspace r. Consider the calibration problem*

$$\min_{\gamma^r \geq 0} \left\| \mathbf{Proj}_{\boldsymbol{a}_i^r}\!\left(\gamma^r \boldsymbol{a}_{\mathrm{merge}}^r\right) - \boldsymbol{a}_i^r \right\|_2^2. \quad (11)$$

Define s_i^r as in Eq. (8). If $s_i^r > 0$, then the optimal calibration has the closed form

$$(\gamma_i^r)^\star \;=\; \frac{1}{s_i^r} \;=\; \frac{\|\boldsymbol{a}_i^r\|_2^2}{\langle \boldsymbol{a}_{\mathrm{merge}}^r, \boldsymbol{a}_i^r \rangle}, \quad (12)$$

whereas if $s_i^r \leq 0$, the optimum is attained at the boundary $(\gamma_i^r)^\star = 0$. If $s_i^r > 0$, the corresponding projection-optimal singular value for best emulating task i in subspace r is

$$(\sigma_i^r)^\star = (\gamma_i^r)^\star \, \sigma^r. \quad (13)$$

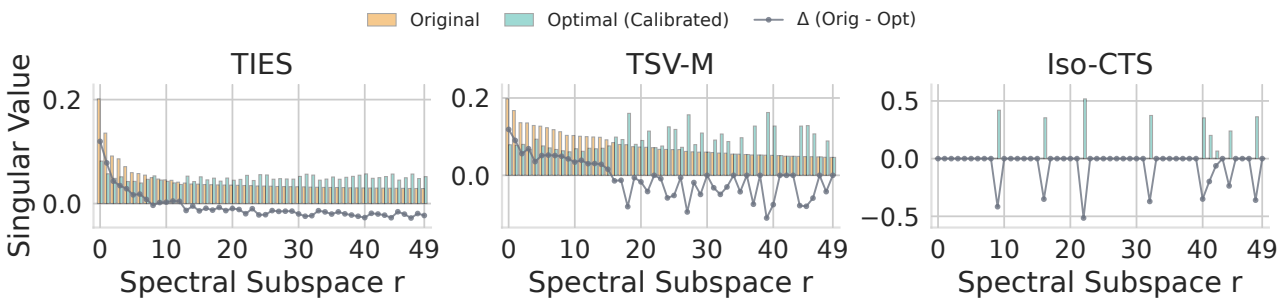

Figure 4. **Generality of the singular-value gap.** We compare the original singular values σ with the calibrated values $\sigma^\star$, where $\sigma^\star$ applies the subspace-wise average of the task-wise optimal scalings $\gamma_i^{r\star}$ from Eq. (13).The gap between σ and $\sigma^\star$ persists across representative merging methods, indicating that singular-value inflation and overly small singular values coexist.

Under weight addition $\Delta\mathbf{W}_{\text{merge}} = \sum_{k=1}^{K} \Delta\mathbf{W}_k$*, if*

$$\sum_{j\neq i} \langle \boldsymbol{a}_j^r, \boldsymbol{a}_i^r \rangle \;>\; 0, \tag{14}$$

then $s_i^r > 1$ *and thus* $(\gamma_i^r)^\star < 1$*. Equivalently,*

$$\sigma^r \;>\; (\sigma_i^r)^\star, \tag{15}$$

showing that positive cross-task overlap inflates the merged singular value above the projection-optimal magnitude.

The condition $\langle \boldsymbol{a}_j^r, \boldsymbol{a}_i^r \rangle > 0$ implies that other tasks contribute constructively along task i's direction in subspace r, increasing $\langle \boldsymbol{a}_{\text{merge}}^r, \boldsymbol{a}_i^r \rangle$ and forcing $(\gamma_i^r)^\star < 1$ to match $\boldsymbol{a}_i^r$ under projection. **In particular, whenever $\sum_{j\neq i}\langle \boldsymbol{a}_j^r, \boldsymbol{a}_i^r \rangle > 0$, the merged response exhibits constructive accumulation, and thus the singular value σ^r is inflated above the projection-induced optimal singular values $(\sigma_i^r)^\star$.** Such accumulation of singular values ultimately leads to a decline in model performance. Empirically, such positive overlap is most pronounced in leading subspaces (Fig. 3), yielding amplified top singular values in the merged spectrum (Fig. 2).

3.3. Generality of Singular-Value Over-Accumulation

So far, we have used weight addition to make the mechanism transparent. We next ask whether the same failure mode appears in other merging methods.

We answer this by comparing the original singular values σ with the calibrated values $\sigma^\star$. Fig. 4 shows that the gap $\Delta = \sigma - \sigma^\star$ persists across representative methods. In many cases, the largest positive gaps still concentrate in the leading spectral subspaces, indicating over-accumulation of shared directions. At the same time, some methods exhibit large variance across the spectrum and can even yield negative gaps in certain subspaces, which corresponds to under-accumulation. Taken together, these observations suggest that most practical merging methods can simultaneously inflate dominant subspaces while shrinking others, motivating a calibration step that acts at the subspace level.

4. Methodology

To correct spectral over-counting in a merged update without additional data or optimization, we introduce **Singular Value Calibration (SVC)**, which post-processes a merged task matrix by adjusting its singular values while keeping the directions unchanged.

Given a pre-trained model $\mathbf{W}_{\text{pre}}$ and K fine-tuned models $\{\mathbf{W}_i\}_{i=1}^K$, we form task matrices:

$$\Delta\mathbf{W}_i = \mathbf{W}_i - \mathbf{W}_{\text{pre}}. \tag{16}$$

Let $\Delta\mathbf{W}_{\text{merge}}$ be the merged task matrix produced by a base merging method (for example, summation, averaging, or masking). SVC takes $\Delta\mathbf{W}_{\text{merge}}$ as input, estimates how much each spectral subspace is over-counted, and then rescales the corresponding singular values accordingly.

Step 1: Merged column-space basis. We first choose a shared coordinate system so that different tasks can be compared within the same subspaces. To do so, we compute the SVD of the merged task matrix

$$\Delta\mathbf{W}_{\text{merge}} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top = \sum_{r=1}^{R} \sigma^r \, \boldsymbol{u}^r (\boldsymbol{v}^r)^\top. \tag{17}$$

We use the left singular vectors $\{\boldsymbol{u}^r\}$ as the merged column-space basis. This aligns with Section 3.2, where subspace-wise interactions are defined by projecting task matrices onto shared column-space directions.

Step 2: Subspace-wise overlap from projections. With this basis fixed, we next quantify how much each subspace is over-counted after merging. For each subspace r and each task i, we compute the task response and the merged

response along $\boldsymbol{u}^r$:

$$\boldsymbol{a}_i^r \;=\; (\boldsymbol{u}^r)^\top \Delta \mathbf{W}_i \in \mathbb{R}^n,\; \boldsymbol{a}_{\text{merge}}^r \;=\; (\boldsymbol{u}^r)^\top \Delta \mathbf{W}_{\text{merge}}. \tag{18}$$

We then measure how the merged response scales task i along its own direction using the projection coefficient from Theorem 3.3:

$$s_i^r \;=\; \frac{\langle \boldsymbol{a}_{\text{merge}}^r, \boldsymbol{a}_i^r \rangle}{\|\boldsymbol{a}_i^r\|_2^2}. \tag{19}$$

If multiple tasks contribute constructively in the same subspace, then $s_i^r > 1$, indicating over-counting.

To produce a single correction per subspace, we aggregate these coefficients across tasks into a calibration factor

$$\gamma^r \;=\; K / \sum_{i=1}^{K} \max(\alpha, s_i^r). \tag{20}$$

Equivalently, γ^r can be viewed as the harmonic mean of the clipped task-wise scalings $\{1/\max(\alpha, s_i^r)\}_{i=1}^K$. This choice is conservative: it down-weights subspaces primarily when many tasks exhibit large s_i^r (strong over-counting), while $\alpha \in (0, 1]$ prevents unstable behavior when some s_i^r are very small. In practice, $\gamma^r \approx 1$ indicates little systematic over-counting in subspace r, while $\gamma^r < 1$ indicates singular-value inflation.

Step 3: Singular-value calibration and reconstruction. By Eq. (10), scaling $\boldsymbol{a}_{\text{merge}}^r$ is equivalent to scaling σ^r. Thus, singular-value inflation is corrected by rescaling each singular value based on the subspace-wise overlap degree.

$$\tilde{\sigma}^r = \gamma^r \, \sigma^r, \tag{21}$$

With α applied inside γ^r, calibration is suppression-only when $\alpha = 1$, since $\max(\alpha, s_i^r) \geq 1$ makes $\gamma^r \leq 1$ for all r. Finally, we reconstruct the calibrated merged task matrix

$$\Delta \tilde{\mathbf{W}}_{\text{merge}} \;=\; \sum_{r=1}^{R} \tilde{\sigma}^r \, \boldsymbol{u}^r (\boldsymbol{v}^r)^\top, \tag{22}$$

and output the final merged weights

$$\mathbf{W}_{\text{merge}} \;=\; \mathbf{W}_{\text{pre}} + \Delta \tilde{\mathbf{W}}_{\text{merge}}. \tag{23}$$

Algorithm 1 Subspace-Consistency Spectral Calibration

Input: $\mathbf{W}_{\text{pre}}$, $\{\mathbf{W}_i\}_{i=1}^K$, $\Delta \mathbf{W}_{\text{merge}}$, α.
Output: $\mathbf{W}_{\text{merge}}$
$\Delta \mathbf{W}_i \leftarrow \mathbf{W}_i - \mathbf{W}_{\text{pre}}$ for $i = 1, \ldots, K$
$\Delta \mathbf{W}_{\text{merge}} = \mathbf{U} \mathbf{\Sigma} \mathbf{V}^\top$ (SVD, where $\boldsymbol{u}^r$ is the r-th column of $\mathbf{U}$ and σ^r is the r-th diagonal entry of $\mathbf{\Sigma}$)
for $r = 1$ **to** R **do**
  $\boldsymbol{a}_i^r \leftarrow (\boldsymbol{u}^r)^\top \Delta \mathbf{W}_i$ for all i
  $\boldsymbol{a}_{\text{merge}}^r \leftarrow (\boldsymbol{u}^r)^\top \Delta \mathbf{W}_{\text{merge}}$
  $s_i^r \leftarrow \dfrac{\langle \boldsymbol{a}_{\text{merge}}^r, \boldsymbol{a}_i^r \rangle}{\|\boldsymbol{a}_i^r\|_2^2}$ for all i
  $\gamma^r \leftarrow K / \sum_{i=1}^K \max(\alpha, s_i^r)$
  $\tilde{\sigma}^r \leftarrow \gamma^r \, \sigma^r$
end for
$\Delta \tilde{\mathbf{W}}_{\text{merge}} \leftarrow \mathbf{U} \, \text{diag}(\tilde{\sigma}) \, \mathbf{V}^\top$
$\mathbf{W}_{\text{merge}} \leftarrow \mathbf{W}_{\text{pre}} + \Delta \tilde{\mathbf{W}}_{\text{merge}}$

5. Experiments

5.1. Experimental Protocol

Baselines and Datasets. We evaluate SVC against representative training-free model merging baselines, including TA (Ilharco et al., 2022), TIES (Yadav et al., 2023), DARE (Yu et al., 2024), TSV-M (Gargiulo et al., 2025), and Iso-CTS (Marczak et al., 2025a). All reported results are produced by our own runs under a unified evaluation protocol. Since the checkpoints used in our study may differ from those in the original papers, absolute numbers can vary from previously reported results. For reference, recent training-free methods often match or exceed training-based approaches such as AdaMerging++ (Yang et al., 2024c) and Surgery (Yang et al., 2024b).

Following common practice (Ilharco et al., 2022; Yang et al., 2024c), we report computer vision (CV) results on 8 multitask classification benchmarks. For natural language processing (NLP), we evaluate on 11 classification benchmarks and additionally report performance on two open LLM leaderboards. Full benchmark lists and evaluation details are provided in the Appendix.

Implementation Details. We use the ViT-B/32 CLIP as the default visual encoder, consistent with the setup in (Yang et al., 2024c). The hyperparameters used in previous methods remain identical to those specified in their original papers. The hyperparameters introduced in this paper are set to $\alpha = 1/K$ by default, where K is the number of tasks. However, for TSV-M w/ SVC, we set $\alpha = 1$ by default (the most stable, suppression-only setting), since TSV-M appears to overestimate the correction coefficients.

5.2. Vision & Language: Main Results

Computer Vision (CV) Experiments. Following prior work (Marczak et al., 2025a), we evaluate average classification accuracy across 8 and 14 datasets, as shown in Tab. 1 (details are deferred to Appendix A). Our SVC method consistently improves SOTA results in diverse merging tasks. Notably, without altering the directions of singular vectors, SVC achieves a 19% improvement over Task Arithmetic.

Natural Language Processing (NLP) Experiments. We

Table 1. **Consolidated average accuracy (%) across CV benchmarks.** Per-dataset results are deferred to the Appendix. Due to the use of different checkpoints, certain methods (*e.g.*, Iso-C and Iso-CTS) differ from those reported in the original paper.

Method	8 Tasks			14 Tasks		
	ViT-B/32	ViT-B/16	ViT-L/14	ViT-B/32	ViT-B/16	ViT-L/14
Reference (non-merging)						
Pretrained	48.0	55.2	64.9	56.6	61.7	70.4
Individual	90.5	93.0	94.4	87.3	89.5	91.4
Training-free merging						
TA (Ilharco et al., 2022)	68.9	73.7	84.3	46.4	57.1	57.7
w/ SVC (Ours)	81.9 (+13.0 ↑)	86.2 (+12.5 ↑)	91.3 (+7.0 ↑)	63.1 (+16.7 ↑)	72.0 (+14.9 ↑)	76.7 (+19.0 ↑)
TIES (Yadav et al., 2023)	72.6	76.6	85.6	61.6	60.1	62.4
w/ SVC (Ours)	80.0 (+7.4 ↑)	84.8 (+8.2 ↑)	90.6 (+5.0 ↑)	62.3 (+0.7 ↑)	63.9 (+3.8 ↑)	63.6 (+1.2 ↑)
DARE (Yu et al., 2024)	65.8	71.5	79.4	63.9	67.0	75.4
w/ SVC (Ours)	80.7 (+14.9 ↑)	84.8 (+13.3 ↑)	90.1 (+10.7 ↑)	71.7 (+7.8 ↑)	70.0 (+3.0 ↑)	77.9 (+2.5 ↑)
TSV-M (Gargiulo et al., 2025)	84.0	87.3	91.5	76.3	76.6	82.3
w/ SVC (Ours)	84.8 (+0.8 ↑)	88.0 (+0.7 ↑)	91.8 (+0.3 ↑)	76.8 (+0.5 ↑)	77.0 (+0.4 ↑)	83.1 (+0.8 ↑)
Iso-C (Marczak et al., 2025a)	83.1	87.5	91.5	73.4	69.6	76.8
w/ SVC (Ours)	84.6 (+1.5 ↑)	88.5 (+1.0 ↑)	92.2 (+0.7 ↑)	74.0 (+0.6 ↑)	71.8 (+2.2 ↑)	78.5 (+1.7 ↑)
Iso-CTS (Marczak et al., 2025a)	81.4	86.9	90.9	76.7	77.6	85.7
w/ SVC (Ours)	85.6 (+4.2 ↑)	89.7 (+2.8 ↑)	92.9 (+2.0 ↑)	76.7 (+0.0 ↑)	78.5 (+0.9 ↑)	85.9 (+0.2 ↑)

Table 2. **Consolidated performance across NLP benchmarks.** Details are deferred to the Appendix. Llama2 is evaluated on two generation benchmarks; others are used as encoders for classification.

Method	Generative Evaluation		Encoder-derived Classification		
	Llama2-7B (FT)		BERT (FT)	T5 (FT)	T0 (PEFT)
	AlpacaEval↑	GSM8K↑	Avg Acc (%)	Avg Acc (%)	Avg Acc (%)
TA (Ilharco et al., 2022)	49.1	46.1	56.9	41.5	53.5
w/ SVC (Ours)	51.7 (+2.5 ↑)	52.2 (+6.1 ↑)	69.0 (+12.1 ↑)	46.3 (+4.8 ↑)	65.8 (+12.3 ↑)
TIES (Yadav et al., 2023)	47.8	45.0	59.7	45.5	54.1
w/ SVC (Ours)	49.2 (+1.3 ↑)	48.1 (+3.0 ↑)	61.3 (+1.6 ↑)	49.7 (+4.2 ↑)	54.1(+0.0 ↑)
DARE (Yu et al., 2024)	46.5	46.1	57.6	41.2	53.3
w/ SVC (Ours)	52.8 (+6.3 ↑)	51.4 (+5.3 ↑)	57.9 (+0.3 ↑)	46.2 (+5.0 ↑)	54.7 (+1.4 ↑)
TSV-M (Gargiulo et al., 2025)	41.7	51.9	60.6	46.5	—
w/ SVC (Ours)	47.3 (+5.6 ↑)	51.9 (+0.0 ↑)	61.3 (+0.8 ↑)	46.6 (+0.1 ↑)	—
Iso-C (Marczak et al., 2025a)	50.0	42.0	56.3	43.9	—
w/ SVC (Ours)	58.9 (+8.9 ↑)	51.4 (+9.4 ↑)	56.6 (+0.3 ↑)	48.9 (+5.0 ↑)	—
Iso-CTS (Marczak et al., 2025a)	43.8	38.7	56.3	38.8	—
w/ SVC (Ours)	51.3 (+7.5 ↑)	48.0 (+9.3 ↑)	56.5 (+0.2 ↑)	46.3 (+7.5 ↑)	—

evaluate our method across NLP models of varying sizes in Table 2 (details are deferred to Appendix A). SVC achieves SOTA performance on both conventional small language models and recent large language models (LLMs). For T0, we follow Yu et al. (2024) and adopt IA^3-based parameter-efficient fine-tuning (PEFT). Because **IA^3 yields task vectors as vectors rather than full weight matrices**, an SVD is not defined; consequently, SVD-dependent approaches (*e.g.*, TSV-M, Iso-C) are not applicable, denoted as —.

5.3. Empirical Analysis and Ablation Studies

Ablation Study. We study the role of the hyperparameter α in SVC. In our calibration rule, $\alpha \in (0, 1]$ controls how aggressively we down-weight over-counted directions by entering the aggregation $\gamma^r = K/\sum_{i=1}^{K} \max(\alpha, s_i^r)$. When $\alpha = 1$, SVC becomes a suppression-only variant and focuses on correcting singular-value inflation caused by spectral over-counting. Smaller α reduces the floor on $\max(\alpha, s_i^r)$ and can yield $\gamma^r > 1$, which amplifies subspaces that are under-accumulated.

Fig. 5 shows that $\alpha = 1$ yields consistent gains, supporting

our main contributor. In contrast, allowing additional amplification by decreasing α can produce mixed outcomes, since boosting subspaces may disturb the spectral balance.

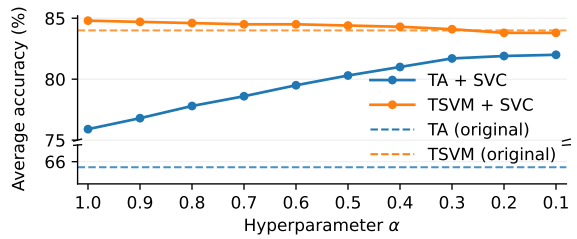

Figure 5. **Effect of hyperparameter α in SVC.** When only suppressing over-counting ($\alpha = 1$), SVC yields a stable improvement. In contrast, additionally boosting singular values ($\alpha \in (0, 1)$) requires caution and can degrade performance as α decreases.

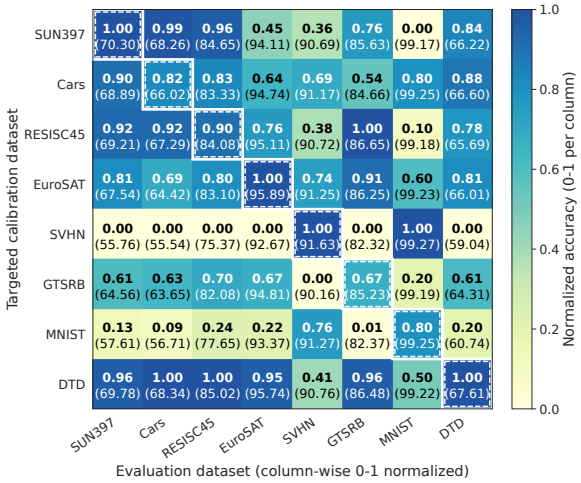

Figure 6. **Targeted calibration for a single task.** Each cell (i, j) shows performance on task j after calibrating the merged model using task i as the target. Values are normalized within each column, and the numbers in parentheses show the accuracy.

Preference Optimisation. SVC can favor a specific task after merging. We compute the calibration strength using a chosen target task (referred to as preference optimization). Concretely, we use the target task's s_i^r and set $\gamma^r = 1/\max(\alpha, s_i^r)$, rather than aggregating across tasks.

Fig. 6 summarizes the effect. Each cell (i, j) calibrates the merge for target task i and evaluates on task j. Diagonal entries are usually the largest, indicating that targeting task i primarily improves task i. At the same time, tasks that share related features can benefit from the same calibration, while tasks with large domain gaps may degrade, *e.g.*, calibrating for `Cars` improves `SUN397`.

Column-space vs. Row-space Calibration. SVC measures subspace-wise overlap in the merged column-space basis, following our projection analysis in Section 3.2. To test whether the choice of singular-vector side matters, we construct a row-space variant that replaces the left singular vectors with the right singular vectors when computing subspace overlap, while keeping all other settings unchanged. Table 3 shows that this row-space variant is far less reliable. It often removes the gains brought by SVC and can even reduce performance below the uncalibrated baseline (for example, `TIES`). This gap is consistent with our analysis. Right singular vectors describe input-side directions, so their overlap reflects how task matrices align with specific input patterns. Overall, the results support column-space overlap as the appropriate quantity for singular-value calibration.

Table 3. **Ablation on singular vector side.** Replacing left singular vectors with right singular vectors for overlap measurement and calibration significantly degrades performance, highlighting the necessity of column-space calibration.

Method	original	SVC (col, ours)	SVC (row)
TA	68.9	81.9 (+13.0 ↑)	64.9 (-4.0 ↓)
TIES	72.6	80.0 (+7.4 ↑)	65.7 (-6.9 ↓)
DARE	65.8	80.7 (+14.9 ↑)	67.5 (+1.7 ↑)
TSV-M	84.0	84.8 (+0.8 ↑)	84.0 (+0.0 ↑)
Iso-C	83.1	84.6 (+1.5 ↑)	82.1 (-1.0 ↓)
Iso-CTS	81.4	85.6 (+4.2 ↑)	85.5 (+4.1 ↑)

Table 4. **Runtime and memory footprint overhead of SVC.** SVC applies SVD and runs once offline, without any training.

Backbone	Time Cost	Memory Usage
ViT-B/32	5.1 s	1,027.4 MiB
ViT-B/16	8.2 s	1,082.8 MiB
ViT-L/14	15.6 s	1,488.5 MiB
LLaMA2 7B	517.2 s	1,898.7 MiB

Cost Analysis. Consistent with TSV-M and ISO-CTS, our method applies SVD, which introduces additional cost. Even on 7B-parameter LLMs, this overhead is acceptable. Overall, it is far cheaper than training-based methods, because it requires no gradient computation. Table 4 summarizes SVC's runtime and memory across backbones.

6. Conclusion

We show that model merging can fail due to spectral overcounting. Using projections onto the merged column-space basis, we find that shared directions are primarily concentrated in top spectral subspaces, which can lead to singular-value inflation. We propose Singular Value Calibration (SVC). SVC measures subspace-wise overlap from these projections and rescales the corresponding singular values, while keeping the spectral directions fixed. Across vision and language benchmarks, SVC consistently improves merging methods and achieves SOTA results.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

References

MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges. https://yann.lecun.com/exdb/mnist/.

Aiello, E., Yu, L., Nie, Y., Aghajanyan, A., and Oguz, B. Jointly Training Large Autoregressive Multimodal Models, September 2023.

Almeida, T. A., Hidalgo, J. M. G., and Yamakami, A. Contributions to the study of SMS spam filtering: New collection and results. In Proceedings of the 11th ACM Symposium on Document Engineering, DocEng '11, pp. 259–262, New York, NY, USA, September 2011. Association for Computing Machinery. ISBN 978-1-4503-0863-2. doi: 10.1145/2034691.2034742.

Bossard, L., Guillaumin, M., and Van Gool, L. Food-101 – Mining Discriminative Components with Random Forests. In Fleet, D., Pajdla, T., Schiele, B., and Tuytelaars, T. (eds.), Computer Vision – ECCV 2014, volume 8694, pp. 446–461, Cham, 2014. Springer International Publishing. ISBN 978-3-319-10598-7 978-3-319-10599-4. doi: 10.1007/978-3-319-10599-4_29.

Cheng, G., Han, J., and Lu, X. Remote Sensing Image Scene Classification: Benchmark and State of the Art. Proceedings of the IEEE, 105(10):1865–1883, October 2017. ISSN 1558-2256. doi: 10.1109/JPROC.2017.2675998.

Chitale, R., Vaidya, A., Kane, A., and Ghotkar, A. Task Arithmetic with LoRA for Continual Learning, November 2023.

Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., and Vedaldi, A. Describing Textures in the Wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3606–3613, 2014.

Clanuwat, T., Bober-Irizar, M., Kitamoto, A., Lamb, A., Yamamoto, K., and Ha, D. Deep Learning for Classical Japanese Literature, December 2018.

Coates, A., Ng, A., and Lee, H. An Analysis of Single-Layer Networks in Unsupervised Feature Learning. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics, pp. 215–223. JMLR Workshop and Conference Proceedings, June 2011.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. arXiv preprint arXiv:2110.14168, 2021.

Cohen, G., Afshar, S., Tapson, J., and van Schaik, A. EMNIST: Extending MNIST to handwritten letters. In 2017 International Joint Conference on Neural Networks (IJCNN), pp. 2921–2926, May 2017. doi: 10.1109/IJCNN.2017.7966217.

Dekoninck, J., Fischer, M., Beurer-Kellner, L., and Vechev, M. Controlled Text Generation via Language Model Arithmetic, March 2024.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers), pp. 4171–4186, 2019.

Dong, X., Yu, Z., Cao, W., Shi, Y., and Ma, Q. A survey on ensemble learning. Frontiers of Computer Science, 14 (2):241–258, April 2020. ISSN 2095-2236. doi: 10.1007/s11704-019-8208-z.

Du, G., Lee, J., Li, J., Jiang, R., Guo, Y., Yu, S., Liu, H., Goh, S. K., Tang, H.-K., He, D., et al. Parameter competition balancing for model merging. Advances in Neural Information Processing Systems, 37:84746–84776, 2024.

Du, G., Fang, Z., Li, J., Li, J., Jiang, R., Yu, S., Guo, Y., Chen, Y., Goh, S. K., Tang, H.-K., et al. Neural parameter search for slimmer fine-tuned models and better transfer. arXiv preprint arXiv:2505.18713, 2025.

Fei-Fei, L., Fergus, R., and Perona, P. Learning Generative Visual Models from Few Training Examples: An Incremental Bayesian Approach Tested on 101 Object Categories. In 2004 Conference on Computer Vision and Pattern Recognition Workshop, pp. 178–178, June 2004. doi: 10.1109/CVPR.2004.383.

Gargiulo, A. A., Crisostomi, D., Bucarelli, M. S., Scardapane, S., Silvestri, F., and Rodolà, E. Task Singular Vectors: Reducing Task Interference in Model Merging, April 2025.

Goddard, C., Siriwardhana, S., Ehghaghi, M., Meyers, L., Karpukhin, V., Benedict, B., McQuade, M., and Solawetz, J. Arcee's MergeKit: A Toolkit for Merging Large Language Models, March 2024.

Goodfellow, I. J., Erhan, D., Carrier, P. L., Courville, A., Mirza, M., Hamner, B., Cukierski, W., Tang, Y., Thaler, D., Lee, D.-H., Zhou, Y., Ramaiah, C., Feng, F., Li, R., Wang, X., Athanasakis, D., Shawe-Taylor, J., Milakov, M., Park, J., Ionescu, R., Popescu, M., Grozea, C., Bergstra, J., Xie, J., Romaszko, L., Xu, B., Chuang, Z., and Bengio, Y. Challenges in Representation Learning: A Report on Three Machine Learning Contests. In Lee, M., Hirose, A., Hou, Z.-G., and Kil, R. M. (eds.), Neural Information Processing, pp. 117–124, Berlin, Heidelberg, 2013. Springer. ISBN 978-3-642-42051-1. doi: 10.1007/978-3-642-42051-1_16.

He, Y., Hu, Y., Lin, Y., Zhang, T., and Zhao, H. Localize-and-Stitch: Efficient Model Merging via Sparse Task Arithmetic, August 2024.

Helber, P., Bischke, B., Dengel, A., and Borth, D. 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, July 2019. ISSN 2151-1535. doi: 10.1109/JSTARS.2019.2918242.

Huang, C., Ye, P., Chen, T., He, T., Yue, X., and Ouyang, W. EMR-Merging: Tuning-Free High-Performance Model Merging, May 2024.

Ilharco, G., Ribeiro, M. T., Wortsman, M., Schmidt, L., Hajishirzi, H., and Farhadi, A. Editing models with task arithmetic. In The Eleventh International Conference on Learning Representations, September 2022.

Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., and Wilson, A. G. Averaging Weights Leads to Wider Optima and Better Generalization, February 2019.

Jin, X., Ren, X., Preotiuc-Pietro, D., and Cheng, P. Dataless Knowledge Fusion by Merging Weights of Language Models, October 2023.

Krause, J., Stark, M., Deng, J., and Fei-Fei, L. 3D Object Representations for Fine-Grained Categorization. In Proceedings of the IEEE International Conference on Computer Vision Workshops, pp. 554–561, 2013.

Krizhevsky, A. Learning Multiple Layers of Features from Tiny Images.

Lee, Y.-A., Ko, C.-Y., Pedapati, T., Chung, I.-H., Yeh, M.-Y., and Chen, P.-Y. Star: Spectral truncation and rescale for model merging. In Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers), pp. 496–505, 2025.

Li, X., Zhang, T., Dubois, Y., Taori, R., Gulrajani, I., Guestrin, C., Liang, P., and Hashimoto, T. B. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval, 5 2023.

Li, Y., Guo, J., Qi, L., Li, W., and Shi, Y. Text and image are mutually beneficial: Enhancing training-free few-shot classification with clip. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pp. 5039–5047, 2025.

Liu, H., Tam, D., Muqeeth, M., Mohta, J., Huang, T., Bansal, M., and Raffel, C. A. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. Advances in Neural Information Processing Systems, 35: 1950–1965, 2022.

Lu, Z., Fan, C., Wei, W., Qu, X., Chen, D., and Cheng, Y. Twin-Merging: Dynamic Integration of Modular Expertise in Model Merging, October 2024.

Luo, H., Sun, Q., Xu, C., Zhao, P., Lou, J., Tao, C., Geng, X., Lin, Q., Chen, S., and Zhang, D. Wizardmath: Empowering mathematical reasoning for large language models via reinforced evol-instruct. arXiv preprint arXiv:2308.09583, 2023.

Maji, S., Rahtu, E., Kannala, J., Blaschko, M., and Vedaldi, A. Fine-Grained Visual Classification of Aircraft, June 2013.

Marczak, D., Magistri, S., Cygert, S., Twardowski, B., Bagdanov, A. D., and van de Weijer, J. No Task Left Behind: Isotropic Model Merging with Common and Task-Specific Subspaces, February 2025a.

Marczak, D., Twardowski, B., Trzciński, T., and Cygert, S. MAGMAX: Leveraging Model Merging for Seamless Continual Learning. In Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., and Varol, G. (eds.), Computer Vision – ECCV 2024, pp. 379–395, Cham, 2025b. Springer Nature Switzerland. ISBN 978-3-031-73013-9. doi: 10.1007/978-3-031-73013-9_22.

Matena, M. S. and Raffel, C. A. Merging Models with Fisher-Weighted Averaging. Advances in Neural Information Processing Systems, 35:17703–17716, December 2022.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., and Ng, A. Y. Reading Digits in Natural Images with Unsupervised Feature Learning.

Ni, S., Chen, D., Li, C., Hu, X., Xu, R., and Yang, M. Forgetting before Learning: Utilizing Parametric Arithmetic for Knowledge Updating in Large Language Models. In ACL (1), January 2024.

Nilsback, M.-E. and Zisserman, A. Automated Flower Classification over a Large Number of Classes. In 2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing, pp. 722–729, December 2008. doi: 10.1109/ICVGIP.2008.47.

Oh, C., Li, Y., Song, K., Yun, S., and Han, D. DaWin: Training-free Dynamic Weight Interpolation for Robust Adaptation. In The Thirteenth International Conference on Learning Representations, October 2024.

Ortiz-Jimenez, G., Favero, A., and Frossard, P. Task Arithmetic in the Tangent Space: Improved Editing of Pre-Trained Models. Advances in Neural Information Processing Systems, 36:66727–66754, December 2023.

Pang, B. and Lee, L. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales, June 2005.

Parkhi, O. M., Vedaldi, A., Zisserman, A., and Jawahar, C. V. Cats and dogs. In 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 3498–3505, June 2012. doi: 10.1109/CVPR.2012.6248092.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. Journal of machine learning research, 21 (140):1–67, 2020.

Ruan, W., Yang, T., Zhou, Y., Liu, T., and Lu, J. From Task-Specific Models to Unified Systems: A Review of Model Merging Approaches, March 2025.

Sanh, V., Webson, A., Raffel, C., Bach, S. H., Sutawika, L., Alyafeai, Z., Chaffin, A., Stiegler, A., Scao, T. L., Raja, A., et al. Multitask prompted training enables zero-shot task generalization. arXiv preprint arXiv:2110.08207, 2021.

Shah, V., Ruiz, N., Cole, F., Lu, E., Lazebnik, S., Li, Y., and Jampani, V. ZipLoRA: Any Subject in Any Style by Effectively Merging LoRAs. In Leonardis, A., Ricci, E., Roth, S., Russakovsky, O., Sattler, T., and Varol, G. (eds.), Computer Vision – ECCV 2024, pp. 422–438, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73232-4. doi: 10.1007/978-3-031-73232-4_24.

Skorobogat, R., Roth, K., and Georgescu, M.-I. Subspace-boosted model merging. arXiv preprint arXiv:2506.16506, 2025.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A., and Potts, C. Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank. In Yarowsky, D., Baldwin, T., Korhonen, A., Livescu, K., and Bethard, S. (eds.), Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing, pp. 1631–1642, Seattle, Washington, USA, October 2013. Association for Computational Linguistics.

Stallkamp, J., Schlipsing, M., Salmen, J., and Igel, C. The German Traffic Sign Recognition Benchmark: A multi-class classification competition. In The 2011 International Joint Conference on Neural Networks, pp. 1453–1460, July 2011. doi: 10.1109/IJCNN.2011.6033395.

Sun, W., Li, Q., Geng, Y.-a., and Li, B. Cat merging: A training-free approach for resolving conflicts in model merging. arXiv preprint arXiv:2505.06977, 2025a.

Sun, W., Li, Q., Wang, W., Geng, Y.-a., and Li, B. Task arithmetic in trust region: A training-free model merging approach to navigate knowledge conflicts. arXiv preprint arXiv:2501.15065, 2025b.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., et al. Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288, 2023.

Veeling, B. S., Linmans, J., Winkens, J., Cohen, T., and Welling, M. Rotation Equivariant CNNs for Digital Pathology. In Frangi, A. F., Schnabel, J. A., Davatzikos, C., Alberola-López, C., and Fichtinger, G. (eds.), Medical Image Computing and Computer Assisted Intervention – MICCAI 2018, pp. 210–218, Cham, 2018. Springer International Publishing. ISBN 978-3-030-00934-2. doi: 10.1007/978-3-030-00934-2_24.

Wan, F., Zhong, L., Yang, Z., Chen, R., and Quan, X. FuseChat: Knowledge Fusion of Chat Models, August 2024.

Wang, K., Dimitriadis, N., Ortiz-Jimenez, G., Fleuret, F., and Frossard, P. Localizing Task Information for Improved Model Merging and Compression, May 2024.

Warstadt, A., Singh, A., and Bowman, S. R. Neural Network Acceptability Judgments. Transactions of the Association for Computational Linguistics, 7:625–641, September 2019. ISSN 2307-387X. doi: 10.1162/tacl_a_00290.

Wortsman, M., Ilharco, G., Gadre, S. Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A. S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., and Schmidt, L. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In Proceedings of the 39th International Conference on Machine Learning, pp. 23965–23998. PMLR, June 2022.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-MNIST: A Novel Image Dataset for Benchmarking Machine Learning Algorithms, September 2017.

Xiao, J., Ehinger, K. A., Hays, J., Torralba, A., and Oliva, A. SUN Database: Exploring a Large Collection of Scene Categories. International Journal of Computer Vision, 119(1):3–22, August 2016. ISSN 0920-5691, 1573-1405. doi: 10.1007/s11263-014-0748-y.

Yadav, P., Tam, D., Choshen, L., Raffel, C. A., and Bansal, M. TIES-Merging: Resolving Interference When Merging Models. Advances in Neural Information Processing Systems, 36:7093–7115, December 2023.

Yang, E., Shen, L., Guo, G., Wang, X., Cao, X., Zhang, J., and Tao, D. Model Merging in LLMs, MLLMs, and Beyond: Methods, Theories, Applications and Opportunities, September 2024a.

Yang, E., Shen, L., Wang, Z., Guo, G., Chen, X., Wang, X., and Tao, D. Representation Surgery for Multi-Task Model Merging, May 2024b.

Yang, E., Wang, Z., Shen, L., Liu, S., Guo, G., Wang, X., and Tao, D. AdaMerging: Adaptive Model Merging for Multi-Task Learning, May 2024c.

Yu, L., Yu, B., Yu, H., Huang, F., and Li, Y. Language Models are Super Mario: Absorbing Abilities from Homologous Models as a Free Lunch. In Forty-First International Conference on Machine Learning, June 2024.

Yuan, C., Zhang, T., and Niu, G. Neighbor-based feature and index enhancement for person re-identification. In Proceedings of the Computer Vision and Pattern Recognition Conference, pp. 5762–5769, 2025.

Zhang, X., Zhao, J., and LeCun, Y. Character-level Convolutional Networks for Text Classification. In Advances in Neural Information Processing Systems, volume 28. Curran Associates, Inc., 2015.

Zhang, Z., Chang, M.-C., and Li, X. Training-free image manipulation localization using diffusion models. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 39, pp. 10376–10384, 2025.

Zhu, D., Sun, Z., Li, Z., Shen, T., Yan, K., Ding, S., Wu, C., and Kuang, K. Model Tailor: Mitigating Catastrophic Forgetting in Multi-modal Large Language Models. In Forty-First International Conference on Machine Learning, June 2024.

A. Additional Implementation Setting

Datasets. Consistent with prior studies (Ilharco et al., 2022; Yang et al., 2024c), our primary experiments are conducted on 8 image classification benchmarks with various domain shift: SUN397 (Xiao et al., 2016), Cars (Krause et al., 2013), RESISC45 (Cheng et al., 2017), EuroSAT (Helber et al., 2019), SVHN (Netzer et al.), GTSRB (Stallkamp et al., 2011), MNIST (MNI), and DTD (Cimpoi et al., 2014). To further demonstrate the versatility of our method, we extend our evaluation to some additional datasets: Caltech101 (Fei-Fei et al., 2004), CIFAR10 (Krizhevsky), CIFAR100 (Krizhevsky), FGVC (Maji et al., 2013), Flowers102 (Nilsback & Zisserman, 2008), Food101 (Bossard et al., 2014), OxfordPets (Parkhi et al., 2012), STL10 (Coates et al., 2011), PCAM (Veeling et al., 2018), FER2013 (Goodfellow et al., 2013), EMNIST (Cohen et al., 2017), FashionMNIST (Xiao et al., 2017), RenderedSST2 (Socher et al., 2013) and KMNIST (Clanuwat et al., 2018). We fine-tune BERT on four binary classification datasets: AG News (Zhang et al., 2015), Rotten Tomatoes (Pang & Lee, 2005), CoLA (Warstadt et al., 2019), and SMS (Almeida et al., 2011). The resulting models are merged into a unified model using model merging techniques and evaluated on each task individually. TA and TIES use default settings, while TSV-M and SVC are applied exclusively to BERT's most critical linear layer ("output.dense.weight"). Finally, to evaluate the performance of the merged model on LLMs, the fine-tuned models WizardMath-7B-V1.0 (Luo et al., 2023) and Llama-2-7b-chat-hf (Touvron et al., 2023) are merged and tested on two benchmarks, AlpacaEval (Li et al., 2023) and GSM8K (Cobbe et al., 2021).

Datasets License. Datasets distributed under the MIT License include: SVHN (Netzer et al.), STL10 (Coates et al., 2011), EMNIST (Cohen et al., 2017), FashionMNIST (Xiao et al., 2017), and KMNIST (Clanuwat et al., 2018).

Datasets released under various Creative Commons licenses consist of: EuroSAT (Helber et al., 2019), DTD (Cimpoi et al., 2014), RESISC45 (Cheng et al., 2017), Food101 (Bossard et al., 2014)

The following datasets are available strictly for non-commercial research or academic use, typically under custom or restrictive academic licenses: SUN397 (Xiao et al., 2016), Cars (Krause et al., 2013), GTSRB (Stallkamp et al., 2011), FGVC (Maji et al., 2013), Flowers102 (Nilsback & Zisserman, 2008), OxfordPets (Parkhi et al., 2012), Caltech101 (Fei-Fei et al., 2004), FER2013 (Goodfellow et al., 2013), PCAM (Veeling et al., 2018), and RenderedSST2 (Socher et al., 2013).

The MNIST (MNI) and CIFAR10/100 (Krizhevsky) datasets are provided for unrestricted research use and are considered to be in the public domain or distributed without explicit license restrictions.

For full details regarding dataset licenses and terms of use, please refer to the official web pages or documentation of the respective datasets.

Implementation Detail. All experiments are conducted using PyTorch on a single NVIDIA GeForce A800 GPU. Although prior work (Ilharco et al., 2022; Yadav et al., 2023) relies on an additional hyperparameter λ to integrate the task update and pre-trained weight, we keep $\lambda = 1$ throughout our experiments. We note that jointly tuning λ on top of SVC (*i.e.*, calibrating singular values and then adjusting the overall update scale) could potentially yield further improvements; however, this introduces additional hyperparameter search and is beyond the scope of this work.

Experiments. Here, we present experiments that were not included in the main paper due to space limitations.

Weight averaging and spectral imbalance. Even under weight averaging (WA) $\Delta\mathbf{W}_{\text{merge}} = \frac{1}{K}\sum_{i=1}^{K}\Delta\mathbf{W}_i$. Fig. 3 indicates that, in the first few subspaces, the original singular value can be much larger than $K \cdot \sigma_i^\star$ (with K tasks), while the $1/K$ factor can make many other singular values overly small.

Merging experiments on 8 CV benchmark. The performance of each method on individual datasets is presented in detail. Complete results for ViT-B/32, ViT-B/16, and ViT-L/14 are provided in Table 5, Table 6, and Table 7, respectively. We additionally compare with CAT Merging (Sun et al., 2025a) and STAR (Lee et al., 2025). Due to limited time and computing resources, we did not re-run these baselines across all backbones and settings; instead, we report results only on the canonical ViT-B/32 8-task benchmark. As shown in Table 5, both CAT and STAR are substantially lower than our SVC-enhanced method.

Merging experiments on 14 CV benchmark. The performance of each method on individual datasets is presented in detail. Complete results for ViT-B/32, ViT-B/16, and ViT-L/14 are provided in Table 8, Table 9, and Table 10, respectively.

Table 5. Comparison of different model merging methods across eight vision benchmarks on ViT-B/32. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	SUN397	CARS	RESISC45	EUROSAT	SVHN	GTSRB	MNIST	DTD	AVG.
PRETRAINED	62.3	59.7	60.7	45.5	31.4	32.6	48.5	43.8	48.1
INDIVIDUAL	75.3	77.7	96.1	99.7	97.5	98.7	99.7	79.4	90.5
TRADITIONAL MTL	73.9	74.4	93.9	98.2	95.8	98.9	99.5	77.9	89.1
TA	55.1	54.9	66.7	77.2	80.2	69.7	97.3	50.1	68.9
w/ SVC (OURS)	68.1	66.3	83.5	95.6	91.2	86.0	99.2	65.6	81.9 (+13.0 ↑)
DARE	64.8	63.5	71.8	72.4	63.8	52.4	87.5	50.5	65.8
w/ SVC (OURS)	68.0	66.6	82.8	92.9	88.8	84.4	99.1	62.9	80.7 (+14.9 ↑)
TIES	59.6	58.6	71.0	81.3	86.1	70.9	98.4	54.7	72.6
w/ SVC (OURS)	69.0	66.0	83.0	90.8	88.7	78.6	98.7	65.0	80.0 (+7.4 ↑)
TSV-M	69.1	70.7	85.5	94.3	92.0	91.9	**99.3**	69.2	84.0
w/ SVC (OURS)	70.3	72.4	86.5	94.9	**92.1**	**92.4**	**99.3**	70.0	84.8 (+0.8 ↑)
ISO-C	74.8	74.1	87.9	92.9	83.1	86.0	98.2	67.9	83.1
w/ SVC (OURS)	74.1	72.6	88.1	**95.3**	88.2	89.6	99.0	70.0	84.6 (+1.5 ↑)
ISO-CTS	74.4	74.4	87.2	90.4	76.8	83.3	97.4	67.0	81.4
w/ **SVC**	**75.6**	**75.2**	**90.1**	94.9	86.1	91.6	98.9	**72.1**	**85.6** (+4.2 ↑)
STAR	55.9	55.1	67.4	77.7	80.2	68.1	97.2	50.1	69.0
CAT	68.1	65.4	80.5	89.5	85.5	78.5	98.6	60.7	78.4

Merging experiments on full fine-tuned BERT. The performance of each method on individual datasets is presented in detail. Complete results for BERT (Devlin et al., 2019) are provided in Table 11.

Merging experiments on full fine-tuned T5. The performance of each method on individual datasets is presented in detail. Complete results for T5 (Raffel et al., 2020) are provided in Table 12.

Merging experiments on PEFT fine-tuned T0. We report detailed performance on each dataset and summarize the full results for T0 (Sanh et al., 2021) in Table 13. We adopt IA^3 (Liu et al., 2022) for parameter-efficient fine-tuning. Across datasets, adding our SVC method consistently improves merged performance, highlighting the benefit of calibrating over-accumulated magnitudes during merging. Notably, methods that rely on SVD (*e.g.*, TSV-M, Iso-C, Iso-CTS) are not applicable in this setting because IA^3 updates are one-dimensional vectors rather than full weight matrices.

To make SVC applicable to 1D updates, we derive a vector-form calibration that keeps the merged direction unchanged and only corrects its scale. For each layer, let $(\{\boldsymbol{\tau}_i\}_{i=1}^K)$ be the task vectors from (K) experts and let $(\boldsymbol{\tau}_{\text{merge}})$ be an initial merged vector (we use simple averaging). We measure how much the merged vector can be explained by each expert via a projection coefficient

$$s_i \;=\; \frac{\langle \boldsymbol{\tau}_{\text{merge}}, \boldsymbol{\tau}_i \rangle}{\langle \boldsymbol{\tau}_i, \boldsymbol{\tau}_i \rangle}, \tag{24}$$

and aggregate them by $(\gamma = \frac{K}{\sum_{i=1}^K s_i})$. A smaller (γ) indicates that the merged update is over-counted along expert directions, which is the 1D counterpart of singular-value accumulation. We then rescale the merged vector as $(\tilde{\boldsymbol{\tau}}_{\text{merge}} = \gamma \boldsymbol{\tau}_{\text{merge}})$. This calibration is lightweight, requires no data, and extends SVC to PEFT scenarios where SVD-based baselines cannot operate.

Generality beyond matrix SVD. In practice, we apply SVC in a layer-wise manner: we perform SVD and calibration separately for the weight update matrix of each linear layer, and then reconstruct the calibrated update for that layer.

While our primary implementation is described for 2D weight matrices (where a standard SVD is directly applicable), the underlying principle of SVC, **measuring over-accumulation along shared directions and calibrating the magnitude without changing the direction**, extends naturally to other parameter shapes. For higher-order weight tensors, one can apply the same idea after tensor unfolding/reshaping into a matrix (or other appropriate spectral decomposition); for 1D parameter updates, SVD is undefined but the same "calibrate-the-scale" rule can be derived using vector projections. Our T0 (PEFT/IA^3) experiments provide a concrete example of this extension, where IA^3 produces one-dimensional updates and SVC still yields consistent gains.

Table 6. Comparison of different model merging methods across eight vision benchmarks on ViT-B/16. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	SUN397.	CARS.	RESISC45.	EUROSAT.	SVHN.	GTSRB.	MNIST.	DTD.	**AVG.**
PRETRAINED	63.8	64.7	66.4	54.6	52.0	43.4	51.7	44.7	55.2
INDIVIDUAL	81.8	86.8	96.9	99.8	97.9	99.2	99.8	82.1	93.0
TA	61.2	66.0	74.5	74.4	88.1	73.9	98.5	52.7	73.7
w/ SVC (OURS)	72.8	77.5	87.6	96.5	93.1	91.9	99.2	71.1	86.2 (+12.5 ↑)
DARE	67.6	70.0	76.0	78.6	75.3	59.8	94.4	50.1	71.5
w/ SVC (OURS)	72.6	77.4	86.6	95.2	91.7	88.9	99.1	67.3	84.8 (+13.3 ↑)
TIES	66.4	70.5	79.8	80.4	89.9	70.3	98.8	57.1	76.6
w/ SVC (OURS)	75.0	76.8	88.5	94.8	91.2	82.7	99.0	70.7	84.8 (+8.2 ↑)
TSV-M	72.8	80.3	89.1	96.6	**93.9**	94.0	**99.3**	72.7	87.3
w/ SVC (OURS)	73.9	81.3	89.8	97.3	93.8	94.8	**99.3**	73.7	88.0 (+0.7 ↑)
ISO-C	78.1	82.3	91.9	96.9	88.3	91.8	98.8	71.9	87.5
w/ SVC (OURS)	77.5	81.8	92.0	97.5	91.6	94.4	99.1	74.1	88.5 (+1.0 ↑)
ISO-CTS	77.9	83.2	92.0	96.4	84.9	91.3	98.4	71.1	86.9
w/ SVC (OURS)	**78.6**	**83.7**	**93.6**	**98.0**	90.5	**96.6**	99.1	**77.0**	**89.7** (+2.8 ↑)
STAR	63.2	66.3	73.7	79.0	85.6	76.4	98.4	51.8	74.3
CAT	72.9	75.9	83.1	92.8	88.2	82.7	98.8	62.7	82.1

Positioning vs. existing spectral-domain merging. SVC also operates in spectral space, but it serves a different purpose from prior SVD-based baselines and can be used as a drop-in post hoc calibration on top of an existing merge. Methods such as TSV-M focus on constructing a merged update that is less affected by conflicts, for example by selecting or reweighting directions to reduce destructive interference. The Iso-* family highlights that dominant singular components can suppress smaller ones, which is an important observation. However, in a single model the top singular values are naturally much larger than the tail of the spectrum. What is missing is a task-interaction explanation that attributes this suppression to repeated accumulation of shared directions across tasks, rather than to inherently larger singular values in the leading subspaces.

Building on our analysis, SVC targets this specific failure mode. When multiple tasks align in the same subspaces, naive aggregation can repeatedly add the same shared components, inflate the subspace strength, and concentrate the merged spectrum. SVC quantifies the degree of this over-counting in each subspace using projection-based coefficients, then corrects the magnitude while keeping the spectral directions unchanged. As a result, **SVC is complementary to strong spectral baselines.** When conflicts have already been largely mitigated, the remaining room for improvement can be small, which explains the typically modest but consistent gains on TSV-M. At the same time, methods that do not explicitly control subspace magnitudes tend to benefit more from this calibration.

Table 7. Comparison of different model merging methods across eight vision benchmarks on ViT-L/14. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	SUN397.	CARS.	RESISC45.	EUROSAT.	SVHN.	GTSRB.	MNIST.	DTD.	**AVG.**
PRETRAINED	66.9	77.9	71.3	62.2	58.5	50.6	76.4	55.4	64.9
INDIVIDUAL	84.9	92.4	97.4	99.7	98.1	99.2	99.7	84.2	94.4
TA	73.9	82.1	86.7	92.7	87.9	86.8	98.9	65.6	84.3
w/ SVC (OURS)	80.9	89.5	93.2	98.6	93.8	96.3	99.4	78.8	91.3 (+7.0 ↑)
DARE	71.1	81.6	82.6	90.6	78.3	70.8	97.0	63.1	79.4
w/ SVC (OURS)	79.3	88.1	92.6	97.7	92.5	94.8	99.3	76.4	90.1 (+10.7 ↑)
TIES	76.4	84.2	88.9	95.2	90.0	83.0	99.0	67.9	85.6
w/ SVC (OURS)	81.7	89.4	93.7	98.1	92.7	92.0	99.3	78.1	90.6 (+5.0 ↑)
TSV-M	79.0	89.8	94.0	98.8	95.3	96.2	**99.5**	79.1	91.5
w/ SVC (OURS)	79.4	90.3	94.2	**98.9**	**95.6**	96.8	**99.5**	79.9	91.8 (+0.3 ↑)
ISO-C	81.9	90.9	94.8	98.7	91.4	95.5	99.2	79.2	91.5
w/ SVC (OURS)	82.7	90.6	94.8	98.5	93.7	96.7	99.4	80.8	92.2 (+0.7 ↑)
ISO-CTS	81.3	91.2	94.7	98.6	89.3	95.3	99.2	77.9	90.9
w/ SVC (OURS)	**83.3**	**91.9**	**96.0**	98.8	93.8	**97.8**	99.4	**82.4**	**92.9** (+2.0 ↑)
STAR	74.5	82.0	86.7	93.1	87.8	87.3	98.8	65.0	84.4
CAT	78.7	88.5	91.1	96.3	91.3	95.7	99.4	75.7	89.6

Table 8. Comparison of model merging methods across 14 benchmarks on ViT-B/32. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	CAL101	CARS	CIF100	DTD	EURO	FGVC	FLO102	FOOD	GTSRB	MNIST	OXFP	RESISC	SUN	SVHN	**AVG.**
PRETRAINED	89.2	59.6	66.1	44.4	45.7	17.0	73.5	79.5	32.6	48.3	82.3	60.3	62.3	31.6	56.6
INDIVIDUAL	95.1	77.7	89.3	79.4	99.8	46.6	87.3	85.0	98.7	99.7	90.5	96.1	79.2	97.5	87.3
TA	58.2	36.7	46.3	32.7	61.6	17.4	29.4	35.6	49.6	91.4	62.8	42.0	27.1	58.8	46.4
w/ SVC (OURS)	87.8	53.2	64.7	53.5	73.7	30.3	47.1	53.8	66.7	95.1	73.6	67.0	57.4	60.0	63.1 (+16.7 ↑)
DARE	**92.7**	62.4	71.7	46.7	64.3	18.5	**73.5**	**78.9**	43.7	76.7	85.0	67.0	63.6	51.4	63.9
w/ SVC (OURS)	**92.7**	63.6	76.7	55.9	85.3	30.2	64.7	76.9	67.5	94.1	85.3	74.9	66.2	70.6	71.7 (+7.8 ↑)
TIES	86.6	55.9	69.7	47.0	70.6	30.0	49.0	65.1	53.0	87.7	69.8	63.5	59.0	55.1	61.6
w/ SVC (OURS)	89.2	55.6	70.8	49.5	68.7	33.9	49.0	65.7	52.4	87.4	70.1	64.7	62.2	52.8	62.3 (+0.7 ↑)
TSV-M	90.6	67.1	74.2	65.6	**94.6**	37.1	59.8	73.9	**88.4**	**99.0**	86.7	81.1	64.1	**86.9**	76.3
w/ SVC (OURS)	91.8	67.2	74.3	65.7	94.3	38.1	61.8	74.3	**88.4**	**99.0**	**86.7**	**81.3**	65.2	86.6	**76.8** (+0.5 ↑)
ISO-C	91.2	63.9	75.5	61.2	90.9	39.0	50.0	69.2	83.6	98.5	78.8	77.5	65.5	82.7	73.4
w/ SVC (OURS)	91.5	65.2	75.7	62.3	91.4	39.2	54.9	70.3	82.3	98.3	78.8	78.2	66.7	80.8	74.0 (+0.6 ↑)
ISO-CTS	**92.7**	**68.7**	77.1	64.9	89.8	39.2	64.7	75.5	85.8	98.5	83.4	**82.8**	68.7	82.4	76.7
w/ SVC (OURS)	91.3	68.0	**78.1**	**66.3**	91.0	**39.8**	63.7	75.1	85.0	98.5	82.9	**84.0**	**69.4**	80.5	76.7 (+0.0 ↑)

Table 9. Comparison of model merging methods across 14 benchmarks on ViT-B/16. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	CAL101	CARS	CIF100	DTD	EURO	FGVC	FLO102	FOOD	GTSRB	MNIST	OXFP	RESISC	SUN	SVHN	**AVG.**
PRETRAINED	86.7	64.7	69.6	44.7	54.6	25.1	67.7	85.7	43.4	51.7	87.2	66.4	63.8	52.0	61.7
INDIVIDUAL	95.7	86.8	83.0	82.2	99.8	46.1	82.4	88.9	99.2	99.8	94.6	96.9	82.0	97.9	88.2
TA	90.1	43.5	66.7	36.4	54.7	20.3	47.1	61.2	47.9	86.0	82.1	52.0	53.5	57.9	57.1
w/ SVC (OURS)	95.0	57.6	77.7	48.0	80.4	33.5	65.7	77.4	71.3	97.7	90.5	71.9	63.1	77.9	72.0 (+14.9 ↑)
DARE	88.2	66.4	76.2	46.4	65.5	27.0	70.6	**84.3**	51.0	78.7	86.7	69.8	65.6	62.3	67.0
w/ SVC (OURS)	93.5	56.2	78.6	45.1	76.5	32.0	64.7	78.4	65.8	96.0	87.5	68.6	62.4	74.4	70.0 (+3.0 ↑)
TIES	90.6	51.4	81.6	38.9	47.6	28.4	57.8	76.9	41.0	78.8	84.2	52.1	60.3	51.2	60.1
w/ SVC (OURS)	89.8	56.2	**82.8**	42.4	56.0	31.2	63.7	82.9	45.0	78.9	87.8	59.1	62.9	56.0	63.9 (+3.8 ↑)
TSV-M	91.9	67.0	79.9	53.0	90.0	37.7	71.6	81.9	82.8	98.4	93.2	77.4	64.9	82.7	76.6
w/ SVC (OURS)	92.1	67.1	79.9	53.3	90.4	**38.6**	73.5	82.3	83.5	98.4	**93.5**	78.0	65.1	83.0	77.0 (+0.4 ↑)
ISO-C	**95.5**	44.4	80.0	42.5	79.5	37.4	57.8	75.1	74.0	97.7	90.5	64.8	56.8	78.0	69.6
w/ SVC (OURS)	95.1	57.1	79.9	46.8	80.6	36.0	63.7	78.6	71.8	97.2	90.0	70.5	62.5	75.6	71.8 (+2.2 ↑)
ISO-CTS	95.0	66.5	76.4	54.0	**92.6**	38.3	74.5	80.3	**88.9**	**98.8**	92.7	78.2	63.4	**87.3**	77.6
w/ SVC (OURS)	94.7	**70.2**	77.8	**55.7**	92.4	38.4	**77.5**	82.8	85.3	98.5	92.9	**81.2**	**67.3**	84.9	**78.5** (+0.9 ↑)

Table 10. Comparison of model merging methods across 14 benchmarks on ViT-L/14. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	CAL101	CARS	CIF100	DTD	EURO	FGVC	FLO102	FOOD	GTSRB	MNIST	OXFP	RESISC	SUN	SVHN	AVG.
PRETRAINED	91.4	77.9	78.5	55.4	62.3	31.5	81.4	89.6	50.5	76.3	93.8	71.3	66.9	58.4	70.4
INDIVIDUAL	95.8	92.3	87.8	84.1	99.7	65.0	88.2	92.6	99.2	99.7	94.3	97.4	84.9	98.1	91.4
TA	91.9	36.0	78.5	41.4	52.6	25.2	60.8	56.0	46.6	84.2	85.9	48.3	55.8	45.1	57.7
W/ SVC (OURS)	93.7	66.8	86.4	56.9	81.2	48.5	74.5	79.1	76.6	97.8	92.1	74.4	66.4	78.9	76.7 (+19.0 ↑)
DARE	92.5	76.9	85.2	58.0	78.5	33.9	81.4	**88.5**	62.2	91.5	93.8	76.6	69.0	68.1	75.4
W/ SVC (OURS)	93.2	70.0	86.8	56.6	83.9	47.9	76.5	82.3	77.2	97.1	93.8	77.8	68.1	79.0	77.9 (+2.5 ↑)
TIES	92.5	51.1	88.7	47.9	48.9	38.1	66.7	76.1	45.9	65.7	91.3	58.5	62.3	39.5	62.4
W/ SVC (OURS)	92.9	51.9	88.8	46.0	49.0	48.6	61.8	77.6	45.9	66.2	93.8	60.5	61.3	46.9	63.6 (+1.2 ↑)
TSV-M	94.0	77.0	**88.9**	62.8	92.8	51.6	80.4	85.2	89.2	98.8	**94.8**	83.3	69.6	84.0	82.3
W/ SVC (OURS)	94.1	78.2	88.6	63.6	94.0	52.7	82.3	86.2	90.1	98.9	94.6	84.5	70.2	85.2	83.1 (+0.8 ↑)
ISO-C	94.0	60.7	87.4	54.7	84.0	53.3	77.5	75.4	80.1	98.3	93.5	71.2	63.2	81.4	76.8
W/ SVC (OURS)	93.8	71.7	86.8	58.1	84.7	50.0	75.5	81.9	79.4	97.9	93.5	78.3	68.8	78.5	78.5 (+1.7 ↑)
ISO-CTS	**94.2**	82.2	87.4	**70.8**	**96.7**	**57.5**	**90.2**	85.1	**94.8**	**99.2**	**94.8**	84.8	71.3	**90.7**	85.7
W/ SVC (OURS)	94.0	**83.9**	88.1	70.1	96.6	56.0	89.2	87.8	93.3	**99.2**	94.6	**87.6**	**73.7**	89.0	**85.9** (+0.2 ↑)

Table 11. Comparison of model merging methods across three NLP benchmarks on BERT. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	AG_NEWS	ROTTEN_TOMATOES	COLA	AVG. ACC.
EXPERT	99.1	84.1	78.3	87.2
WA	48.5	58.9	72.1	59.8
TA	50.4	51.1	69.1	56.9
W/ SVC (OURS)	**67.9**	**78.3**	60.8	**69.0** (+12.1 ↑)
TIES	51.6	57.4	70.1	59.7
W/ SVC (OURS)	52.7	61.1	70.2	61.3 (+1.6 ↑)
DARE	50.0	50.4	72.4	57.6
W/ SVC (OURS)	50.0	50.7	**73.1**	57.9 (+0.3 ↑)
TSV-M	59.4	58.2	64.1	60.6
W/ SVC (OURS)	59.5	59.0	65.5	61.3 (+0.8 ↑)
ISO-C	49.7	50.0	69.1	56.3
W/ SVC (OURS)	50.4	50.2	69.1	56.6 (+0.3 ↑)
ISO-CTS	49.8	50.0	69.1	56.3
W/ SVC (OURS)	50.2	50.1	69.1	56.5 (+0.2 ↑)

Table 12. Comparison of model merging methods across 11 NLP benchmarks using bigscience/T5. Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	RTE	CB	WINOGR.	WIC	WSC	COPA	H-SWAG	STORY	ANLI-R1	ANLI-R2	ANLI-R3	AVG.
TA	40.6	53.1	34.4	60.9	37.5	48.4	21.9	50.0	34.4	40.6	34.4	41.5
W/ SVC (OURS)	43.8	62.5	46.9	68.8	37.5	59.4	21.9	59.4	34.4	**43.8**	31.2	46.3 (+4.8 ↑)
DARE	40.6	53.1	31.2	59.4	37.5	50.0	21.9	50.0	34.4	40.6	34.4	41.2
W/ SVC (OURS)	43.8	71.9	34.4	65.6	37.5	60.9	21.9	68.8	34.4	37.5	31.2	46.2 (+5.0 ↑)
TIES	40.6	56.2	34.4	**71.9**	37.5	**68.8**	21.9	59.4	34.4	**43.8**	31.2	45.5
W/ SVC (OURS)	43.8	75.0	**50.0**	**71.9**	37.5	59.4	25.0	**78.1**	34.4	40.6	31.2	**49.7** (+4.2 ↑)
TSV-M	**43.8**	71.9	40.6	57.8	37.5	59.4	25.0	68.8	34.4	40.6	31.2	46.5
W/ SVC (OURS)	**43.8**	71.9	40.6	57.8	37.5	57.8	25.0	71.9	34.4	40.6	31.2	46.6 (+0.1 ↑)
ISO-C	40.6	68.8	37.5	57.8	37.5	50.0	**28.1**	59.4	34.4	40.6	28.1	43.9
W/ SVC (OURS)	43.8	**78.1**	43.8	68.8	37.5	62.5	**28.1**	68.8	34.4	40.6	31.2	48.9 (+5.0 ↑)
ISO-CTS	40.6	53.1	37.5	42.2	**39.1**	48.4	25.0	43.8	31.2	37.5	28.1	38.8
W/ SVC (OURS)	43.8	75.0	43.8	53.1	37.5	56.2	**28.1**	65.6	**34.4**	40.6	31.2	46.3 (+7.5 ↑)

Table 13. Comparison of model merging methods across 11 NLP benchmarks using bigscience/T0_3B (IA3). Bold values indicate the best performance among merging-based techniques. The notation "(Ours)" highlights the integration of our proposed SVC method.

METHOD	RTE	CB	WINOGR.	WIC	WSC	COPA	H-SWAG	STORY	ANLI-R1	ANLI-R2	ANLI-R3	**AVG.**
TA	71.9	56.2	53.1	29.6	**65.6**	78.1	46.9	87.5	46.9	28.1	25.0	53.5
w/ SVC	71.9	**81.2**	**59.4**	**67.2**	43.8	**93.8**	**50.0**	**93.8**	**56.2**	**46.9**	**59.4**	**65.8** (+12.3 ↑)
TIES	71.9	59.4	53.1	31.2	62.5	79.6	46.9	87.5	46.9	31.2	25.0	54.1
w/ SVC	71.9	59.4	53.1	29.6	**65.6**	78.1	46.9	78.1	46.9	31.2	25.0	54.1 (+0.0 ↑)
DARE	71.9	59.4	53.1	29.6	62.5	78.1	43.8	87.5	46.9	28.1	25.0	53.3
w/ SVC	**75.0**	56.2	53.1	31.2	62.5	79.6	46.9	87.5	50.0	31.2	28.1	54.7 (+1.4 ↑)