UNDERSTANDING SPARSIFICATION IN MULTI-TASK MODEL MERGING VIA LINEARIZED FUNCTIONAL DECOMPOSITION

Anonymous authors
Paper under double-blind review

ABSTRACT

Multi-task model merging offers an efficient alternative to multi-task training, yet post-hoc parameter sparsification remains weakly explained by training-time analyses. We introduce linearized functional decomposition (LFD), which separates target-task and non-target contributions and bounds the merged–LFD discrepancy. The resulting path diagnostics also track LFD–target consistency and retained single-task quality. We examine how transformations change task-vector norms and cross-task interactions, then compare the bound with measured path behavior. Across vision and language tasks, effective sparsification mainly narrows the merged–LFD gap, while the other diagnostics reveal changes in target consistency and capability. The diagnostics also predict relative gains for held-out tasks and show where transformations beyond sparsification help or hurt.

1 INTRODUCTION

Model merging (Yang et al., 2024a) combines multiple fine-tuned checkpoints into a single model without retraining a full multi-task system. In the pre-training and fine-tuning workflow, this is especially attractive: all downstream checkpoints share the same initialization, so their parameter differences can be compared and combined directly. Task arithmetic (Ilharco et al., 2023) turns this observation into a simple rule by adding task vectors, i.e., the parameter changes induced by fine-tuning. Compared with conventional multi-task learning (Sener and Koltun, 2018), this post-processing route avoids data mixing, loss balancing, and gradient surgery (Kendall et al., 2018; Liu et al., 2021; 2019; Yu et al., 2020). Merged performance nevertheless usually trails the corresponding single-task fine-tuned models, raising the question of how task-specific information and cross-task interactions jointly shape this gap when several tasks share one parameter state at inference time.

Much of the empirical progress has come from modifying task vectors after fine-tuning. Representative methods reset small parameter values, randomly drop parameters, or remove parameters that conflict across tasks (Davari and Belilovsky, 2024; Du et al., 2024; Wang et al., 2024; Yadav et al., 2023; Yu et al., 2024). These sparsification rules are useful in practice because they are cheap, often data-free, and can be layered on top of newer merging procedures (Yang et al., 2025; Zhou et al., 2025). Their behavior is also notable: dropping a very large fraction of parameters can preserve, and sometimes improve, merged accuracy. This behavior points to redundancy and structured interference in task vectors. A mask changes the information retained in each vector as well as the interactions among vectors after they are summed. We therefore examine these effects separately.

Existing theory relates merged and fine-tuned models through linearization and weight disentanglement (Figure 1) (Jin et al., 2025; Ortiz-Jimenez et al., 2024; Yoshida et al., 2025). Linearization characterizes a network's response to parameter changes, while disentanglement concerns how task contributions combine. Together, these properties give conditions under which parameter combinations produce predictable functions. They can be encouraged during training, for example through NTK-style fine-tuning or cross-task orthogonality regularization (Ortiz-Jimenez et al., 2024; Yoshida et al., 2025). Ordinary fine-tuning can realize these properties to different degrees across tasks. A mask then changes the vectors that enter the merge, potentially shifting both the model's response and the interactions among tasks. Its effect therefore calls for a description of the resulting merged model and the target-task model in their existing parameter state.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

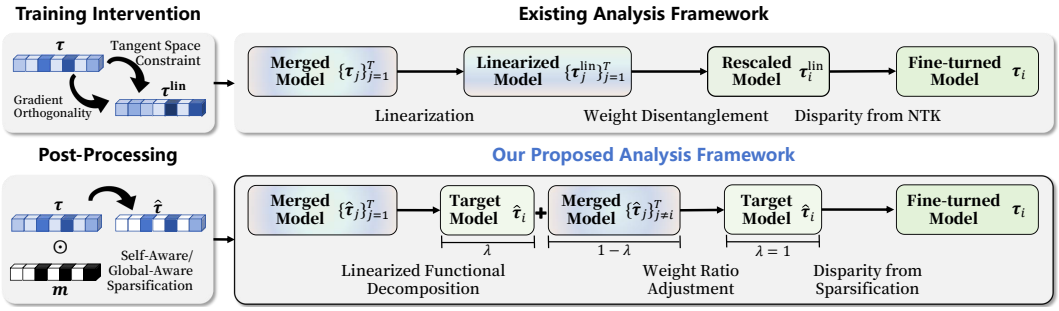


Figure 1: Comparison of training-time analysis and post-hoc sparsification. LFD measures how empirical masks alter the path from a merged model to each target-task model.

Our question is: *which part of the merged-to-target gap do empirical masks change?* An answer should apply to vectors from ordinary fine-tuning, distinguish redundancy within a vector from interference across vectors, and describe the geometry actually produced by each mask. Methods built from a vector's own magnitudes and methods built from cross-task comparisons may therefore improve merging for different reasons. Small pairwise cosines alone do not determine the sign of the combined interaction. The comparison should also track how much of the target task's own capability survives the transformation, alongside changes in the merged-to-target path.

We address this question with *linearized functional decomposition* (LFD). LFD inserts an intermediate state between the merged model and the target-task model. For each target task, it separates the contribution of the target task vector from the aggregate contribution of all non-target task vectors. Under linearized fine-tuning this decomposition is exact; under standard fine-tuning it defines two gaps to examine: merged–LFD and LFD–target. We call this sequence of states the LFD path. We derive a Taylor-style bound for the representation gap between the merged model and the LFD form, then use this bound to locate where different sparsification mechanisms act along the path.

Contributions.

- **A target-conditioned path.** LFD separates the merged–LFD and LFD–target gaps and records the single-task capability retained by each transformation. These quantities can be measured on ordinarily fine-tuned models after the checkpoints are available.
- **A geometric explanation.** A representation-gap bound separates individual task-vector squared norms from signed cross-task interactions. Conditional comparisons explain self-aware and global-aware masks, with finite-interaction checks on ViT and BERT.
- **Predictive and cross-method evidence.** The diagnostics predict relative mask gains on held-out ViT tasks. A controlled lowrank/TSV comparison applies the same analysis to orthogonalization and exposes its dependence on the merge weight.

2 Background

2.1 Multi-Task Model Merging

Let $f : \mathcal{X} \times \Theta \to \mathcal{Y}$ be a neural network initialized at pre-trained weights $\boldsymbol{\theta}_0$. For task i, inputs follow μ_i, and fine-tuning on task data produces $\boldsymbol{\theta}_i^*$. The corresponding task vector is $\boldsymbol{\tau}_i = \boldsymbol{\theta}_i^* - \boldsymbol{\theta}_0$. Task arithmetic (Ilharco et al., 2023) merges T tasks by

$$\boldsymbol{\theta}_{\text{merged}} = \boldsymbol{\theta}_0 + \lambda \sum_{i=1}^{T} \boldsymbol{\tau}_i, \tag{1}$$

where λ is searched on validation data. This simple form avoids the engineering burden of joint multi-task training, but it also exposes task interference when different $\boldsymbol{\tau}_i$ compete over the same parameters. We therefore study the geometry of $\{\boldsymbol{\tau}_i\}_{i=1}^{T}$.

2.2 Empirical Methods on Multi-Task Model Merging

We study Task Arithmetic (TA) (Ilharco et al., 2023), TIES (Yadav et al., 2023), Drop and Rescale (DARE) (Yu et al., 2024), TALL (Wang et al., 2024), and Breadcrumbs (BRE) (Davari and Belilovsky, 2024). Their differences can be expressed by a sparsified task vector

$$f_{\text{merged}}(x) = f(x; \boldsymbol{\theta}_0 + \lambda \sum_{i=1}^{T} \underbrace{\boldsymbol{\alpha} \odot \boldsymbol{m}_i^{(\text{self})} \odot \boldsymbol{m}_i^{(\text{glob})} \odot \boldsymbol{\tau}_i}_{\hat{\boldsymbol{\tau}}_i}), \tag{2}$$

We label minimum-magnitude deletion MIN, random deletion RAND, and sign election ELT. The task-specific conflict-preserving mask is CPS; TALL applies the shared consensus of these masks. MIN followed by ELT is MEL. These labels also determine the lowercase roman superscripts: (ta), (min), (rand), (dare), (bre), (elt), (cps), (tall), and (mel).

Here $\boldsymbol{m}_i^{(\text{self})}$ is computed from task i alone, $\boldsymbol{m}_i^{(\text{glob})}$ is computed from multiple task vectors, and $\boldsymbol{\alpha}$ rescales retained parameter values. Self-aware masks mainly remove intra-task redundancy: for example, minimum-value masking assumes low-magnitude parameter values are less task-critical, while BRE also removes extreme parameter values that may be overly task-specific. Global-aware masks use cross-task information: sign-election masks keep parameter values whose signs agree with the majority direction, while conflict-preserving masks compare one task vector with the aggregate of the others. DARE combines random self-aware masking with a rescaling factor, making it useful for separating sparsification from scale correction in our comparisons.

For Table 1, let d be the task-vector dimension and $\boldsymbol{\rho}_i = (\rho_{i,1}, \ldots, \rho_{i,d})$ give the coordinatewise ascending ranks of $|\boldsymbol{\tau}_{i,k}|$ from 1 to d, with ties resolved consistently.

Table 1: Detailed mapping of representative model-merging methods into Equation 2.

Method	**Self-Aware Mask $\boldsymbol{m}_i^{(\text{self})}$**	**Global-Aware Mask $\boldsymbol{m}_i^{(\text{glob})}$**	**Rescaling $\boldsymbol{\alpha}$**
Task Arithmetic	1	1	1
TIES	$\boldsymbol{m}_i^{(\text{min})} = \mathbb{1}[\boldsymbol{\rho}_i > \lfloor pd \rfloor]$	$\boldsymbol{m}_i^{(\text{elt})} = \mathbb{1}(\text{sgn}(\boldsymbol{\tau}_i) = \text{sgn}(\sum_j^T \boldsymbol{\tau}_j))$	$1/\sum_i^T \boldsymbol{m}_i$
DARE	$\boldsymbol{m}^{(\text{rand})} = \mathbb{1}(\delta = 0), \delta \sim \text{Bernoulli}(p)^d$	1	$1/(1-p)$
TALL	1	$\boldsymbol{m}_i^{(\text{cps})} = \mathbb{1}(\lvert\boldsymbol{\tau}_i\rvert \geq r \lvert \sum_{j\neq i}^T \boldsymbol{\tau}_j \rvert)$ $\boldsymbol{m}^{(\text{tall})} = \mathbb{1}(\sum_i^T \boldsymbol{m}_i^{(\text{cps})} \geq c)$	1
BRE	$\boldsymbol{m}_i^{(\text{bre})} = \mathbb{1}[\lfloor p_1 d \rfloor < \boldsymbol{\rho}_i \leq \lfloor p_2 d \rfloor]$	1	1

2.3 Theoretical Analysis on Model Merging

Prior theory connects merging to two intermediate states, shown on the upper path of Figure 1. The first is linearization. Around $\boldsymbol{\theta}_0$, a linearized model satisfies

$$f^{\text{lin}}(x; \boldsymbol{\theta}_0 + \delta) = f(x; \boldsymbol{\theta}_0) + \nabla_{\boldsymbol{\theta}} f(x; \boldsymbol{\theta}_0)^\top \delta, \tag{3}$$

so adding task vectors in parameter space induces an additive relation in function space. This makes the merged model easier to analyze and is encouraged by training-time procedures such as NTK fine-tuning (Ortiz-Jimenez et al., 2024), which promotes this additive behavior.

The second state is weight disentanglement (Jin et al., 2025; Ortiz-Jimenez et al., 2024). Informally, when x belongs to task i, the output of a model containing many task vectors should depend on the i-th task vector alone. A compact representation is

$$f\big(x; \boldsymbol{\theta}_0 + \sum_{i=1}^{T} \alpha_i \boldsymbol{\tau}_i\big) = \sum_{i=1}^{T} g_i(x, \alpha_i \boldsymbol{\tau}_i) + g_0(x), \tag{4}$$

Here $D_i = \text{supp}(\mu_i)$ is task i's input support. The function $g_i : \mathcal{X} \times \Theta \to \mathcal{Y}$ is a task-local component: $g_i(x, \alpha_i \boldsymbol{\tau}_i) = 0$ for $x \notin D_i$, and its parameter dependence involves only $\alpha_i \boldsymbol{\tau}_i$. The function $g_0 : \mathcal{X} \to \mathcal{Y}$ is independent of the task vectors and vanishes on $\bigcup_i D_i$. Under the disjoint-support formulation, one can take $g_i(x, \alpha_i \boldsymbol{\tau}_i) = f(x; \boldsymbol{\theta}_0 + \alpha_i \boldsymbol{\tau}_i)\mathbb{1}[x \in D_i]$ and $g_0(x) = f(x; \boldsymbol{\theta}_0)\mathbb{1}[x \notin \bigcup_i D_i]$ (Ortiz-Jimenez et al., 2024). Orthogonality constraints and related fine-tuning interventions can encourage this property (Yoshida et al., 2025).

Figure 1 motivates a complementary question. Given already fine-tuned task vectors, we can compare the merged model with a target-conditioned intermediate state and then with the target-task model. This comparison locates the functional changes caused by a post-hoc transformation, whether it removes coordinates or changes their geometry after fine-tuning.

3 Analysis Framework

To identify where a post-hoc transformation acts, we insert LFD between the merged and target-task models. Its target-conditioned construction keeps the target contribution explicit and groups the remaining task vectors into a second parameter state.

3.1 Linearized Functional Decomposition

For target task i and $x \sim \mu_i$, linearized merging is functionally additive:

$$f^{\text{lin}}(x; \boldsymbol{\theta}_0 + \lambda \sum_{j=1}^{T} \boldsymbol{\tau}_j^{\text{lin}}) = \underbrace{\lambda f^{\text{lin}}_{\text{target}}(x) + (1-\lambda) f^{\text{lin}}_{\text{others}}(x)}_{f^{\text{lin}}_{\text{lfd}}(x)}, \tag{5}$$

where $f^{\text{lin}}_{\text{target}}(x) = f^{\text{lin}}(x; \boldsymbol{\theta}_0 + \boldsymbol{\tau}_i^{\text{lin}})$ and $f^{\text{lin}}_{\text{others}}(x) = f^{\text{lin}}(x; \boldsymbol{\theta}_0 + \frac{\lambda}{1-\lambda} \sum_{j \neq i} \boldsymbol{\tau}_j^{\text{lin}})$. We keep the same interpolation form after standard fine-tuning:

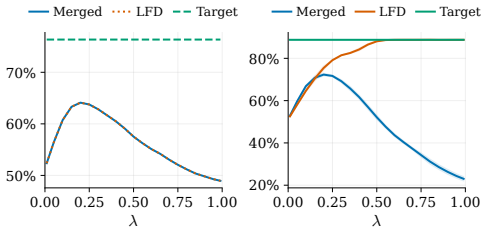
$$f_{\text{lfd}}(x) = \lambda f_{\text{target}}(x) + (1-\lambda) f_{\text{others}}(x), \tag{6}$$

with $f_{\text{target}}(x) = f(x; \boldsymbol{\theta}_0 + \hat{\boldsymbol{\tau}}_i)$ and $f_{\text{others}}(x) = f(x; \boldsymbol{\theta}_0 + \frac{\lambda}{1-\lambda} \sum_{j \neq i} \hat{\boldsymbol{\tau}}_j)$. For each target task, we represent the remaining task vectors by their aggregate contribution.

The sequence $f_{\text{merged}} \to f_{\text{lfd}} \to f_{\text{target}}$ defines the LFD path. The first gap records departures from functional additivity, while the second records the effect of mixing the target output with the aggregate of other tasks. The merged and LFD forms share their constant and first-order terms around $\boldsymbol{\theta}_0$; Appendix C.1 bounds the remaining representation discrepancy caused by curvature.

3.2 Performance Gaps Along the LFD Path

Weight Ratio Adjustment. Equation 5 makes the merged and LFD outputs coincide after linearized fine-tuning. With standard fine-tuning, curvature can separate them; in Figure 2, LFD has higher accuracy over the displayed nonlinear range. Changing λ also changes the mixture in Equation 6. The identity $f_{\text{lfd}} - f_{\text{target}} = (1-\lambda)(f_{\text{others}} - f_{\text{target}})$ explains its target-directed trend whenever the non-target output remains bounded near $\lambda = 1$.



(a) NTK fine-tuning (b) Standard fine-tuning

Figure 2: LFD under different strategies: one NTK run and the three-run standard mean.

Disparity from Sparsification. A mask changes the target state from $f_{\text{ft}}(x) = f(x; \boldsymbol{\theta}_0 + \boldsymbol{\tau}_i)$ to $f_{\text{target}}(x) = f(x; \boldsymbol{\theta}_0 + \hat{\boldsymbol{\tau}}_i)$. NormAccuracy below measures the resulting single-task capability. At a fixed λ, the same mask can change both LFD path gaps. The experiments in Section 4 show especially large changes in the merged–LFD gap, motivating a bound that exposes the task-vector contribution to this stage.

Theorem 3.1 (Merged–LFD gap bound)**.** *Fix x and $0 < \lambda < 1$. Suppose $f(x; \cdot)$ is twice continuously differentiable on a convex region containing the target and other-task parameters, with Hessian operator norm bounded by M throughout that region. Then*

$$\|f_{\text{merged}}(x) - f_{\text{lfd}}(x)\| \leq \frac{M}{2} \underbrace{\left\| \sqrt{\lambda(1-\lambda)}\hat{\boldsymbol{\tau}}_i - \lambda \sqrt{\frac{\lambda}{1-\lambda}} \sum_{j \neq i} \hat{\boldsymbol{\tau}}_j \right\|^2}_{\widehat{\mathcal{B}}_i}. \tag{7}$$

The quantity $\widehat{\mathcal{B}}_i$ is the task-vector term in the bound. It measures how the vectors and λ affect the bound for a fixed curvature bound M. The proof uses the interpolation error along the segment joining the two parameter states in parameter space (Appendix C.1).

For processed vectors $\{\hat{\boldsymbol{\tau}}_j\}$, the bound term separates into squared norms and cross-task interactions:

$$\widehat{\mathcal{B}}_i = \underbrace{\lambda(1-\lambda)\|\hat{\boldsymbol{\tau}}_i\|^2 + \frac{\lambda^3}{1-\lambda}\sum_{j\neq i}\|\hat{\boldsymbol{\tau}}_j\|^2}_{\widehat{\mathrm{E}}_i} + \underbrace{\frac{2\lambda^3}{1-\lambda}\sum_{\substack{j<\ell \\ j,\ell\neq i}}\langle\hat{\boldsymbol{\tau}}_j,\hat{\boldsymbol{\tau}}_\ell\rangle - 2\lambda^2\sum_{j\neq i}\langle\hat{\boldsymbol{\tau}}_i,\hat{\boldsymbol{\tau}}_j\rangle}_{\widehat{\mathcal{B}}_{i,\mathrm{cross}}}. \tag{8}$$

Here $\widehat{\mathrm{E}}_i$ is the weighted sum of individual squared norms, and $\widehat{\mathcal{B}}_{i,\mathrm{cross}}$ is the signed cross-task contribution. Removing the hats gives the corresponding quantities for the original vectors $\boldsymbol{\tau}_j$. For a specified method, we use its label from Section 2.2 as the lowercase roman superscript: RAND gives $\hat{\boldsymbol{\tau}}_j^{(\mathrm{rand})}$, MIN gives $\hat{\boldsymbol{\tau}}_j^{(\mathrm{min})}$, and BRE gives $\hat{\boldsymbol{\tau}}_j^{(\mathrm{bre})}$. Scalar quantities use the same superscripts, such as $\mathcal{B}_i^{(\mathrm{min})}$ and $\mathrm{E}_i^{(\mathrm{min})}$. We suppress the dependence on λ. Equation 8 separates the contributions of individual task-vector magnitudes and cross-task interactions.

For task i, write $\pi_i(f,x)$ for the predicted class obtained from output $f(x)$ using the task's evaluation rule. Define $\mathrm{dist}_i(f(x),g(x)) = \mathbb{1}[\pi_i(f,x)\neq\pi_i(g,x)]$. These indicators give per-task and mean disagreements for both stages of the LFD path on each target task:

$$\begin{aligned}
\xi_{\mathrm{mg-lfd},i} &= \mathbb{E}_{x\sim\mu_i}[\mathrm{dist}_i(f_{\mathrm{merged}}(x), f_{\mathrm{lfd}}(x))], \quad \xi_{\mathrm{mg-lfd}} = \frac{1}{T}\sum_i \xi_{\mathrm{mg-lfd},i}, \\
\xi_{\mathrm{lfd-tgt},i} &= \mathbb{E}_{x\sim\mu_i}[\mathrm{dist}_i(f_{\mathrm{lfd}}(x), f_{\mathrm{target}}(x))], \quad \xi_{\mathrm{lfd-tgt}} = \frac{1}{T}\sum_i \xi_{\mathrm{lfd-tgt},i}.
\end{aligned} \tag{9}$$

We also measure the single-task capability retained by masking:

$$\mathrm{NormAccuracy} = \frac{1}{T}\sum_{i=1}^{T}\frac{A_i(\hat{\boldsymbol{\tau}}_i)}{A_i(\boldsymbol{\tau}_i)}, \qquad A_i(\delta) = \mathrm{acc}_i[f(\cdot;\boldsymbol{\theta}_0+\delta)]. \tag{10}$$

Here $\mathrm{acc}_i[f]$ is accuracy on task i under the same prediction rule. The following result bounds the merged-to-target accuracy gap through both path disagreements:

Proposition 3.2 (Diagnostics and the accuracy gap)**.** *For every classification task i,*

$$|\,\mathrm{acc}_i[f_{\mathrm{merged}}] - \mathrm{acc}_i[f_{\mathrm{target}}]| \leq \xi_{\mathrm{mg-lfd},i} + \xi_{\mathrm{lfd-tgt},i}.$$

Two models' correctness indicators can differ only when their predictions differ. Applying this fact to each stage and using the triangle inequality proves the result.

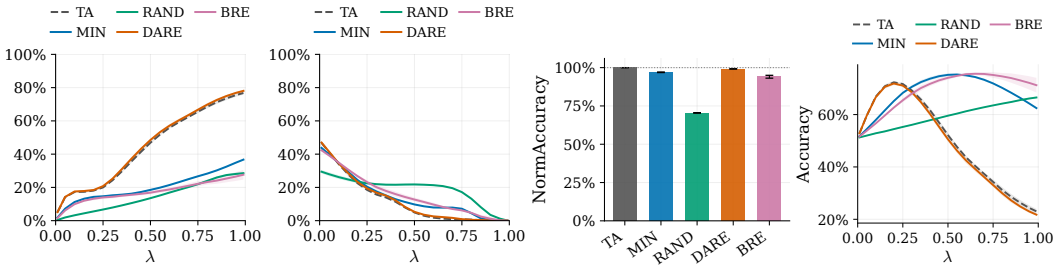
This proposition concerns classification accuracy. NormAccuracy complements the two disagreements by recording whether a mask preserves the target-task vector's own performance.

4 Analysis of Parameter Sparsification

4.1 Experimental Setup

We analyze the sparsification methods in Table 1 in both CV and NLP settings. The main ViT and BERT results use averages over three random seeds. We first average equally over the eight tasks within each run, then average the three runs; shaded bands and error bars show one standard deviation across runs. Appendix D.2 gives the seed settings and the scope of the supplementary experiments.

For CV, we follow the prior merging protocols of image classification (Ilharco et al., 2023; Zhou et al., 2024b), and evaluate eight image-classification tasks with a ViT-B-32 (Dosovitskiy, 2020) image encoder initialized by CLIP (Radford et al., 2021). We use the KNN evaluation protocol, avoiding fixed-head alignment artifacts noted by Kong et al. (2024). The default hyperparameters are $p = p_1 = 0.9$, $p_2 = 0.99$, $r = 0.5$, and $c = 2$. Appendix D.1 gives the dataset and search details.



(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{lfd-tgt}}$ (ViT) (c) NormAccuracy (ViT) (d) Merged accuracy (ViT)

Figure 3: Self-aware masks on ViT: three-run means with one standard deviation across runs. The merged–LFD gap decreases, with different amounts of single-task degradation.

For NLP, we follow the merging protocols of language understanding (Yu et al., 2024) (Appendix D.2) and evaluate on the GLUE benchmark (Wang et al., 2018) with a BERT-base (Devlin et al., 2019) backbone. The default hyperparameters are $p = p_1 = 0.8$, $p_2 = 0.99$, $r = 0.3$, and $c = 2$.

The main figures present ViT diagnostics. BERT shows corresponding mask-dependent changes along the LFD path in Appendix Figures 10 and 11.

4.2 Self-Aware Sparsification

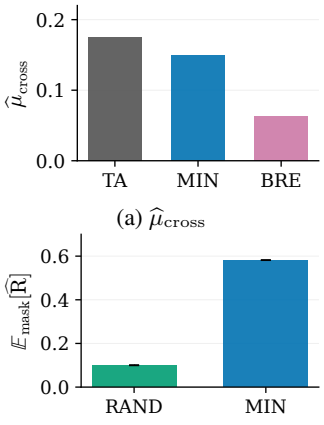
Empirical observation. Figure 3 shows that self-aware masks primarily change $\xi_{\mathrm{mg-lfd}}$. MIN reduces this gap while preserving most single-task accuracy. RAND and BRE can reduce it further, with greater losses in NormAccuracy. DARE's rescaling changes this pattern. We explain these differences through the task-vector term in the bound.

An orthogonal model. We first use pairwise orthogonality as a reference case: it sets the signed cross-task contribution in Equation 8 to zero, so differences in the bound can be attributed to squared norms. Prior CLIP and language-model studies report small cross-task alignments (Ilharco et al., 2023; Zhou et al., 2024a). We measure the largest absolute pairwise cosine by

$$\widehat{\mu}_{\mathrm{cross}} = \max_{j \neq \ell} \frac{|\langle \hat{\boldsymbol{\tau}}_j, \hat{\boldsymbol{\tau}}_\ell \rangle|}{\|\hat{\boldsymbol{\tau}}_j\| \|\hat{\boldsymbol{\tau}}_\ell\|}.$$

Figure 4a shows the original, MIN, and BRE sets on ViT; Appendix B includes BERT and the other masks. The cosine measurements motivate this reference case, while the finite comparisons below retain the measured signed interactions. Under pairwise orthogonality within each deterministic vector set, $\widehat{\mathcal{B}}_i = \widehat{\mathrm{E}}_i$.
For nonzero original $\boldsymbol{\tau}_j$, the squared-norm ratios are

$$\widehat{\mathrm{R}}_j = \frac{\|\hat{\boldsymbol{\tau}}_j\|^2}{\|\boldsymbol{\tau}_j\|^2}, \qquad \widehat{\mathrm{R}} = \frac{1}{T} \sum_j \widehat{\mathrm{R}}_j. \tag{11}$$



(a) $\widehat{\mu}_{\mathrm{cross}}$

(b) $\mathbb{E}_{\mathrm{mask}}[\widehat{\mathrm{R}}]$

Figure 4: ViT geometry. Panel (b): three-run mean ± SD.

$\mathbb{E}_{\mathrm{mask}}$ averages over masks and equals the realized value for deterministic methods. Method superscripts follow the bound notation, e.g., $\mathrm{R}_j^{(\mathrm{min})}$ and $\mathrm{R}_j^{(\mathrm{rand})}$. Since $\|\hat{\boldsymbol{\tau}}_j\|^2 = \widehat{\mathrm{R}}_j \|\boldsymbol{\tau}_j\|^2$, these ratios directly scale the terms in $\widehat{\mathrm{E}}_i$. Pure deletion gives ratios in $[0, 1]$; rescaling can make them larger than one. With $q = 1 - p$, MIN satisfies $\mathrm{R}_j^{(\mathrm{min})} \geq q = \mathbb{E}_{\mathrm{mask}}[\mathrm{R}_j^{(\mathrm{rand})}]$. Figure 4b compares the same ratio for both methods. Appendix C.2 proves the MIN guarantee and RAND expectation.

At equal coordinate budgets, MIN preserves more squared task-vector magnitude than RAND in expectation. In the orthogonal model, RAND and BRE achieve larger bound reductions by removing more energy, alongside the observed NormAccuracy losses in Figure 3. The coefficients in Equation 8 make each vector's effect depend on merge weight as well as retained magnitude.

Proposition 4.1 (Bound ordering for self-aware masks: orthogonal case)**.** *Fix $p, \lambda \in (0, 1)$ and nonzero original task vectors. Assume the original, MIN, and BRE vector sets are each pairwise orthogonal. MIN keeps the $\lceil (1-p)d \rceil$ largest-magnitude coordinates, and BRE further removes upper-tail coordinates. Suppose $\mathrm{R}_j^{(\min)} > 1 - p$ for at least one task, and BRE deletes at least one nonzero coordinate retained by MIN. For Bernoulli masks independent across tasks and coordinates,*

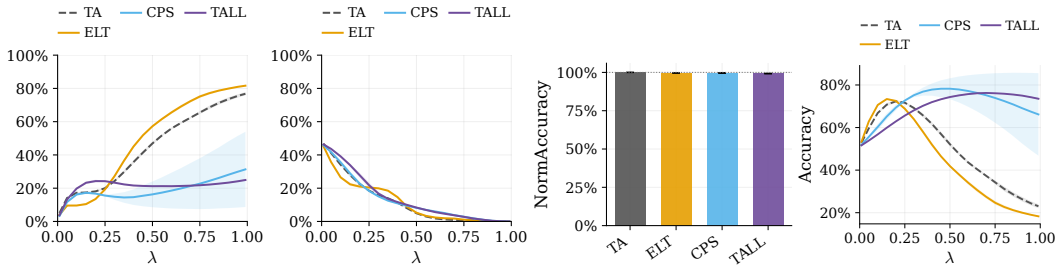
$$\mathbb{E}[\mathcal{B}_i^{(\mathrm{rand})}] < \mathcal{B}_i^{(\min)} < \mathbb{E}[\mathcal{B}_i^{(\mathrm{dare})}], \qquad \mathcal{B}_i^{(\mathrm{bre})} < \mathcal{B}_i^{(\min)}. \tag{12}$$

Under this model, RAND gives $\mathbb{E}[\mathcal{B}_i^{(\mathrm{rand})}] = (1-p)\mathcal{B}_i$, while MIN preserves more squared norm. DARE's rescaling gives $\mathbb{E}[\mathcal{B}_i^{(\mathrm{dare})}] = \mathcal{B}_i/(1-p)$, and BRE's additional deletion lowers the bound relative to MIN. This ordering is consistent with the observed gap reductions. The NormAccuracy curves show the accompanying loss of single-task capability.

Appendix C.2 extends the same comparisons to nonorthogonal vectors using their signed cross-task interactions, proves the orthogonal case as a specialization, and checks the exact conditions across three runs of each model family, connecting the reference case to measured ViT and BERT geometry.

4.3 Global-Aware Sparsification

Empirical observation. Figure 5 shows that CPS and TALL suppress the growth of $\xi_{\mathrm{mg-lfd}}$ at larger weights, while ELT rises more rapidly. TALL can exceed TA at smaller weights before crossing below it. The changes in $\xi_{\mathrm{lfd-tgt}}$ and NormAccuracy are smaller. We examine how the three masking rules change the bound underlying these diagnostic trends.



(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{lfd-tgt}}$ (ViT) (c) NormAccuracy (ViT) (d) Merged accuracy (ViT)

Figure 5: Global-aware masks on ViT: three-run means and one standard deviation across runs. The masks have different effects on the growth of the merged–LFD gap.

CPS: an orthogonal model and its extension. CPS keeps coordinates where a task's magnitude is sufficiently large relative to the aggregate of the other tasks. We again begin with the orthogonal reference case to isolate the effect of coordinate deletion on squared norms. Figure 12a reports the relevant cosines; the subsequent finite comparison restores the measured cross-task terms.

Proposition 4.2 (CPS bound reduction: orthogonal case)**.** *Assume that the original vectors $\{\boldsymbol{\tau}_j\}$ and the CPS vectors $\{\hat{\boldsymbol{\tau}}_j^{(\mathrm{cps})}\}$ are each pairwise orthogonal. For every task i and $0 < \lambda < 1$,*

$$\mathcal{B}_i^{(\mathrm{cps})} = \mathrm{E}_i^{(\mathrm{cps})} \le \mathrm{E}_i = \mathcal{B}_i. \tag{13}$$

The inequality is strict if CPS removes at least one nonzero coordinate from any task vector.

The norm terms clarify how CPS can act differently across merge weights. In E_i, the coefficient on each other-task squared norm relative to the target coefficient is $[\lambda/(1-\lambda)]^2$: it grows from $1/16$ at $\lambda = 0.2$ to 16 at $\lambda = 0.8$. Target-vector retention thus carries more weight at small λ, while deleting energy from other tasks carries more weight at large λ. This shift helps explain why CPS separates most clearly from TA at larger weights in Figure 5a. CPS selects coordinates separately for each task, so its support changes can also shift signed inner products. The balance between norm reduction and these interaction changes determines the finite-vector comparison examined next.

Orthogonality removes the cross-task terms, and coordinate deletion decreases E_i. For nonorthogonal vectors, the squared-norm reduction must cover any increase in $\mathcal{B}_{i,\mathrm{cross}}$. Let Δ_{cps} denote the original quantity minus its CPS counterpart. Figure 6 divides each signed CPS-minus-original change by $\Delta_{\mathrm{cps}}\mathrm{E}_i > 0$: the negative norm contribution and positive interaction contribution sum to a negative bound change, even at each run's least favorable normalized configuration. The bound decreases for every measured task and weight across all three runs of both model families. Appendix C.3 gives the condition and finite checks for CPS.

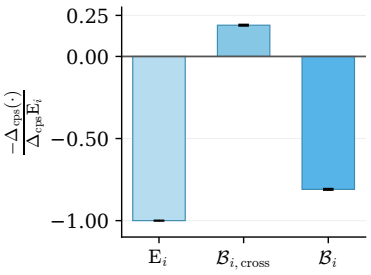


Figure 6: CPS on ViT: signed changes, three-run mean ± SD.

TALL and ELT: shared masking and sign election. TALL applies one common binary mask, so it deletes coordinates directly from the vector inside $\mathcal{B}_i$. ELT uses task-specific sign masks. Write $\mathbb{E}_i$ for expectation over a uniformly chosen target task; RAND/DARE expectations in Section 4.2 are over random masks.

Proposition 4.3 (Bound reductions from shared masking and sign election)**.** *For $T \geq 2$, TALL reduces each bound at $0 < \lambda < 1$; ELT reduces the target-task average when $\lambda \leq 2/T$:*

$$\begin{aligned} \mathcal{B}_i^{(\mathrm{tall})} &\leq \mathcal{B}_i \qquad \textit{for every } i, \\ \mathbb{E}_i[\mathcal{B}_i^{(\mathrm{elt})}] &\leq \mathbb{E}_i[\mathcal{B}_i] \quad \textit{if } \lambda \leq 2/T. \end{aligned} \tag{14}$$

These comparisons require no orthogonality assumption.

ELT decreases the sum of individual squared norms and increases the squared norm of the vector sum. Both changes lower $\mathbb{E}_i[\mathcal{B}_i]$ for $\lambda \leq 2/T$; at larger weights, the growing vector-sum term can reverse the comparison. For eight tasks, the guarantee covers $\lambda \leq 0.25$, while the measured crossover is near 0.32 in both model families. Appendix C.3 gives the expectation identity and exact crossover. The geometric bounds describe the parameter contribution to the merged–LFD gap; the observed diagnostic curves also reflect the model's nonlinear output changes.

4.4 Supplementary Experiments and Mask Composition

The appendices extend the main comparisons with BERT diagnostics, hyperparameter sweeps, complete bound curves, and exact finite-interaction checks (Appendices B, C.2, D.1, and D.2). Appendix Table 3 maps each analytical statement to the observed diagnostics; its arrows summarize changes relative to Task Arithmetic. Applying the ELT result to MIN gives $\mathbb{E}_i[\mathcal{B}_i^{(\mathrm{mel})}] \leq \mathbb{E}_i[\mathcal{B}_i^{(\mathrm{min})}]$ for $\lambda \leq 2/T$. The measured MEL composition also further reduces $\xi_{\mathrm{mg-lfd}}$ in the reported low-weight regime. Appendix E gives the complete comparisons.

4.5 Predictive Value of the Diagnostics

The path analysis suggests a practical test: do diagnostic relationships learned from seven target tasks transfer to the eighth within the same eight-task merge? We match each mask to TA at the same ViT-B/32 task, seed, and interpolation weight across three seeds. For each match, we use

$$y = \frac{a - a_{\mathrm{TA}}}{a_{\mathrm{ft}} - a_{\mathrm{TA}}}, \qquad \mathbf{x} = (\Delta\xi_{\mathrm{mg-lfd}}, \Delta\xi_{\mathrm{lfd-tgt}}, \Delta\mathrm{NormAccuracy})^\top, \tag{15}$$

where a is merged accuracy, a_{ft} is original single-task accuracy, and y is the fraction of TA's remaining gap recovered by the mask. For either path gap, we set $\Delta\xi = \xi^{(\mathrm{ta})} - \xi^{(\mathrm{mask})}$, where the second term is measured after applying the current mask; $\Delta\mathrm{NormAccuracy} = A_i(\hat{\boldsymbol{\tau}}_i)/A_i(\boldsymbol{\tau}_i) - 1$. Here $y > 0$ denotes a gain over TA; positive $\Delta\xi$ denotes reduced disagreement, while negative $\Delta\mathrm{NormAccuracy}$ denotes lost target capability. The 4,000 retained configurations satisfy $a_{\mathrm{ft}} - a_{\mathrm{TA}} > 0.02$.

All eight task vectors remain in each merge. For each held-out target task, ridge fitting and feature scaling use only the other seven tasks' evaluation rows. Every seed, mask, and weight for the target task is excluded from fitting. The merged–LFD gap is the strongest single predictor; adding the other diagnostics improves ranking and fixed-configuration comparisons, as detailed in Appendix F.

Table 2: Predictive value on ViT-B/32 across three seeds. Spearman measures held-out rank correlation; pairwise accuracy compares masks at fixed task, seed, and weight.

Predictors	Spearman ↑	Pairwise (%) ↑
Intercept	-0.157	50.0
Merged–LFD	0.816	73.6
Both path gaps	0.812	78.0
All three	0.856	81.7

4.6 Beyond Sparsification: Orthogonal Task Vectors

The bound also separates squared norms from cross-task interactions for transformed vectors outside the masking family. TSV-Merge (Gargiulo et al., 2025) provides a controlled example: it orthogonalizes the singular bases of rank-truncated matrix components. We compare this transformation with rank truncation alone (lowrank) using the same ViT-B/32 checkpoints, seed 43, eight tasks, and 8-NN evaluation. Figure 7 shows the complete-test results at shared coefficients. At $\lambda = 0.2$, TSV reduces $\xi_{\mathrm{mg-lfd}}$ slightly but trails lowrank by 8.02 accuracy points. At 0.9, it reduces the first-stage disagreement from 0.636 to 0.152 and leads by 45.43 points.

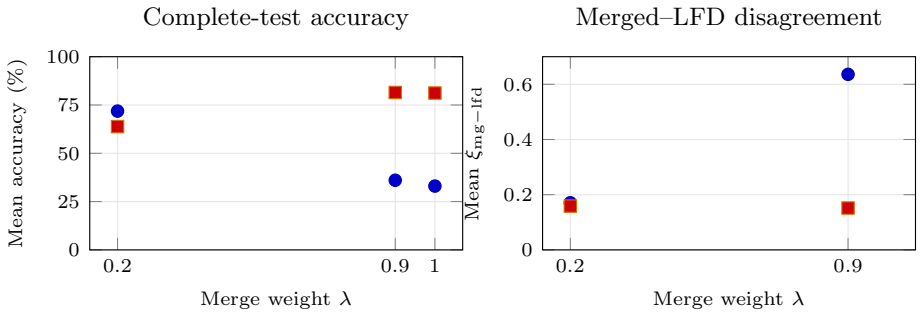


Figure 7: Measured lowrank (blue circles) and TSV (orange squares) results at shared merge weights on eight ViT-B/32 complete test sets (seed 43). LFD disagreement is defined for $\lambda < 1$.

At $\lambda = 0.2$, TSV's smaller first-stage disagreement comes with larger LFD–target disagreement (0.215 to 0.243) and lower NormAccuracy (0.994 to 0.972). At 0.9, $\xi_{\mathrm{lfd-tgt}}$ changes from 0.0135 to 0.0097. The measured task-vector bound term falls by 17.08% at this weight but rises by 2.28% at 0.2; the ideal interaction-only calculation changes sign at $\lambda = 2/T = 0.25$. Validation selects 0.2 for lowrank and 0.9 for TSV, yielding full-test accuracies of 71.84% and 81.48%. Thus the framework identifies both the high-weight first-stage gain and the accompanying single-task capability cost. Appendix G details the finite-precision geometry and fixed-weight controls.

5 Conclusion

LFD provides a target-conditioned path for studying how post-hoc task-vector transformations affect merging. Its two disagreement measures locate changes along that path, while NormAccuracy records the capability retained by each transformed target vector. A representation-gap bound connects the first stage to squared norms and signed interactions; orthogonal reference cases clarify the norm effect, and finite comparisons account for measured geometry. Across ViT and BERT, this view explains distinct sparsification mechanisms and predicts relative gains on held-out ViT tasks. The single-seed TSV comparison shows how the same diagnostics reveal a merge-weight-dependent benefit of orthogonalization together with a modest loss in target capability. Extending these tests across architectures and seeds will clarify how broadly the measured relationships transfer.

AI USE STATEMENT

Generative AI assisted language editing, presentation, and checking mathematical derivations. The authors remain responsible for verifying technical claims, proofs, citations, and experimental details and for the final paper.

ETHICS STATEMENT

Reusing merged models can reduce multi-task training compute. Bias, privacy leakage, and safety vulnerabilities in component models can propagate to merged models; deployment requires task-appropriate safety and fairness evaluation.

REPRODUCIBILITY STATEMENT

Definitions and results appear in Sections 3.2–4.6; proofs and experimental details are in the appendices. Code and scripts will be released upon acceptance.

REFERENCES

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. Semeval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th international workshop on semantic evaluation (SemEval-2017)*, pages 1–14, 2017.

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

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

Ido Dagan, Oren Glickman, and Bernardo Magnini. The pascal recognising textual entailment challenge. In *Machine learning challenges workshop*, pages 177–190. Springer, 2005.

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

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 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)*, pages 4171–4186, 2019.

William B Dolan and Chris Brockett. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the third international workshop on paraphrasing (IWP2005)*, 2005.

Alexey Dosovitskiy. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

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. *Advances in Neural Information Processing Systems*, 37:84746–84776, 2024.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodolà. Task singular vectors: Reducing task interference in model merging. *arXiv preprint arXiv:2412.00081*, 2025.

Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

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

Ruochen Jin, Bojian Hou, Jiancong Xiao, Weijie J Su, and Li Shen. Fine-tuning attention modules only: Enhancing weight disentanglement in task arithmetic. In *The Thirteenth International Conference on Learning Representations*, 2025.

Alex Kendall, Yarin Gal, and Roberto Cipolla. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7482–7491, 2018.

Fanshuang Kong, Richong Zhang, Zhijie Nie, Ziqiao Wang, and Qiang Sun. Fine-tuning aligned classifiers for merging outputs: Towards a superior evaluation protocol in model merging. *arXiv preprint arXiv:2412.13526*, 2024.

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

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

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

Shikun Liu, Edward Johns, and Andrew J Davison. End-to-end multi-task learning with attention. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 1871–1880, 2019.

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

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. *Advances in Neural Information Processing Systems*, 36, 2024.

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

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 2383–2392, 2016.

Ozan Sener and Vladlen Koltun. Multi-task learning as multi-objective optimization. *Advances in neural information processing systems*, 31, 2018.

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 *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *The 2011 international joint conference on neural networks*, pages 1453–1460. IEEE, 2011.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP workshop BlackboxNLP: Analyzing and interpreting neural networks for NLP*, pages 353–355, 2018.

Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jiménez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression. In *Proceedings of the 41st International Conference on Machine Learning*, pages 50268–50287, 2024.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641, 2019.

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

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

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7959–7971, 2022.

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

Prateek Yadav, Derek Tam, Leshem Choshen, Colin A Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models. *Advances in Neural Information Processing Systems*, 36:7093–7115, 2023.

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

Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xiaojun Chen, Xingwei Wang, and Dacheng Tao. Representation surgery for multi-task model merging. In *Proceedings of the 41st International Conference on Machine Learning*, pages 56332–56356, 2024b.

Yan Yang, Yixia Li, Hongru Wang, Xuetao Wei, James Jianqiao Yu, Yun Chen, and Guanhua Chen. Impart: Importance-aware delta-sparsification for improved model compression and merging in llms. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18817–18829, 2025.

Kotaro Yoshida, Yuji Naraki, Takafumi Horie, Ryosuke Yamaki, Ryotaro Shimizu, Yuki Saito, Julian McAuley, and Hiroki Naganuma. Mastering task arithmetic: τ jp as a key indicator for weight disentanglement. In *The Thirteenth International Conference on Learning Representations*, 2025.

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Forty-first International Conference on Machine Learning*, 2024.

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

Yuhang Zhou, Giannis Karamanolakis, Victor Soto, Anna Rumshisky, Mayank Kulkarni, Furong Huang, Wei Ai, and Jianhua Lu. Mergeme: Model merging techniques for homogeneous and heterogeneous moes. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2315–2328, 2025.

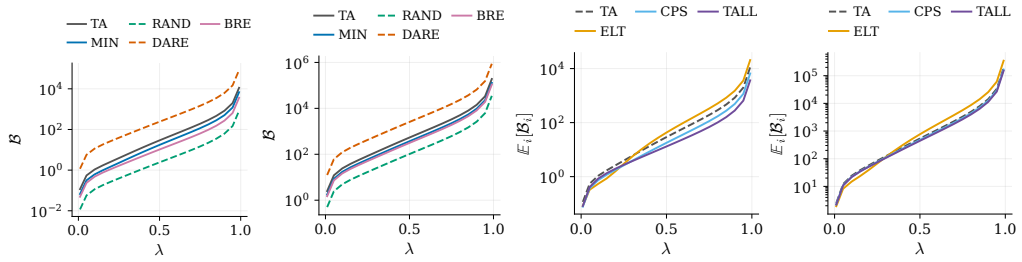
Yuyan Zhou, Liang Song, Bingning Wang, and Weipeng Chen. Metagpt: Merging large language models using model exclusive task arithmetic. *arXiv preprint arXiv:2406.11385*, 2024a.

Zhanpeng Zhou, Zijun Chen, Yilan Chen, Bo Zhang, and Junchi Yan. On the emergence of cross-task linearity in pretraining-finetuning paradigm. In *Forty-first International Conference on Machine Learning*, 2024b.

A Method and Diagnostic Summary

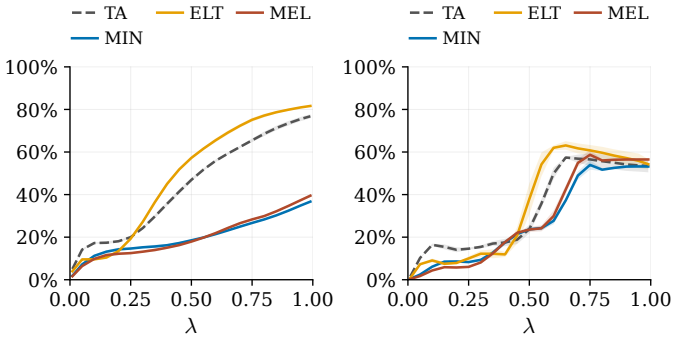
Table 3: Analytical statements and observed LFD-path diagnostic patterns. Masks follow Table 1, resolving boundary ties by exact rank. BRE retains ascending magnitude ranks $\lfloor p_1 d \rfloor + 1$ through $\lfloor p_2 d \rfloor$ and is compared with MIN at the same lower cutoff. The last column reports $\xi_{\mathrm{mg-lfd}}$, $\xi_{\mathrm{lfd-tgt}}$, $A_i(\hat{\boldsymbol{\tau}}_i)$, and final merged accuracy at the reported setting relative to TA. Arrows indicate increases or decreases relative to TA; lower disagreement and higher accuracy indicate improvement. Arrow counts indicate relative empirical magnitude.

Method	**Processed parameter value**	**Geometric statement**	$\xi_{\mathrm{mg-lfd}}$ / $\xi_{\mathrm{lfd-tgt}}$ / $A_i(\hat{\boldsymbol{\tau}}_i)$ / **merge**
TA	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{ta})} = \boldsymbol{\tau}_{i,k}$	reference	$\approx$; $\approx$; $\approx$; $\approx$
MIN	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{min})} = m_{i,k}^{(\mathrm{min})} \boldsymbol{\tau}_{i,k}$	$\mathrm{R}_i^{(\mathrm{min})} \geq 1-p$; conditional bound ordering	$\downarrow$; $\uparrow$; $\approx$; $\uparrow$
BRE	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{bre})} = m_{i,k}^{(\mathrm{bre})} \boldsymbol{\tau}_{i,k}$	$\mathrm{R}_i^{(\mathrm{bre})} \leq \mathrm{R}_i^{(\mathrm{min})}$; signed comparison	$\downarrow\downarrow$; $\uparrow$; $\downarrow$; $\uparrow$
RAND	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{rand})} = z_{i,k} \boldsymbol{\tau}_{i,k}$	$\mathbb{E}[\mathcal{B}^{(\mathrm{rand})}] = (1-p)^2\mathcal{B} + p(1-p)\mathrm{E}$	$\downarrow\downarrow$; $\uparrow\uparrow$; $\downarrow\downarrow$; $\downarrow$
DARE	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{dare})} = z_{i,k}\boldsymbol{\tau}_{i,k}/(1-p)$	$\mathbb{E}[\mathcal{B}^{(\mathrm{dare})}] = \mathcal{B} + p\mathrm{E}/(1-p)$	$\approx$; $\approx$; $\approx$; $\approx$
ELT	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{elt})} = m_{i,k}^{(\mathrm{elt})} \boldsymbol{\tau}_{i,k}$	$\mathbb{E}_i[\mathcal{B}_i^{(\mathrm{elt})}] \leq \mathbb{E}_i[\mathcal{B}_i]$ for $\lambda \leq 2/T$	$\downarrow$; $\approx$; $\downarrow$; $\uparrow$
CPS	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{cps})} = m_{i,k}^{(\mathrm{cps})} \boldsymbol{\tau}_{i,k}$	orthogonal reduction; exact finite comparison	$\downarrow\downarrow\downarrow$; $\approx$; $\downarrow$; $\uparrow\uparrow$
TALL	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{tall})} = m_{k}^{(\mathrm{tall})} \boldsymbol{\tau}_{i,k}$	$\mathcal{B}_i^{(\mathrm{tall})} \leq \mathcal{B}_i$ for all λ	$\downarrow\downarrow$; $\uparrow$; $\approx$; $\uparrow$
MEL	$\hat{\boldsymbol{\tau}}_{i,k}^{(\mathrm{mel})} = m_{i,k}^{(\mathrm{elt})} m_{i,k}^{(\mathrm{min})} \boldsymbol{\tau}_{i,k}$	$\mathbb{E}_i[\mathcal{B}_i^{(\mathrm{mel})}] \leq \mathbb{E}_i[\mathcal{B}_i^{(\mathrm{min})}]$ for $\lambda \leq 2/T$	$\downarrow\downarrow$; $\uparrow$; $\approx$; $\uparrow$



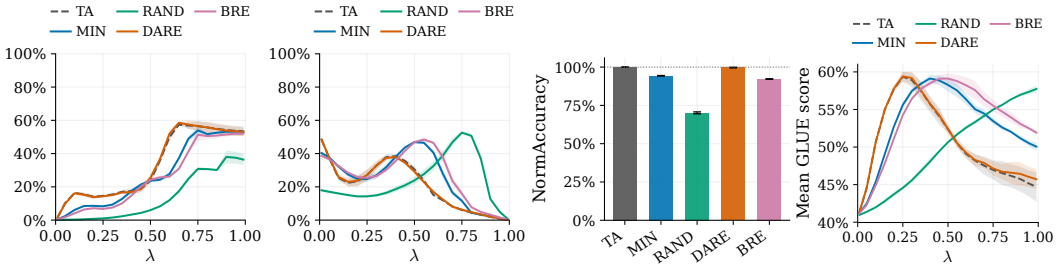
(a) Self-aware, ViT (b) Self-aware, BERT (c) Global-aware, ViT (d) Global-aware, BERT

Figure 8: Geometric evidence from three runs each for ViT and BERT (means and one standard deviation). Panels (a)–(b) show $\mathcal{B}$ for deterministic masks and its exact mask expectation for RAND/DARE (dashed). Panels (c)–(d) show $\mathbb{E}_i[\mathcal{B}_i]$ for global-aware masks, with uniform expectation over target tasks.



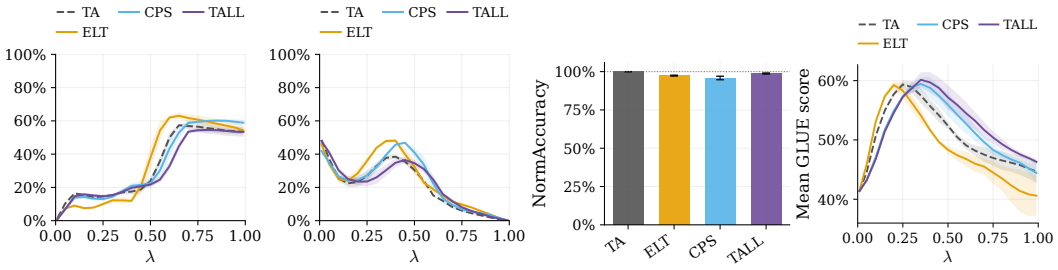
(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{mg-lfd}}$ (BERT)

Figure 9: Observed effect of composing magnitude and sign masks: three-run means and one standard deviation.



(a) $\xi_{\mathrm{mg-lfd}}$ (BERT) (b) $\xi_{\mathrm{lfd-tgt}}$ (BERT) (c) NormAccuracy (BERT) (d) Mean GLUE score (BERT)

Figure 10: Self-aware sparsification on BERT: three-run means and one standard deviation, corresponding to Figure 3.

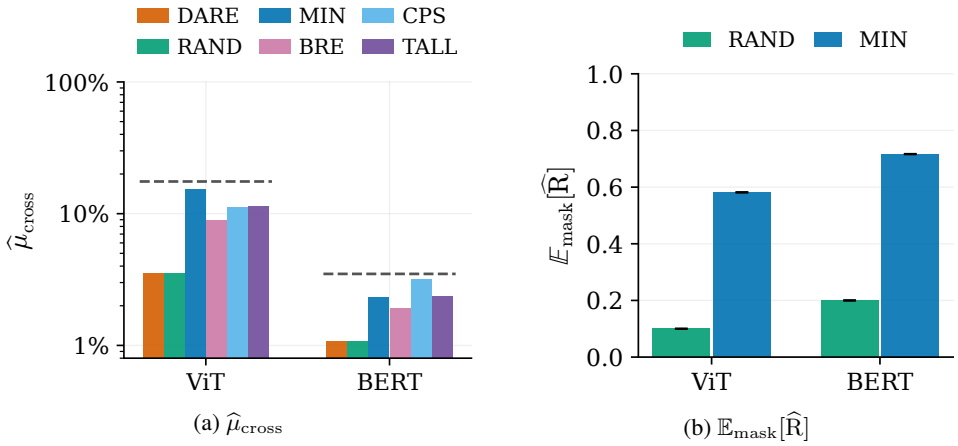


(a) $\xi_{\mathrm{mg-lfd}}$ (BERT) (b) $\xi_{\mathrm{lfd-tgt}}$ (BERT) (c) NormAccuracy (BERT) (d) Mean GLUE score (BERT)

Figure 11: Global-aware sparsification on BERT: three-run means and one standard deviation, corresponding to Figure 5.

B Measured Task-Vector Geometry

We report the geometric measurements used by the method-specific analyses. Figure 12 complements the ViT evidence in Figure 4 with both model families and the complete dedicated cosine comparison.



(a) $\widehat{\mu}_{\mathrm{cross}}$ (b) $\mathbb{E}_{\mathrm{mask}}[\widehat{\mathrm{R}}]$

Figure 12: ViT and BERT geometry. Panel (a) shows dedicated cosine measurements, with original vectors dashed. Panel (b) compares squared-norm ratios at the default masks: three-run means ± SD.

Both model families use the common squared-norm ratio in Equation 11. Figure 12b reports $\mathbb{E}_{\mathrm{mask}}[\mathrm{R}^{(\mathrm{rand})}]$ and $\mathbb{E}_{\mathrm{mask}}[\mathrm{R}^{(\mathrm{min})}]$, with $q = 0.1$ for ViT and $q = 0.2$ for BERT. The expectation is over the mask at fixed task vectors; a deterministic mask has its realized value as its expectation. We average the task-wise ratios equally within each run, then report the mean and sample standard deviation across three runs.

The original vectors are nonzero. A zero processed vector has squared-norm ratio zero. In $\widehat{\mu}_{\mathrm{cross}}$, zero-vector pairs are omitted; the maximum is zero if no nonzero pair exists. Removing the hat gives the original cosine μ_{cross}; method-specific scalars use matching superscripts, such as $\mu_{\mathrm{cross}}^{(\mathrm{min})}$. The cosine plots retain their dedicated measurement coverage, as specified in Appendix D.2.

Equation 18 connects the task-wise squared-norm ratios to the MIN–RAND comparison; Equation 21 relates cosine to interaction magnitude. CPS uses the same signed finite-comparison approach. ELT and MEL use expectation over target tasks and the decomposition in Equation 28 (Appendix C.3).

Interaction magnitudes across weights. We compute full-vector float64 Gram quantities for three runs of each model family, eight tasks, the default masks, and all 21 interpolation weights. TA and the six deterministic masks give 7,056 task–method–run–weight observations. Table 4 reports the maximum $|\widehat{\mathcal{B}}_{i,\mathrm{cross}}|/\widehat{\mathrm{E}}_i$ across tasks and runs, both over the full grid and over the six weights

$\lambda \in \{0.05, 0.10, 0.15, 0.20, 0.25, 0.30\}$. These measurements describe how interaction strength varies with the mask and weight; the signed ordering conditions are checked in Appendix C.2.

Table 4: Maximum $100|\widehat{\mathcal{B}}_{i,\mathrm{cross}}|/\widehat{\mathrm{E}}_i$ across tasks and three runs per model family, for each weight range.

	$\lambda = 0.05$–0.30		All 21 weights	
Method	BERT	ViT	BERT	ViT
TA	11.37	10.27	12.56	43.60
MIN	6.69	10.28	7.76	49.95
BRE	5.64	2.75	6.53	9.43
CPS	8.52	5.22	8.64	24.17
ELT	156.13	50.48	177.07	240.56
MEL	60.24	18.43	68.60	100.18

Magnitude masks have smaller relative interactions in the lower-weight slice. At larger weights, the signed change can reinforce a bound reduction, as shown by the BRE comparison in Appendix C.2. ELT and MEL use the exact task-expectation identity in Appendix C.3.

C Proof

C.1 Proof of Theorem 3.1

Proof. For fixed x, set $a = \boldsymbol{\theta}_0 + \hat{\boldsymbol{\tau}}_i$, $b = \boldsymbol{\theta}_0 + \lambda \sum_{j \neq i} \hat{\boldsymbol{\tau}}_j/(1-\lambda)$, and $z = \lambda a + (1-\lambda)b = \boldsymbol{\theta}_{\mathrm{merged}}$. The Hessian norm is the bilinear operator norm, so $\|D^2 f(x;\theta)[u,u]\| \leq M\|u\|^2$. Taylor's formula with an integral remainder, expanded about z, gives

$$f(x;a) = f(x;z) + Df(x;z)(a-z) + R_a, \qquad \|R_a\| \leq \tfrac{M}{2}\|a-z\|^2,$$

and the analogous expression for b. The linear terms cancel after weighting by λ and $1-\lambda$. Since $a - z = (1-\lambda)(a-b)$ and $b - z = -\lambda(a-b)$,

$$\begin{aligned}\|f_{\mathrm{lfd}}(x) - f_{\mathrm{merged}}(x)\| &\leq \tfrac{M}{2}\{\lambda\|a-z\|^2 + (1-\lambda)\|b-z\|^2\} \\ &= \tfrac{M}{2}\lambda(1-\lambda)\|a-b\|^2 = \tfrac{M}{2}\widehat{\mathcal{B}}_i.\end{aligned}$$

This proves the stated bound directly under the uniform Hessian condition. □

First-order agreement and the endpoint. Expanding about $\boldsymbol{\theta}_0$ shows that both merged and LFD outputs have linear term $\lambda Df(x;\boldsymbol{\theta}_0)\sum_j \hat{\boldsymbol{\tau}}_j$. For a nonlinear model, convergence of LFD to target follows from $(1-\lambda)\|f_{\mathrm{others}}\| \to 0$; bounded non-target outputs suffice. For the linearized model, in contrast, $(1-\lambda)f^{\mathrm{lin}}_{\mathrm{others}} = (1-\lambda)f(x;\boldsymbol{\theta}_0) + \lambda Df(x;\boldsymbol{\theta}_0)\sum_{j\neq i}\boldsymbol{\tau}^{\mathrm{lin}}_j$. Its non-target contribution generally remains nonzero. Functional additivity and target convergence are therefore separate properties. The experiments evaluate finite interpolation weights; their feature interpolation uses the raw model features.

C.2 Self-aware bound ordering beyond orthogonality

We derive the same three comparisons as Proposition 4.1 while retaining the signed cross-task contribution $\widehat{\mathcal{B}}_{i,\mathrm{cross}}$ defined in Equation 8. Throughout this subsection, $p, \lambda \in (0,1)$, MIN retains $k = \lceil (1-p)d \rceil$ largest-magnitude coordinates with consistent tie-breaking, and BRE removes an additional upper tail from that support.

Squared-norm ratios and their bound contribution. For every method, Equation 11 gives

$$\widehat{\mathrm{E}}_i = \lambda(1-\lambda)\widehat{\mathrm{R}}_i\|\boldsymbol{\tau}_i\|^2 + \frac{\lambda^3}{1-\lambda}\sum_{j\neq i}\widehat{\mathrm{R}}_j\|\boldsymbol{\tau}_j\|^2. \tag{16}$$

The task-wise ratios determine this weighted quantity; their equal-task average $\widehat{\mathrm{R}}$ is used for visualization. The next result is a property of MIN under the same ratio definition.

Proposition C.1 (MIN squared-norm ratio)**.** *For each nonzero $\boldsymbol{\tau}_j \in \mathbb{R}^d$, let MIN retain its $k = \lceil qd \rceil$ largest-magnitude coordinates, where $q \in (0, 1]$. Then*

$$\mathrm{R}_j^{(\min)} = \frac{\|\hat{\boldsymbol{\tau}}_j^{(\min)}\|^2}{\|\boldsymbol{\tau}_j\|^2} \geq \frac{k}{d} \geq q. \tag{17}$$

For $k < d$, equality in the first inequality holds exactly when all squared coordinate magnitudes are equal.

Proof. Order the squared magnitudes of $\boldsymbol{\tau}_j$ as $w_1 \geq \cdots \geq w_d \geq 0$. The average of the largest k is at least the full-vector average: $\sum_{\ell=1}^{k} w_\ell/k \geq \sum_{\ell=1}^{d} w_\ell/d$. Dividing by the positive total squared norm proves the claim; equality for $k < d$ forces every w_ℓ to be equal. □

Applying this inequality to each task vector with the positive weights in E_i gives

$$(1-p)\mathrm{E}_i \leq \frac{k}{d}\mathrm{E}_i \leq \mathrm{E}_i^{(\min)} \leq \mathrm{E}_i, \qquad \mathrm{E}_i^{(\mathrm{bre})} \leq \mathrm{E}_i^{(\min)}.$$

The first comparison has the explicit margin

$$\begin{aligned} \mathrm{E}_i^{(\min)} - (1-p)\mathrm{E}_i = \lambda(1-\lambda)[\mathrm{R}_i^{(\min)} - (1-p)]\|\boldsymbol{\tau}_i\|^2 \\ + \frac{\lambda^3}{1-\lambda}\sum_{j\neq i}[\mathrm{R}_j^{(\min)} - (1-p)]\|\boldsymbol{\tau}_j\|^2. \end{aligned} \tag{18}$$

Thus magnitude concentration makes this margin positive. DARE's margin $\mathrm{E}_i/(1-p) - \mathrm{E}_i^{(\min)}$ is positive whenever $\mathrm{E}_i > 0$. BRE's margin $\mathrm{E}_i^{(\min)} - \mathrm{E}_i^{(\mathrm{bre})}$ is positive when its additional deletion removes a nonzero coordinate.

Exact random-mask expectations.

Lemma C.2 (RAND and DARE expectations)**.** *For Bernoulli masks independent across tasks and coordinates and any fixed nonzero original task vectors, the same squared-norm ratio satisfies $\mathbb{E}_{\mathrm{mask}}[\mathrm{R}_j^{(\mathrm{rand})}] = 1 - p$ and $\mathbb{E}_{\mathrm{mask}}[\mathrm{R}_j^{(\mathrm{dare})}] = 1/(1-p)$. The exact bound expectations are*

$$\begin{aligned} \mathbb{E}[\mathcal{B}_i^{(\mathrm{rand})}] &= (1-p)\mathrm{E}_i + (1-p)^2\mathcal{B}_{i,\mathrm{cross}}, \\ \mathbb{E}[\mathcal{B}_i^{(\mathrm{dare})}] &= \frac{\mathrm{E}_i}{1-p} + \mathcal{B}_{i,\mathrm{cross}}. \end{aligned} \tag{19}$$

Proof. For a mask entry $z_{j,k} \sim \mathrm{Bernoulli}(1-p)$, $\mathbb{E}[z_{j,k}^2] = 1-p$, while independence gives $\mathbb{E}[z_{j,k}z_{\ell,k}] = (1-p)^2$ for $j \neq \ell$. Apply these identities to the squared-norm and cross-task terms in Equation 8 to obtain the RAND expectation. DARE divides each retained value by $1-p$, so its quadratic bound term is divided by $(1-p)^2$. □

Equivalently, the expectations are $(1-p)^2\mathcal{B}_i + p(1-p)\mathrm{E}_i$ and $\mathcal{B}_i + p\mathrm{E}_i/(1-p)$, respectively. These identities retain all interactions in the original vectors.

Exact finite comparisons.

Proposition C.3 (Finite-interaction extension of Proposition 4.1)**.** *For arbitrary original task vectors and the masks specified above, the three strict comparisons in Equation 12 are respectively equivalent to*

$$\begin{aligned} \mathrm{E}_i^{(\min)} - (1-p)\mathrm{E}_i &> (1-p)^2\mathcal{B}_{i,\mathrm{cross}} - \mathcal{B}_{i,\mathrm{cross}}^{(\min)}, \\ \frac{\mathrm{E}_i}{1-p} - \mathrm{E}_i^{(\min)} &> \mathcal{B}_{i,\mathrm{cross}}^{(\min)} - \mathcal{B}_{i,\mathrm{cross}}, \\ \mathrm{E}_i^{(\min)} - \mathrm{E}_i^{(\mathrm{bre})} &> \mathcal{B}_{i,\mathrm{cross}}^{(\mathrm{bre})} - \mathcal{B}_{i,\mathrm{cross}}^{(\min)}. \end{aligned} \tag{20}$$

Each condition compares a squared-norm advantage with the signed interaction change that can offset it. A negative right-hand side strengthens the comparison.

Proof. Write each deterministic bound term as $\mathcal{B}_i = \mathrm{E}_i + \mathcal{B}_{i,\mathrm{cross}}$, with the corresponding method superscripts, and substitute the expectations in Equation 19. The left side minus the right side of the first two conditions equals $\mathcal{B}_i^{(\min)} - \mathbb{E}[\mathcal{B}_i^{(\mathrm{rand})}]$ and $\mathbb{E}[\mathcal{B}_i^{(\mathrm{dare})}] - \mathcal{B}_i^{(\min)}$, respectively. For the third condition,

$$\mathcal{B}_i^{(\min)} - \mathcal{B}_i^{(\mathrm{bre})} = \mathrm{E}_i^{(\min)} - \mathrm{E}_i^{(\mathrm{bre})} - (\mathcal{B}_{i,\mathrm{cross}}^{(\mathrm{bre})} - \mathcal{B}_{i,\mathrm{cross}}^{(\min)}).$$

Positivity of these exact differences proves each equivalence. Replacing strict inequalities on both sides of each equivalence by non-strict inequalities gives the corresponding non-strict comparisons. □

Recovering the orthogonal case.

Proof of Proposition 4.1. Pairwise orthogonality within the original, MIN, and BRE sets makes their three cross-task contributions zero. Equation 20 then requires only positive squared-norm margins. Equation 18 is strictly positive when one task has $\mathrm{R}_j^{(\min)} > 1 - p$, because every task has a positive weight for $0 < \lambda < 1$. Also, $\mathrm{E}_i/(1-p) > \mathrm{E}_i \geq \mathrm{E}_i^{(\min)}$ for nonzero original vectors. Finally, BRE's additional deletion of a nonzero retained coordinate gives $\mathrm{E}_i^{(\mathrm{bre})} < \mathrm{E}_i^{(\min)}$. Applying Proposition C.3 proves the stated ordering. The random-mask expectations follow from the original vectors; individual realizations can have nonzero cross terms. □

If top-magnitude retention or additional deletion has zero margin, the corresponding RAND–MIN or BRE–MIN comparison becomes non-strict. The MIN–DARE comparison remains strict for nonzero original vectors and $p > 0$.

Relation to measured cosines. For any processed vector set,

$$|\widehat{\mathcal{B}}_{i,\mathrm{cross}}| \leq (T-1)\widehat{\mu}_{\mathrm{cross}}\widehat{\mathrm{E}}_i. \tag{21}$$

To see this, set $c_i = \sqrt{\lambda(1-\lambda)}$ and $c_j = -\lambda\sqrt{\lambda/(1-\lambda)}$ for $j \neq i$. Bounding each cross-task inner product by its cosine and applying Cauchy–Schwarz gives

$$|\widehat{\mathcal{B}}_{i,\mathrm{cross}}| \leq \widehat{\mu}_{\mathrm{cross}}\big[(\sum_j |c_j|\|\hat{\boldsymbol{\tau}}_j\|)^2 - \widehat{\mathrm{E}}_i\big] \leq (T-1)\widehat{\mu}_{\mathrm{cross}}\widehat{\mathrm{E}}_i.$$

This provides a conservative certificate when only magnitudes and maximum cosines are available. The exact conditions in Equation 20 use the signed Gram contributions, including changes that reinforce the squared-norm advantage.

Three-run verification. We evaluate Equation 20 using full-vector float64 Gram quantities at the default masks, for three runs each of ViT and BERT, eight tasks, and 21 weights. Each of the three exact conditions holds in all 1,008 task–run–weight comparisons (504 per model family), matching the directly computed orderings. The calculation uses the RAND/DARE mask expectations for each run. At every evaluated ViT weight $\lambda \geq 0.55$, BRE reduces both the weighted squared norms and the signed cross-task contribution relative to MIN, for every task and run. Thus the interaction change strengthens the BRE–MIN ordering in this regime. Table 4 separately reports interaction magnitudes across the grid.

C.3 Global-aware masks: exact identities and comparisons

We follow the order of Section 4.3. All masks are fixed as λ varies. The CPS comparison uses the same squared-norm term E_i and signed cross-task term $\mathcal{B}_{i,\mathrm{cross}}$ as the self-aware analysis. Expectations $\mathbb{E}_i$ average uniformly over the T target tasks.

CPS: exact finite comparison.

Proposition C.4 (CPS bound reduction with signed interactions)**.** *For any task vectors, task i, and $0 < \lambda < 1$,*

$$\mathcal{B}_i^{(\mathrm{cps})} \leq \mathcal{B}_i \quad \Longleftrightarrow \quad \mathrm{E}_i - \mathrm{E}_i^{(\mathrm{cps})} \geq \mathcal{B}_{i,\mathrm{cross}}^{(\mathrm{cps})} - \mathcal{B}_{i,\mathrm{cross}}. \tag{22}$$

Strict inequalities are equivalent as well.

The left-hand side of the condition is the weighted squared norm removed by CPS, which is nonnegative. The right-hand side is the signed change in cross-task interactions. Thus the condition compares the norm reduction with the interaction increase it must cover.

Proof. Subtract the two instances of Equation 8:

$$\mathcal{B}_i - \mathcal{B}_i^{(\mathrm{cps})} = (\mathrm{E}_i - \mathrm{E}_i^{(\mathrm{cps})}) - (\mathcal{B}_{i,\mathrm{cross}}^{(\mathrm{cps})} - \mathcal{B}_{i,\mathrm{cross}}).$$

This gives both equivalences directly, as in Proposition C.3. □

Recovering the orthogonal case.

Proof of Proposition 4.2. Pairwise orthogonality within the original and CPS vector sets gives $\mathcal{B}_{i,\mathrm{cross}} = \mathcal{B}_{i,\mathrm{cross}}^{(\mathrm{cps})} = 0$. Since CPS only deletes coordinates, $\|\hat{\boldsymbol{\tau}}_j^{(\mathrm{cps})}\|^2 \le \|\boldsymbol{\tau}_j\|^2$ for every j. All weights in E_i are positive for $0 < \lambda < 1$, yielding Equation 13. Deleting a nonzero coordinate makes the inequality strict for every target task. □

CPS can introduce cross-task interactions between originally orthogonal vectors. For example, $(1, 1, 1)$ and $(4, -2, -2)$ have inner product zero, but CPS with $r = 1/2$ gives $(0, 1, 1)$ and $(4, -2, -2)$, whose inner product is -4. At $\lambda = 1/2$, each task's bound increases from 6.75 to 8.5.

A condition covering every weight. To check Equation 22 over all $0 < \lambda < 1$, reuse the difference notation from the main text: Δ_{cps} means a quantity evaluated on $\{\boldsymbol{\tau}_j\}$ minus the same quantity on $\{\hat{\boldsymbol{\tau}}_j^{(\mathrm{cps})}\}$. For example, $\Delta_{\mathrm{cps}}\|\boldsymbol{\tau}_i\|^2 = \|\boldsymbol{\tau}_i\|^2 - \|\hat{\boldsymbol{\tau}}_i^{(\mathrm{cps})}\|^2 \ge 0$. The following conditions are independent of λ:

$$\begin{aligned} &\Delta_{\mathrm{cps}}\Big\|\sum_{j\neq i}\boldsymbol{\tau}_j\Big\|^2 \ge 0, \\ &\Delta_{\mathrm{cps}}\Big\langle \boldsymbol{\tau}_i, \sum_{j\neq i}\boldsymbol{\tau}_j\Big\rangle \le \sqrt{\Delta_{\mathrm{cps}}\|\boldsymbol{\tau}_i\|^2\ \Delta_{\mathrm{cps}}\Big\|\sum_{j\neq i}\boldsymbol{\tau}_j\Big\|^2}. \end{aligned} \tag{23}$$

The aggregate of the other tasks must lose squared norm, and the loss of its alignment with the target must be no larger than the geometric mean of their two norm reductions. These conditions guarantee $\mathcal{B}_i^{(\mathrm{cps})} \le \mathcal{B}_i$ throughout $(0, 1)$.

Proof. Expanding Equation 7 before and after CPS gives

$$\begin{aligned} \mathcal{B}_i - \mathcal{B}_i^{(\mathrm{cps})} = \lambda(1-\lambda)\Delta_{\mathrm{cps}}\|\boldsymbol{\tau}_i\|^2 + \frac{\lambda^3}{1-\lambda}\Delta_{\mathrm{cps}}\Big\|\sum_{j\neq i}\boldsymbol{\tau}_j\Big\|^2 \\ - 2\lambda^2\Delta_{\mathrm{cps}}\Big\langle \boldsymbol{\tau}_i, \sum_{j\neq i}\boldsymbol{\tau}_j\Big\rangle. \end{aligned} \tag{24}$$

The arithmetic–geometric mean inequality bounds the first two terms below by $2\lambda^2$ times the square root in Equation 23. Subtracting the last term leaves a nonnegative difference. If the second inequality in Equation 23 is strict, the bound decreases strictly for every weight. □

In the orthogonal case, the alignment difference is zero and the aggregate norm reduction equals $\sum_{j\neq i}\Delta_{\mathrm{cps}}\|\boldsymbol{\tau}_j\|^2 \ge 0$. Thus this extension recovers the same main-text comparison. It also verifies the signed condition in Equation 22 simultaneously for all weights.

TALL: contraction of the full bound.

Proof of the TALL comparison in Proposition 4.3. The common binary mask $\boldsymbol{m}^{(\mathrm{tall})}$ commutes with summation, hence

$$\mathcal{B}_i^{(\mathrm{tall})} = \frac{\lambda}{1-\lambda}\Big\|\boldsymbol{m}^{(\mathrm{tall})} \odot \Big(\boldsymbol{\tau}_i - \lambda \sum_j \boldsymbol{\tau}_j\Big)\Big\|^2 \le \frac{\lambda}{1-\lambda}\Big\|\boldsymbol{\tau}_i - \lambda \sum_j \boldsymbol{\tau}_j\Big\|^2 = \mathcal{B}_i.$$

The inequality is strict exactly when the mask deletes a nonzero coordinate of the vector inside the norm. □

ELT: an exact expectation over target tasks. For any fixed processed vectors, uniform expectation over the task index gives

$$\mathbb{E}_i[\widehat{\mathcal{B}}_i] = \frac{\lambda}{T(1-\lambda)}\left[\sum_i \|\hat{\boldsymbol{\tau}}_i\|^2 + \lambda(T\lambda - 2)\Big\|\sum_i \hat{\boldsymbol{\tau}}_i\Big\|^2\right]. \tag{25}$$

This follows by expanding $\widehat{\mathcal{B}}_i = \lambda\|\hat{\boldsymbol{\tau}}_i - \lambda\sum_j \hat{\boldsymbol{\tau}}_j\|^2/(1-\lambda)$ and averaging over i. The expectation introduces no randomness in ELT or the task vectors.

Proof of the ELT comparison in Proposition 4.3. ELT keeps values agreeing with the sign of the task-vector sum. At a coordinate with nonzero total, write the sum of magnitudes with the elected sign as A and with the opposite sign as Q. Then $A \ge Q$, the original sum has magnitude $A - Q$, and the retained sum has magnitude A. At zero total, either tie choice preserves or increases the squared sum. Our implementation breaks zero-total ties by sign vote, then by the positive sign if that vote ties. Therefore,

$$\sum_i \|\hat{\boldsymbol{\tau}}_i^{(\mathrm{elt})}\|^2 \le \sum_i \|\boldsymbol{\tau}_i\|^2, \qquad \Big\|\sum_i \hat{\boldsymbol{\tau}}_i^{(\mathrm{elt})}\Big\|^2 \ge \Big\|\sum_i \boldsymbol{\tau}_i\Big\|^2. \tag{26}$$

For $\lambda \le 2/T$, the coefficient $\lambda(T\lambda - 2)$ in Equation 25 is nonpositive. Both changes reduce $\mathbb{E}_i[\mathcal{B}_i]$, proving the comparison. Removal of any nonzero value makes it strict. □

For larger weights, let $L = \sum_i(\|\boldsymbol{\tau}_i\|^2 - \|\hat{\boldsymbol{\tau}}_i^{(\mathrm{elt})}\|^2)$ be the lost squared norm and $G = \|\sum_i \hat{\boldsymbol{\tau}}_i^{(\mathrm{elt})}\|^2 - \|\sum_i \boldsymbol{\tau}_i\|^2$ the gain in squared sum. Both are nonnegative, and

$$\begin{aligned} \mathbb{E}_i[\mathcal{B}_i^{(\mathrm{elt})} - \mathcal{B}_i] &= \frac{\lambda}{T(1-\lambda)}[-L + \lambda(T\lambda - 2)G], \\ \lambda_\star &= \frac{1 + \sqrt{1 + TL/G}}{T} \qquad (G > 0). \end{aligned} \tag{27}$$

The difference is negative below $\lambda_\star$ and positive above it within $(0, 1)$. For $G = 0$, it is nonpositive throughout. This expectation result allows individual increases: for one-dimensional vectors $(1, 10, -9)$ and $\lambda = 1/5$, the first task's bound changes from 0.09 to 0.36 while its expectation over tasks decreases.

An equivalent decomposition separates task-vector dispersion from the squared mean norm. With $\overline{\hat{\boldsymbol{\tau}}} = T^{-1}\sum_i \hat{\boldsymbol{\tau}}_i$,

$$\mathbb{E}_i[\widehat{\mathcal{B}}_i] = \frac{\lambda}{1-\lambda}\Bigg[\underbrace{\frac{1}{T}\sum_i \|\hat{\boldsymbol{\tau}}_i - \overline{\hat{\boldsymbol{\tau}}}\|^2}_{\mathrm{Var}(\hat{\boldsymbol{\tau}})} + (T\lambda - 1)^2\|\overline{\hat{\boldsymbol{\tau}}}\|^2\Bigg]. \tag{28}$$

ELT reduces dispersion while increasing the squared mean norm; the latter term explains the reversal at larger weights. Figure 13 reports both components on ViT.

MEL: applying ELT to the MIN vectors. Applying the same proof to $\{\hat{\boldsymbol{\tau}}_j^{(\min)}\}$ gives

$$\mathbb{E}_i[\mathcal{B}_i^{(\mathrm{mel})}] \le \mathbb{E}_i[\mathcal{B}_i^{(\min)}], \qquad 0 < \lambda < 1, \quad \lambda \le 2/T. \tag{29}$$

Equation 27 also applies with MIN as the reference. No orthogonality assumption enters either result.

Three-run verification. We use the existing full-vector geometry for three runs each of ViT and BERT, eight tasks, the default masks, and all 21 weights. The CPS signed condition in Equation 22 holds strictly in all 1,008 task–run–weight comparisons. Equation 23 also holds strictly for all 48 task–run vector sets, certifying the comparison over the full interval $(0, 1)$. Table 5 reports how close these conditions are to their thresholds. The orthogonality measurements motivate the main-text idealization; the signed conditions establish the comparison for the measured nonorthogonal vectors.

TALL reduces all 1,008 task–run–weight bounds. The expectation identity reconstructs the task-wise results to relative residual below 2×10^{-15}. ELT's crossover is near 0.317 on ViT and 0.321 on BERT; its task expectation is smaller at every evaluated weight through 0.30 and larger from 0.35 onward. MEL–MIN crossovers are near 0.380 and 0.356, respectively. Figure 8 shows the complete bound curves.

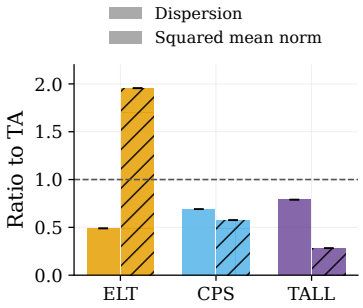


Figure 13: ViT dispersion and squared mean norm relative to TA: three-run means and one standard deviation, corresponding to Equation 28.

Table 5: CPS conditions across three runs per model family. Maxima are over all covered tasks and runs, and also all 21 weights for the finite condition. Both normalized quantities must be at most one. All individual and other-task aggregate squared-norm reductions are positive.

Quantity	ViT	BERT
$-\Delta_{\mathrm{cps}}\mathcal{B}_{i,\mathrm{cross}}/\Delta_{\mathrm{cps}}\mathrm{E}_i$ (finite condition)	0.195	0.493
$\dfrac{\Delta_{\mathrm{cps}}\langle \boldsymbol{\tau}_i, \sum_{j\neq i}\boldsymbol{\tau}_j\rangle}{\sqrt{\Delta_{\mathrm{cps}}\lVert\boldsymbol{\tau}_i\rVert^2\ \Delta_{\mathrm{cps}}\lVert\sum_{j\neq i}\boldsymbol{\tau}_j\rVert^2}}$ (all weights)	0.321	0.540
Strict finite comparisons	504/504	504/504
Strict all-weight conditions	24/24	24/24

D Experiment Setup

D.1 Experiment Setup for CV

Dataset Our evaluation follows the eight image-classification tasks used in task arithmetic (Ilharco et al., 2023): Cars (Krause et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), GTSRB (Stallkamp et al., 2011), MNIST (LeCun, 1998), RESISC45 (Cheng et al., 2017), SUN397 (Xiao et al., 2016), and SVHN (Netzer et al., 2011).

Training We fine-tuned the CLIP (Radford et al., 2021) ViT-B/32 (Dosovitskiy, 2020) image encoder on each dataset's training split. The NTK illustration in Figure 2 uses the implementation[1] released by Ortiz-Jimenez et al. (2024). The reported CV runs use AdamW with learning rate 10^{-5}, weight decay 0.1, a cosine learning-rate schedule, and 500 warmup steps. For ViT-B/32, the batch size is 64 with gradient accumulation 1; the same code uses batch size 32 and gradient accumulation 4 for ViT-L/14. Task-specific epoch counts are Cars 35, DTD 76, EuroSAT 12, GTSRB 11, MNIST 5, RESISC45 15, SUN397 14, and SVHN 4. Train/validation splits are fixed across runs. The eight image datasets use heterogeneous split conventions. We save the initial and final checkpoints and construct every CV task vector from this pair.

CV prompt templates For CLIP classifier construction, the class name c is inserted into the following templates. Cars uses `a photo of a/the/my {c}., i love my {c}!`,

[1] https://github.com/gortizji/tangent_task_arithmetic

and `a photo of my dirty/clean/new/old {c}.`; DTD uses `a photo of a/the {c} texture/pattern/thing/object.`; EuroSAT uses `a centered satellite photo of {c}/a {c}/the {c}.`; GTSRB uses `a zoomed in/centered/close up photo of a "{c}" traffic sign.`; MNIST and SVHN use `a photo of the number: "{c}".`; RESISC45 uses `satellite/aerial imagery/photo/view of {c}/a {c}/the {c}.`; and SUN397 uses `a photo of a {c}.` and `a photo of the {c}.`.

Evaluation CLIP-based image classification commonly uses a head initialized from text-encoder caption embeddings and refined during fine-tuning (Ilharco et al., 2023; Wortsman et al., 2022). Merged representations can differ from those used to train such heads, causing fixed-head evaluations to underestimate merged performance (Kong et al., 2024; Wei et al., 2025; Yang et al., 2024b). We therefore fit an object-specific KNN classifier for each of f_{merged}, f_{lfd}, f_{target}, and f_{others}. For each task and object, eight training examples per class provide the representation exemplars.

Hyperparameter Our hyperparameter settings are as follows:

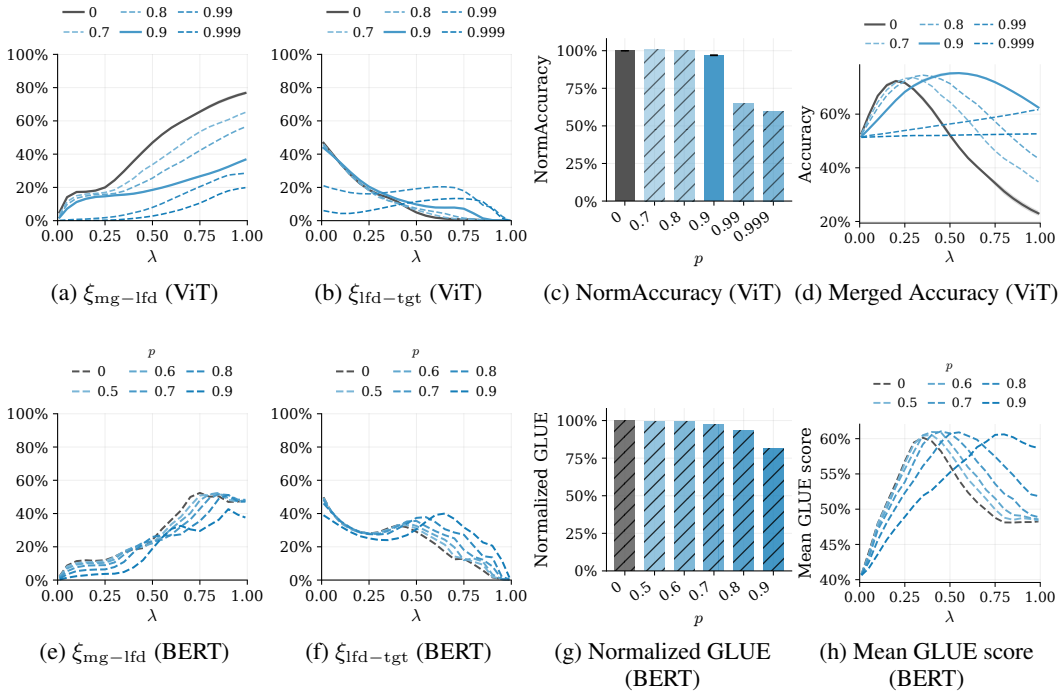
- λ**.** We evaluate interpolation weights $\{0.01\} \cup \{0.05n\}_{n=1}^{19} \cup \{0.99\}$. The denominator $1 - \lambda$ in f_{others} restricts this path to $\lambda < 1$. Observations at 0.99 concern that evaluated weight (Section 3.2).
- **MIN ($\hat{\boldsymbol{\tau}}^{(\min)}$) and RAND ($\hat{\boldsymbol{\tau}}^{(\text{rand})}$).** Dropout rate p is set to $\{0.7, 0.8, 0.9, 0.99, 0.999\}$. Figures 14 (a)–(d) and 15 (a)–(d) report the results for MIN and RAND, respectively. For MIN, increasing p from 0.7 to 0.9 steadily reduces $\xi_{\text{mg}-\text{lfd}}$, while $\xi_{\text{lfd}-\text{tgt}}$ and NormAccuracy remain nearly unchanged. When p reaches 0.99 or 0.999, NormAccuracy declines significantly, indicating that task-critical parameter values are being discarded. For RAND, increasing p reduces $\xi_{\text{mg}-\text{lfd}}$, $\xi_{\text{lfd}-\text{tgt}}$, and NormAccuracy together, so the distance gains are offset by the loss in NormAccuracy and merged accuracy never exceeds the value before RAND. Accordingly, the main-text metrics for MIN, RAND, and variants using them are reported with $p = 0.9$.
- **BRE ($\hat{\boldsymbol{\tau}}^{(\text{bre})}$).** We set $p_1 = 0.9$ and $p_2 = 0.99$, matching the recommended hyperparameters in the original paper (Davari and Belilovsky, 2024). This discards 91% of parameter values in total, making the sparsity comparable to MIN ($\hat{\boldsymbol{\tau}}^{(\min)}$) and RAND ($\hat{\boldsymbol{\tau}}^{(\text{rand})}$).
- **CPS ($\hat{\boldsymbol{\tau}}^{(\text{cps})}$).** Sparsification threshold r is set to $\{0.2, 0.3, 0.4, 0.5, 0.6\}$, which aligns with the search range recommended in the original paper (Wang et al., 2024). Figure 17 reports the results for CPS. Different values of r exhibit consistent trends across evaluation metrics and yield nearly identical final accuracy for f_{merged}. We therefore select r mainly based on the behavior of TALL ($\hat{\boldsymbol{\tau}}^{(\text{tall})}$).
- **TALL ($\hat{\boldsymbol{\tau}}^{(\text{tall})}$).** Consistency threshold c is set to 2, matching the default value in the original paper (Wang et al., 2024). Since TALL is built on CPS ($\hat{\boldsymbol{\tau}}^{(\text{cps})}$), Figure 18 reports the performance variation of TALL with $r = \{0.2, 0.3, 0.4, 0.5, 0.6\}$ and $c = 2$. The setting $r = 0.5$ gives the most balanced merged-accuracy curve, so we use it as the CV default.

D.2 Experiment Setup for NLP

Dataset Following DARE (Yu et al., 2024), we evaluate eight tasks from the GLUE benchmark: CoLA (Warstadt et al., 2019), MNLI (Williams et al., 2018), MRPC (Dolan and Brockett, 2005), QNLI (Rajpurkar et al., 2016), QQP[2], RTE (Dagan et al., 2005), SST-2 (Socher et al., 2013), and STS-B (Cer et al., 2017).

Training We perform standard hard-prompt fine-tuning on the GLUE training sets (Wang et al., 2018) using BERT-base-uncased (Devlin et al., 2019). Each task has a fixed prompt template and task-specific verbalizer; label logits come from the language-model logits at the `[MASK]` position. A single canonical MLM head loaded from the same pre-trained BERT checkpoint is frozen and shared across all tasks, seeds, and evaluation objects; only the BERT backbone parameters are

[2]Quora dataset release page



(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{lfd-tgt}}$ (ViT) (c) NormAccuracy (ViT) (d) Merged Accuracy (ViT)

(e) $\xi_{\mathrm{mg-lfd}}$ (BERT) (f) $\xi_{\mathrm{lfd-tgt}}$ (BERT) (g) Normalized GLUE (BERT) (h) Mean GLUE score (BERT)

Figure 14: Effect of dropout rate p for MIN ($\hat{\boldsymbol{\tau}}^{(\min)}$). In the ViT panels, solid curves and unhatched bars use three-run means with one standard deviation; dashed curves and hatched bars show the other tested rates from one run.

optimized. Task checkpoints contain backbone parameters only, and all task-vector construction, merging, and parameter interpolation are restricted to backbone keys. The NTK experiments use the same implementation (Ortiz-Jimenez et al., 2024) as in the CV setting.

Optimization details The reported hard-prompt runs use AdamW optimizer with $\beta = (0.9, 0.999)$ and $\epsilon = 10^{-8}$. The learning rate is 5×10^{-5} for CoLA, MRPC, and STS-B, and 10^{-5} for SST-2, QQP, MNLI, QNLI, and RTE. The per-device training and evaluation batch sizes are both 16. We train for 10 epochs with weight decay 0.01, a linear schedule, and warmup ratio 0.06. The training seed varies across runs, while the data split and evaluation sampling remain fixed as specified in Appendix D.2. We evaluate and save once per epoch and keep only one best checkpoint. The best checkpoint is selected by the validation metric: Matthews correlation for CoLA, averaged accuracy/F1 scores for MRPC and QQP, and accuracy for the remaining tasks.

Hard prompts and verbalizers The prompt templates and verbalizers are as follows. CoLA uses `{sentence} This is [MASK].` with unacceptable ↦ incorrect/wrong and acceptable ↦ correct/right; SST-2 uses `{sentence} It was [MASK].` with negative ↦ terrible/bad and positive ↦ great/good; MRPC uses `{sentence1} [MASK], {sentence2}` with not-equivalent ↦ no/false and equivalent ↦ yes/true; STS-B uses `{sentence1} [MASK], {sentence2}` with labels 0–4 mapped to different/low, related/fair, similar/okay, close/good, and same/high; QQP uses `{question1} [MASK], {question2}` with not-duplicate ↦ no/false and duplicate ↦ yes/true; MNLI uses `{premise}? [MASK], {hypothesis}` with entailment ↦ yes/true, neutral ↦ maybe/neutral, and contradiction ↦ no/false; QNLI uses `{question}? [MASK], {sentence}` with entailment ↦ yes/true and not-entailment ↦ no/false; and RTE uses `{sentence1}? [MASK], {sentence2}` with entailment ↦ yes/true and not-entailment ↦ no/false.

Evaluation For each GLUE task, we construct task vectors from the difference between the hard-prompt-tuned encoder and the pre-trained encoder. Given a target task, we evaluate f_{target}, f_{others}, f_{merged}, and the feature-level interpolation f_{lfd} on the target validation set. Specifically,

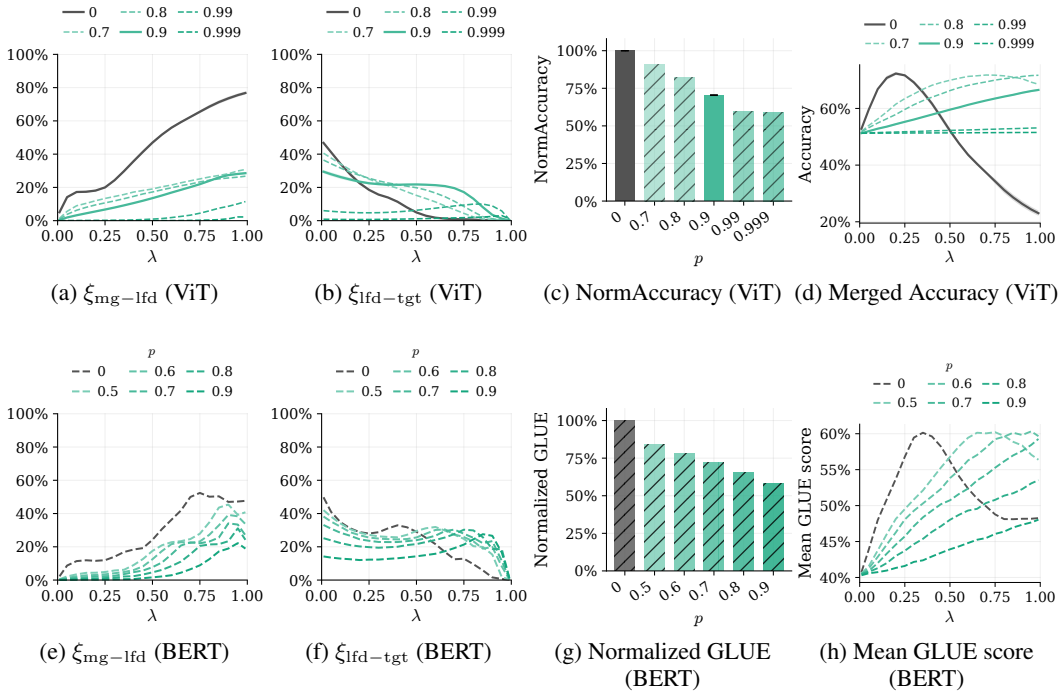


Figure 15: Effect of dropout rate p for RAND ($\hat{\boldsymbol{\tau}}^{(\mathrm{rand})}$). In the ViT panels, solid curves and unhatched bars use three-run means with one standard deviation; dashed curves and hatched bars show the other tested rates from one run.

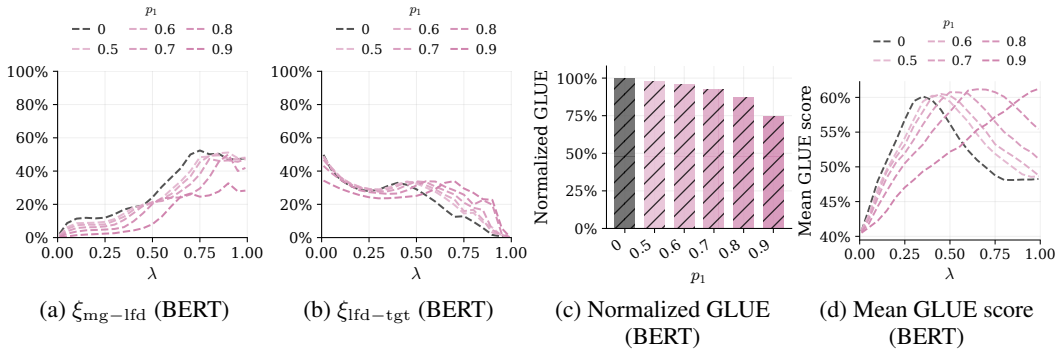
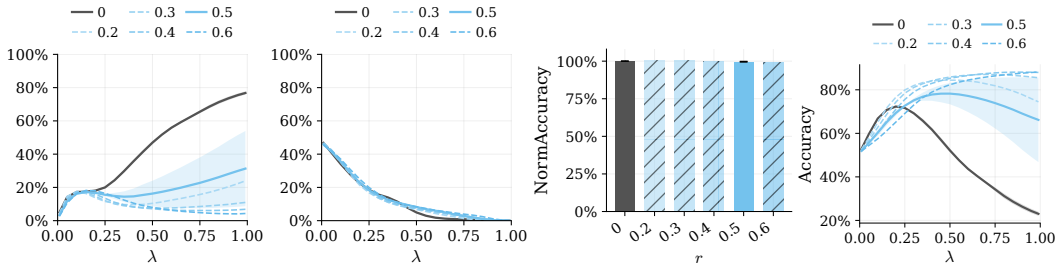


Figure 16: Effect of dropout rate p_1 for BRE ($\hat{\boldsymbol{\tau}}^{(\mathrm{bre})}$) with $p_2 = 0.99$.

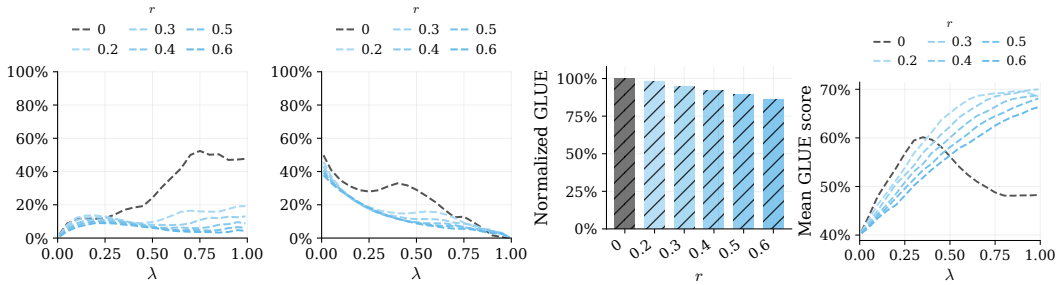
for nonlinear checkpoints, we extract the hidden representation at the `[MASK]` position from each evaluation object's BERT backbone and feed it into the same shared canonical frozen MLM head with the current target task's fixed verbalizer to obtain label logits. The verbalizers are fixed token-ID mappings and contain no trainable parameters. For NTK checkpoints, we directly evaluate the corresponding linearized logits. We then report the task-specific GLUE metric, including Matthews correlation for CoLA, accuracy for SST-2, STS-B, MNLI, QNLI, and RTE, and the average of accuracy and F1 for MRPC and QQP.

Hyperparameter

- λ**.** As in CV, we evaluate interpolation weights $\{0.01\} \cup \{0.05n\}_{n=1}^{19} \cup \{0.99\}$.
- **MIN ($\hat{\boldsymbol{\tau}}^{(\mathrm{min})}$) and RAND ($\hat{\boldsymbol{\tau}}^{(\mathrm{rand})}$).** Preliminary experiments suggest that BERT is more sensitive to parameter sparsification than ViT, so dropout rate p is set to



(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{lfd-tgt}}$ (ViT) (c) NormAccuracy (ViT) (d) Merged Accuracy (ViT)



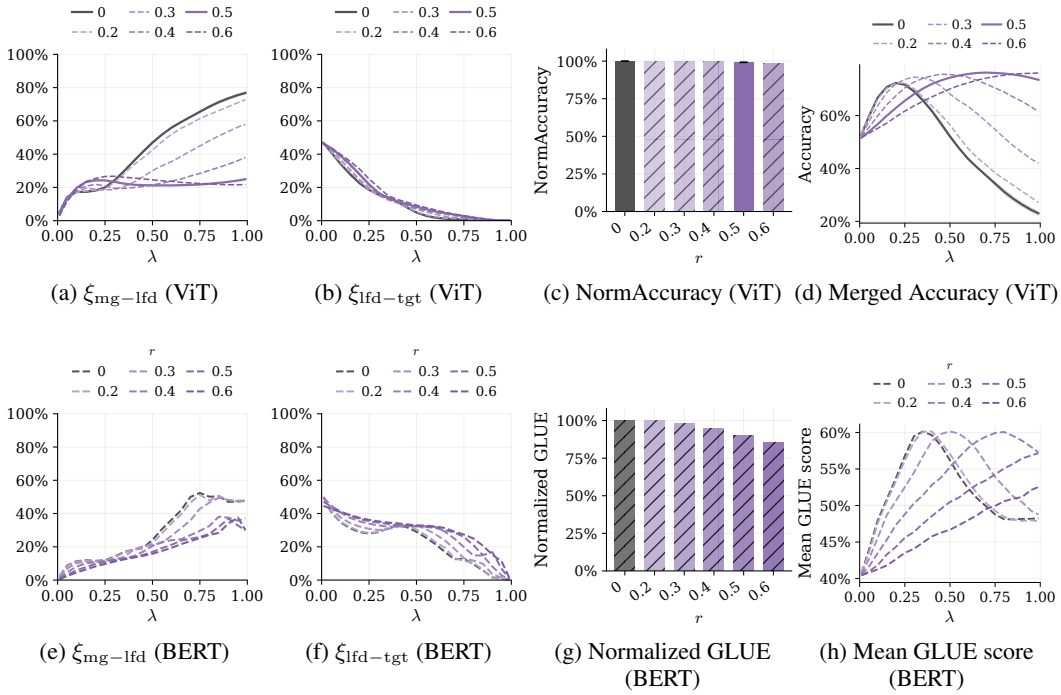
(e) $\xi_{\mathrm{mg-lfd}}$ (BERT) (f) $\xi_{\mathrm{lfd-tgt}}$ (BERT) (g) Normalized GLUE (BERT) (h) Mean GLUE score (BERT)

Figure 17: Effect of sparsification threshold r for CPS ($\hat{\boldsymbol{\tau}}^{(\mathrm{cps})}$). ViT defaults and zero-rate references use three-run means and standard deviations (solid curves and unhatched bars); other rates use one run (dashed curves and hatched bars).

$\{0.5, 0.6, 0.7, 0.8, 0.9\}$. Figures 14 (e)–(h) and 15 (e)–(h) report the results for MIN and RAND, respectively. For MIN, increasing p from 0.5 to 0.8 steadily reduces $\xi_{\mathrm{mg-lfd}}$, while $\xi_{\mathrm{lfd-tgt}}$ and NormAccuracy remain nearly unchanged. When p reaches 0.9, NormAccuracy begins to decline significantly, indicating that task-critical parameter values are being discarded. For RAND, increasing p reduces $\xi_{\mathrm{mg-lfd}}$, $\xi_{\mathrm{lfd-tgt}}$, and NormAccuracy together. Accordingly, as in the CV setting, the main-text metrics for MIN, RAND, and variants using them are reported with $p = 0.8$.

- **BRE ($\hat{\boldsymbol{\tau}}^{(\mathrm{bre})}$).** For BERT, we fix $p_2 = 0.99$ and search p_1 over $\{0.5, 0.6, 0.7, 0.8, 0.9\}$. Figure 16 shows that $p_1 = 0.8$ gives the best merged results. This discards 81% of parameter values in total, making the sparsity comparable to MIN ($\hat{\boldsymbol{\tau}}^{(\mathrm{min})}$) and RAND ($\hat{\boldsymbol{\tau}}^{(\mathrm{rand})}$).
- **CPS ($\hat{\boldsymbol{\tau}}^{(\mathrm{cps})}$).** Sparsification threshold r is set to $\{0.2, 0.3, 0.4, 0.5, 0.6\}$, which aligns with the search range recommended in the original paper (Wang et al., 2024). Figure 17 reports the results for CPS. The settings $r \in \{0.2, 0.3, 0.4\}$ exhibit consistent trends across evaluation metrics and yield nearly identical final accuracy for f_{merged}. We therefore select r mainly based on the behavior of TALL ($\hat{\boldsymbol{\tau}}^{(\mathrm{tall})}$).
- **TALL ($\hat{\boldsymbol{\tau}}^{(\mathrm{tall})}$).** Consistency threshold c is set to 2, matching the default value in the original paper (Wang et al., 2024). Since TALL is built on CPS ($\hat{\boldsymbol{\tau}}^{(\mathrm{cps})}$), Figure 18 reports the performance variation of TALL with $r = \{0.2, 0.3, 0.4, 0.5, 0.6\}$ and $c = 2$. The setting $r = 0.3$ gives the most balanced merged-accuracy curve, so we use it as the NLP default.

Statistical reporting The main ViT and BERT comparisons use training seeds 42, 43, and 44; split and evaluation-sampling seeds are fixed at 0. Each default-method comparison covers eight tasks and 21 interpolation weights per run. For each method and weight, we compute a task mean within each run and then report its mean and sample standard deviation across the three runs. NormAccuracy averages per-task accuracy ratios. BERT merged performance averages the task-specific GLUE scores, while its disagreement diagnostics and NormAccuracy use predicted classes.



(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{lfd-tgt}}$ (ViT) (c) NormAccuracy (ViT) (d) Merged Accuracy (ViT)

(e) $\xi_{\mathrm{mg-lfd}}$ (BERT) (f) $\xi_{\mathrm{lfd-tgt}}$ (BERT) (g) Normalized GLUE (BERT) (h) Mean GLUE score (BERT)

Figure 18: Effect of sparsification threshold r for TALL ($\hat{\boldsymbol{\tau}}^{(\mathrm{tall})}$). ViT defaults and zero-rate references use three-run means and standard deviations (solid curves and unhatched bars); other rates use one run (dashed curves and hatched bars).

The default-method comparisons, the standard-fine-tuning panel in Figure 2, the geometric curves in Figure 8, and the finite-condition checks use all three runs of each model family. The squared-norm ratio plots in Figures 4b and 12b, and the global-aware decomposition in Figure 13, also use three-run means and sample standard deviations. In the squared-norm ratio plots, $\mathbb{E}_{\mathrm{mask}}$ averages over the mask at fixed task vectors: RAND uses its exact expectation and deterministic masks use their realized values. Figure 6 selects the largest interaction increase divided by squared-norm reduction over tasks and weights within each run. At that same configuration, its three signed changes are divided by $\mathrm{E}_i - \mathrm{E}_i^{(\mathrm{cps})}$: the norm contribution is -1, the interaction contribution is the selected ratio, and their sum is the net bound change. We then report the mean and sample standard deviation across the three runs. The denominator is the decrease in weighted squared norm. Table 5 reports worst-case condition values over all covered runs and tasks. ViT hyperparameter plots use three-run estimates at the default rates and at the zero-rate reference (TA, or ELT for MEL), shown by solid curves and unhatched bars with uncertainty. Other rates and the BERT parameter sweeps use dedicated single-run data, shown by dashed curves and hatched bars without seed-uncertainty bands. The BERT sweeps label their retained-performance ratio as "Normalized GLUE": mean masked GLUE score divided by mean original GLUE score. It differs from the classification-based NormAccuracy in the main comparisons. The NTK illustration and cosine measurements also retain their dedicated single-run coverage.

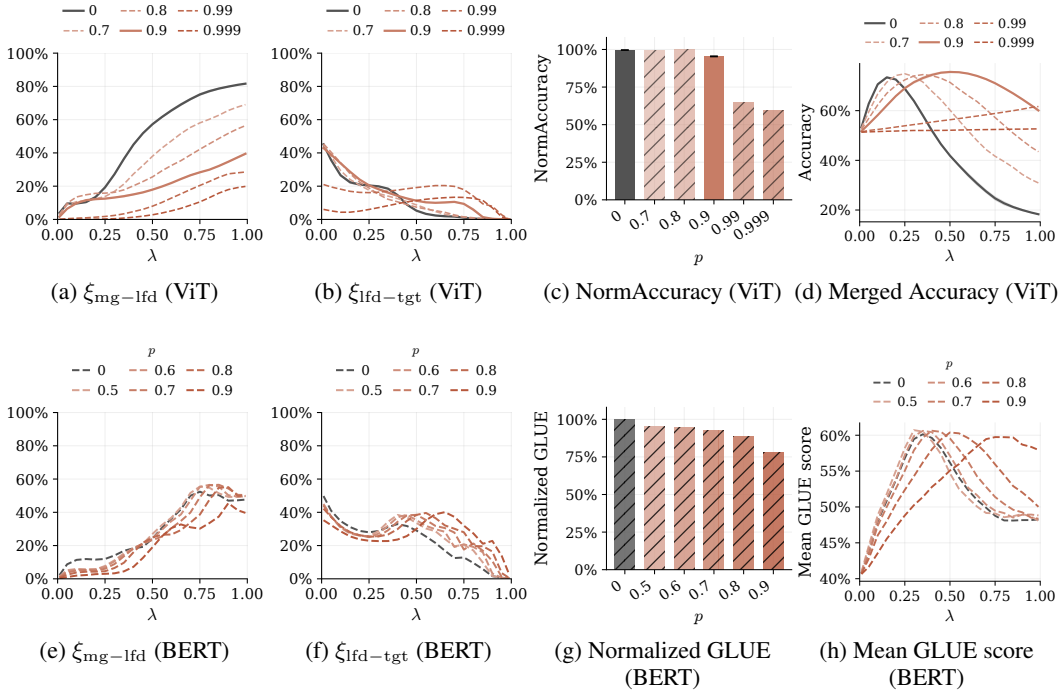
Compute The additional experiments were conducted on NVIDIA V100 GPUs. The finite-condition evaluation uses full-vector float64 Gram accumulation.

Code release The code and scripts needed to reproduce the main experiments will be released upon acceptance. Raw datasets remain available from their original benchmark sources.

E Additional Results on Mask Composition

MEL applies ELT to the MIN-processed vectors. Equation 29 gives $\mathbb{E}_i[\mathcal{B}_i^{(\mathrm{mel})}] \leq \mathbb{E}_i[\mathcal{B}_i^{(\mathrm{min})}]$ for $0 < \lambda < 1$ and $\lambda \leq 2/T$, where i is a uniformly chosen target task. The proof uses the loss of individual squared norms and gain in the squared vector sum, with no orthogonality assumption. Appendix C.3 gives the exact crossover at larger weights.

Figure 9 shows a smaller average $\xi_{\mathrm{mg-lfd}}$ for MEL than for either component at the displayed weights $\lambda < 0.3$ in both model families. Figure 19 reports the dropout-rate dependence, while Figure 20 compares both path diagnostics, retained capability, and merged performance. These comparisons show how the lower first-stage gap trades off against single-task degradation.



(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{lfd-tgt}}$ (ViT) (c) NormAccuracy (ViT) (d) Merged Accuracy (ViT)

(e) $\xi_{\mathrm{mg-lfd}}$ (BERT) (f) $\xi_{\mathrm{lfd-tgt}}$ (BERT) (g) Normalized GLUE (BERT) (h) Mean GLUE score (BERT)

Figure 19: Comparison across dropout rates p for MEL ($\hat{\boldsymbol{\tau}}^{(\mathrm{mel})}$). ViT defaults and zero-rate references use three-run means and standard deviations (solid curves and unhatched bars); other rates use one run (dashed curves and hatched bars).

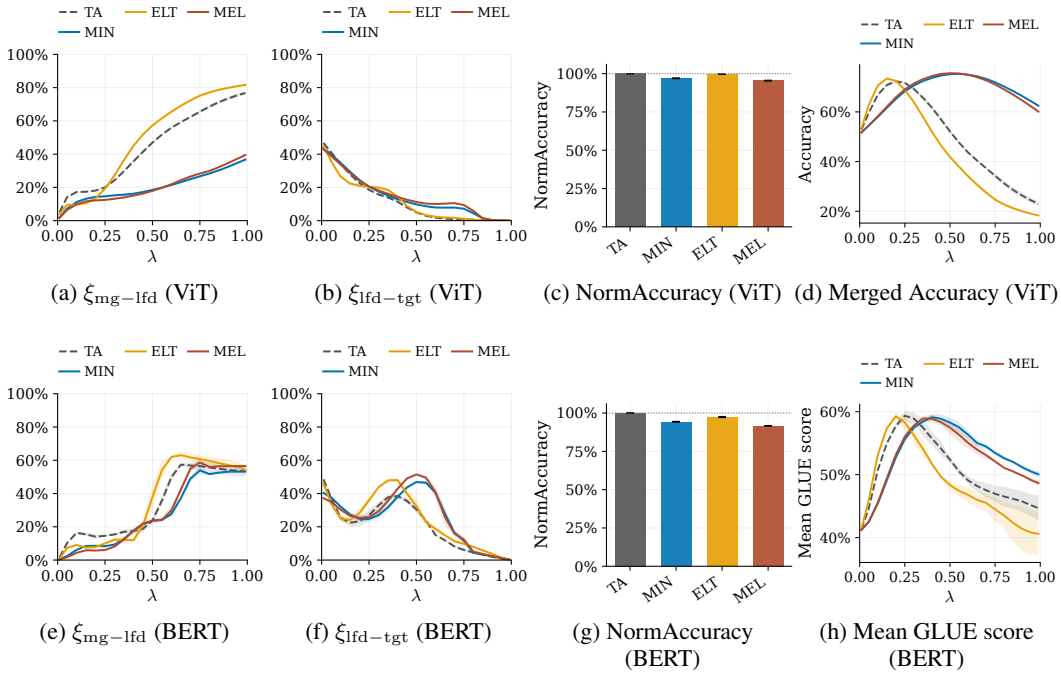
F Diagnostic Prediction Protocol and Ablations

We evaluate ViT-B/32 using the three runs specified in Appendix D.2. Each seed covers eight tasks, eight non-TA methods, and 21 interpolation weights. MIN and MEL use dropout rate $p = 0.9$, while CPS and TALL use threshold $r = 0.5$. We express accuracies as fractions, measure path discrepancies by predicted-label disagreement, and compute NormAccuracy as the ratio of masked to original single-task accuracy.

We retain configurations with $a_{\mathrm{ft}} - a_{\mathrm{TA}} > 0.02$ to avoid unstable normalization, leaving 4,000 observations across the three runs. Reference accuracies are matched within task, seed, and weight. All observations of the held-out task remain outside regression fitting, including feature standardization. For each training fold $\mathcal{T}$, the coefficients solve

$$(\hat{b}, \hat{\mathbf{w}}) = \arg\min_{b,\mathbf{w}} \sum_{j \in \mathcal{T}} (y_j - b - \mathbf{w}^\top \mathbf{z}_j)^2 + \|\mathbf{w}\|_2^2. \tag{30}$$

The intercept is unpenalized and the penalty is fixed. Table 6 pools out-of-fold scores. Pairwise accuracy excludes tied observed outcomes and counts predicted ties as half correct.



(a) $\xi_{\mathrm{mg-lfd}}$ (ViT) (b) $\xi_{\mathrm{lfd-tgt}}$ (ViT) (c) NormAccuracy (ViT) (d) Merged Accuracy (ViT)

(e) $\xi_{\mathrm{mg-lfd}}$ (BERT) (f) $\xi_{\mathrm{lfd-tgt}}$ (BERT) (g) NormAccuracy (BERT) (h) Mean GLUE score (BERT)

Figure 20: MIN, ELT, and MEL compared with TA: three-run means and one standard deviation.

Table 6: Three-seed diagnostic ablations on ViT-B/32. MAE and R^2 refer to normalized gain y.

Predictors	MAE ↓	R^2 ↑	Spearman ↑	Pairwise (%) ↑
Intercept only	0.502	-0.024	-0.157	50.0
Merged–LFD only	0.326	0.285	0.816	73.6
LFD–target only	0.533	-0.051	-0.144	48.0
NormAccuracy only	0.511	-0.013	-0.092	52.4
Both path diagnostics	0.329	0.315	0.812	78.0
All three diagnostics	0.321	0.288	0.856	81.7

Using both path diagnostics yields higher R^2 for gain magnitude than using all three features; the three-feature model achieves the highest Spearman and pairwise scores. These are predictive associations within the observed ViT suite; causal effects and evaluation-free selection remain open.

G TSV-Merge Case Study

We compare the official TSV-Merge matrix transformation with its rank-truncation-only ablation (lowrank) on ViT-B/32, seed 43, across eight tasks. Each task matrix keeps $\lfloor \min(m, n)/8 \rfloor$ singular components; TSV then orthogonalizes the concatenated left and right singular bases across tasks. Both arms use the same arithmetic averaging for other parameters and the original object-specific Euclidean 8-NN evaluation. A fixed set of 128 validation queries per task selects each method's interpolation weight; the reported test values use the complete test sets.

For the lowrank matrix components v_i, let $C = \sum_{i \neq j} \langle v_i, v_j \rangle$. Equation 8 gives the ideal case when TSV preserves their squared norms and removes cross-task inner products:

$$\overline{\mathcal{B}}^{(\mathrm{tsv})} - \overline{\mathcal{B}}^{(\mathrm{lowrank})} = -\frac{\lambda^2}{1-\lambda}\left(\lambda - \frac{2}{T}\right) C, \qquad 0 < \lambda < 1. \tag{31}$$

The measured lowrank matrix C is 17.1458 and the TSV value is approximately zero. For $T = 8$, the ideal sign changes at $\lambda = 0.25$. The measured mean bound term increases by 2.28% at $\lambda = 0.2$ and decreases by 17.08% at 0.9; these calculations retain the actual finite-precision Gram terms. The

official float32 implementation exceeds the strict ideal squared-norm tolerance 2×10^{-5} in 49 layers (maximum normalized deviation 9.50×10^{-5}), so the ideal equality is interpreted with this measured correction. Classification outcomes also depend on the model outputs and the 8-NN decision rule.

Table 7: Complete-test mean accuracy for lowrank and TSV on eight ViT-B/32 tasks, seed 43. Weights 0.2 and 0.9 give shared-coefficient comparisons; 1 reports accuracy only because the LFD path is defined for $\lambda < 1$.

λ	Lowrank (%)	TSV (%)	TSV − lowrank (pp)
0.2	71.84	63.82	−8.02
0.9	36.05	81.48	+45.43
1.0	33.03	81.18	+48.15

At $\lambda = 0.9$, mean $\xi_{\mathrm{mg-lfd}}$ changes from 0.6360 to 0.1517, mean $\xi_{\mathrm{lfd-tgt}}$ from 0.0135 to 0.0097, and NormAccuracy from 0.9935 to 0.9717. Validation selects $\lambda = 0.2$ for lowrank and 0.9 for TSV, giving a 9.64-point accuracy difference across their selected configurations. The corresponding target-task accuracy falls by 1.90 points, documenting a capability cost alongside the merge gain.

H Existing Assets and Licenses

We cite the original creators of all existing assets and use them under their public licenses or access terms. Raw benchmark datasets remain with their original distributors. Table 8 summarizes the assets used in the experiments.

Table 8: Existing assets and public licenses or access terms.

Asset	**License or public access terms**	**Use in this work**
CLIP ViT-B/32 (Radford et al., 2021)	MIT License	Pre-trained image encoder initialization.
BERT-base-uncased (Devlin et al., 2019)	Apache License 2.0	Pre-trained masked language model backbone.
Tangent task arithmetic code (Ortiz-Jimenez et al., 2024)	MIT License	Reference implementation for NTK fine-tuning experiments.
GLUE benchmark (Wang et al., 2018)	Component-specific public terms; QQP is subject to Quora non-commercial terms	NLP evaluation benchmark.
Cars (Krause et al., 2013)	ImageNet-style access terms reported by public metadata; no standard SPDX license identified	CV classification task.
DTD (Cimpoi et al., 2014)	Research-use public benchmark terms; no standard SPDX license identified	CV classification task.
EuroSAT (Helber et al., 2019)	MIT License	CV classification task.
GTSRB (Stallkamp et al., 2011)	CC0 Public Domain	CV classification task.
MNIST (LeCun, 1998)	Creative Commons Attribution-Share Alike 3.0	CV classification task.
RESISC45 (Cheng et al., 2017)	Public research benchmark; no standard SPDX license identified	CV classification task.
SUN397 (Xiao et al., 2016)	Research-use public benchmark terms	CV classification task.
SVHN (Netzer et al., 2011)	Public research benchmark; no standard SPDX license identified	CV classification task.