PRETRAINED AWARE MERGER LIFTING: BRINGING PRETRAINED RESPONSES INTO MODEL MERGING

Anonymous authors
Paper under double-blind review

ABSTRACT

Model merging combines models fine-tuned for different tasks without joint retraining. Task vector values alone do not describe how the shared pretrained model responds to parameter updates. A controlled experiment shows that identical perturbation values can produce substantially different output responses at different parameter locations, revealing information that update values alone do not capture. We introduce Pretrained Aware merger Lifting (PAL), which estimates pretrained sensitivity from a small set of generic unlabeled inputs and constructs an invertible parameter transform. PAL applies the original merging rules in transformed coordinates and maps the merged update back to the original parameter space. The sensitivity estimate and transform require no downstream task data and are computed once per checkpoint for reuse across task sets and mergers. PAL substantially improves recent state of the art mergers across CLIP vision tasks, RoBERTa on GLUE, and Llama-3.2-3B capability merging, with higher aggregate scores for every evaluated base merger.

1 INTRODUCTION

The widespread use of pretrained models has led to a growing collection of models fine-tuned for different tasks. Model merging (Ilharco et al., 2022; Matena & Raffel, 2022) offers an efficient way to combine multiple task-specific models into a single multi-task model without joint retraining. In this setting, Task Arithmetic (Ilharco et al., 2022) represents the updates learned during fine-tuning as task vectors relative to a shared pretrained checkpoint. TIES-Merging (Yadav et al., 2023), DARE (Yu et al., 2024), TSV-M (Gargiulo et al., 2025), and Iso-C/Iso-CTS (Marczak et al., 2025) improve task vector composition through sparsification, conflict resolution, and spectral decomposition.

A merged task vector specifies changes to the weights of the shared pretrained model. The effect of a task vector on model outputs depends on the pretrained model (Ortiz-Jimenez et al., 2023). Operations based only on task vector values do not explicitly account for this dependence. The change in output features, which we call the *pretrained output response*, can provide this missing information for composing updates (Figure 1(a)). To illustrate why this response information matters, we conduct a controlled experiment in a single pretrained CLIP ViT-B/32 (Radford et al., 2021) weight matrix. We preselect 30 disjoint sets of parameter locations with closely matched pretrained weight magnitude distributions, without using sensitivity estimates or output responses. For each of 32 perturbation seeds, we apply the same signed perturbation values to all 30 sets, changing only where the values are applied. The resulting pretrained output responses vary substantially across the 30 sets (Figure 1(b)). Across the 32 seeds, the median ratio between the largest and smallest responses over the 30 sets is $8.89\times$. These results show that perturbation values alone do not capture the effects of parameter updates on pretrained outputs.

We introduce Pretrained Aware merger Lifting (PAL) to incorporate pretrained output response information into existing model merging methods (Figure 1(c)). Evaluating this response directly requires forward passes with perturbed parameters for each candidate update. We instead approximate the response locally using a pretrained sensitivity metric. The metric is estimated from a small set of generic unlabeled inputs, such as 80 ImageNet-1k images (Deng et al., 2009) or 80 WikiText-2 texts (Merity et al., 2016), and used to construct an invertible parameter transform. Neither estimation nor transform construction requires downstream task data or access to the original pretraining

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.

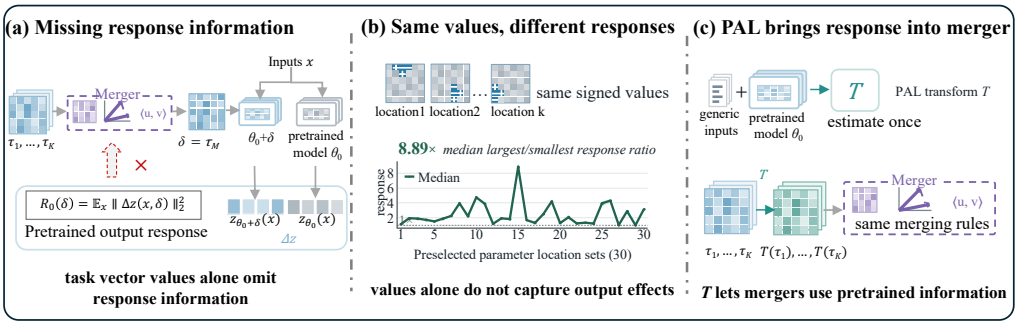

Figure 1: (a) Task vector values alone do not describe the resulting changes in pretrained outputs. We call these changes the pretrained output response. (b) The same signed perturbation values produce different pretrained output responses at different parameter locations. Across 32 perturbation seeds, the median ratio between the largest and smallest responses over 30 location sets is $8.89\times$, showing that perturbation values alone do not capture their effects on pretrained outputs. (c) PAL estimates pretrained sensitivity once from generic inputs and encodes it in a reusable transform, allowing existing mergers to use this information while keeping their original merging rules.

corpus. With the feature definition and estimation setup fixed, the sensitivity estimate and transform are computed once per checkpoint and reused across task sets and mergers. PAL transforms task vectors and the merger's relevant auxiliary quantities in parameter space, applies the original merging rules in the transformed coordinates, and maps the merged update back to the original parameter space. PAL delivers substantial improvements over recent state of the art mergers across CLIP vision tasks, RoBERTa on GLUE, and Llama-3.2-3B capability merging. Aggregate performance improves in all 45 vision comparisons, for all six GLUE mergers, and for all four Llama mergers, with mean gains of 1.26, 2.72, and 1.12 points, respectively.

Our main contributions are summarized below.

1. We identify pretrained output response as information missing from operations based only on task vector values. A controlled experiment shows that identical perturbation values can have markedly different effects on pretrained outputs.
2. We propose PAL, which constructs a reusable parameter transform from generic unlabeled inputs to incorporate pretrained sensitivity into existing mergers. It preserves the original merging rules and adds no inference overhead.
3. We demonstrate substantial improvements over recent state of the art mergers across vision, GLUE, and Llama capability merging. Aggregate scores improve in all 45 vision comparisons and for all six GLUE and four Llama mergers.

2 Related Work

2.1 Model Merging

We focus on merging models derived from the same pretrained checkpoint without additional joint training. Task Arithmetic (Ilharco et al., 2022) merges models by adding task vectors defined relative to a shared pretrained checkpoint. TIES-Merging (Yadav et al., 2023), DARE (Yu et al., 2024), TSV-M (Gargiulo et al., 2025), Iso-C/Iso-CTS (Marczak et al., 2025), and ESM (Li et al., 2026) improve task vector composition through conflict resolution, sparsification, projection, and spectral decomposition. DARE (Yu et al., 2024) modifies task vectors before merging and can be combined with different base mergers.

Some methods further incorporate information beyond the task vectors themselves, including Fisher information (Matena & Raffel, 2022), pretrained linearization (Ortiz-Jimenez et al., 2023), loss-landscape or curvature information (Tam et al., 2023; Lu et al., 2026), and task-specific activation shifts (Li et al., 2026). PAL measures sensitivity at the shared pretrained checkpoint using generic unlabeled inputs. A reusable parameter transform carries this pretrained output response information into existing mergers while preserving the original merging rules.

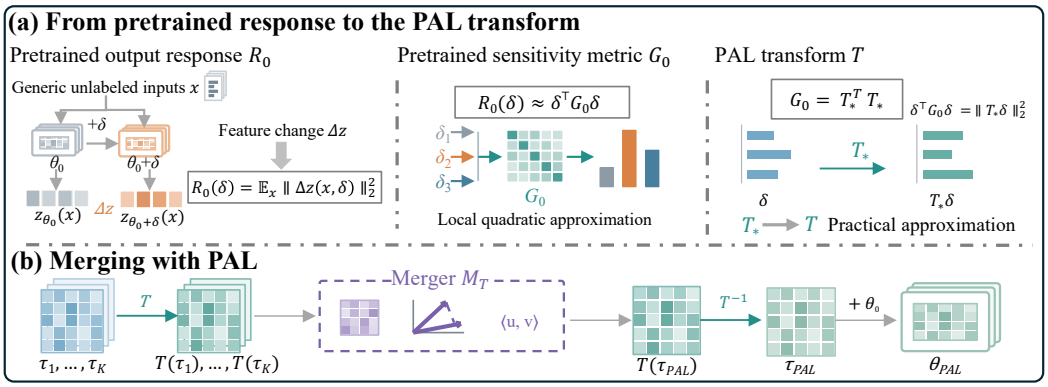

Figure 2: (a) The pretrained output response measures how a parameter update changes output features. A local approximation of this response gives the sensitivity metric G_0. We express this metric through an ideal transform $T_\star$ and construct a practical approximation T for merging. (b) PAL uses T to transform the task vectors and adapts the merger's auxiliary quantities where needed. The base merger applies its original rules in the transformed coordinates. PAL then uses T^{-1} to map the merged update back and adds it to the pretrained checkpoint.

3 Pretrained Aware Merger Lifting

3.1 Setup and Overview

Let f_θ be a model with parameters $\theta \in \mathbb{R}^P$ and a pretrained checkpoint θ_0. Fine-tuning on each of K tasks gives parameters θ_t and the corresponding task vectors (Ilharco et al., 2022)

$$\tau_t = \theta_t - \theta_0, \qquad t = 1, \ldots, K. \tag{1}$$

Task Arithmetic (Ilharco et al., 2022) combines these updates as $\theta_{\mathrm{TA}} = \theta_0 + \alpha \sum_{t=1}^{K} \tau_t$, where α controls the update scale. More generally, a base merger M produces

$$\tau_M = M(\tau_1, \ldots, \tau_K; \xi_M), \qquad \theta_M = \theta_0 + \tau_M. \tag{2}$$

Here ξ_M denotes auxiliary quantities required by the merger, such as activation statistics or bases in parameter space. We omit ξ_M when the context is clear.

PAL uses pretrained output response to inform how the base merger handles parameter updates (Figure 2). A local approximation of this response defines the pretrained sensitivity metric. We estimate the diagonal of this metric from generic unlabeled inputs and construct an invertible parameter transform. Task vectors and the merger's relevant auxiliary quantities in parameter space are represented in the transformed coordinates. We apply the original merging rules and map the merged update back to the original parameter space.

3.2 Pretrained Sensitivity Metric

Let $z_\theta(x) \in \mathbb{R}^d$ denote the normalized output features of f_θ on input x. A parameter perturbation $\delta \in \mathbb{R}^P$ induces the feature change $\Delta z(x, \delta) = z_{\theta_0+\delta}(x) - z_{\theta_0}(x)$. We quantify the pretrained output response over a generic input distribution $\mathcal{D}_g$ as

$$R_0(\delta) = \mathbb{E}_{x \sim \mathcal{D}_g} \left[\|\Delta z(x, \delta)\|_2^2 \right]. \tag{3}$$

Evaluating R_0 directly requires forward passes with perturbed parameters for each candidate update. A first order expansion around θ_0 gives

$$\Delta z(x, \delta) \approx J_0(x)\delta, \qquad J_0(x) = \left. \frac{\partial z_\theta(x)}{\partial \theta} \right|_{\theta=\theta_0}. \tag{4}$$

Substituting this expansion into the response function yields

$$R_0(\delta) \approx \delta^\top G_0 \delta, \qquad G_0 = \mathbb{E}_{x \sim \mathcal{D}_g}[J_0(x)^\top J_0(x)]. \tag{5}$$

We call G_0 the *pretrained sensitivity metric*. Its quadratic form provides a local approximation to the output response that can be reused across parameter updates. Appendix A.1 gives the approximation error and proof.

The positive semidefinite matrix G_0 has a symmetric square root, defining the ideal transform $T_\star = G_0^{1/2}$. It satisfies

$$G_0 = T_\star^\top T_\star, \qquad \delta^\top G_0 \delta = \|T_\star \delta\|_2^2. \tag{6}$$

This expresses the same quadratic measurement as a squared norm in transformed coordinates (Figure 2(a)). The ideal transform may be singular, so Section 3.3 constructs an invertible practical approximation T. PAL uses this measurement to inform merging. It does not minimize R_0 or require the merged model to preserve pretrained outputs.

3.3 Scalable Estimation and PAL Transform

The ideal transform $T_\star = G_0^{1/2}$ requires the full $P \times P$ sensitivity metric. We approximate G_0 by its diagonal and estimate each diagonal entry using F_i,

$$F_i \approx [G_0]_{ii} = \mathbb{E}_{x \sim \mathcal{D}_g} \left[\left[J_0(x)^\top J_0(x) \right]_{ii} \right]. \tag{7}$$

Each diagonal entry measures the sensitivity of output features to one parameter coordinate. The diagonal approximation retains these individual sensitivities and omits interactions between coordinates.

For each generic unlabeled input x_n, we draw Q probes $v_{nq} \in \mathbb{R}^d$, independently of the input, satisfying $\mathbb{E}[v_{nq} v_{nq}^\top] = I_d$. We compute

$$g_{nq} = J_0(x_n)^\top v_{nq} = \nabla_\theta \big(v_{nq}^\top z_\theta(x_n) \big) \big|_{\theta = \theta_0}. \tag{8}$$

One backward pass evaluates this product for all parameter coordinates while the pretrained parameters remain fixed. Averaging the squared gradient entries over N inputs and Q probes gives

$$F_i = \frac{1}{NQ} \sum_{n=1}^{N} \sum_{q=1}^{Q} g_{nq,i}^2. \tag{9}$$

For fixed inputs, F_i is unbiased for the corresponding empirical diagonal entry. Sampling inputs from $\mathcal{D}_g$ also gives an unbiased estimate of $[G_0]_{ii}$ under the conditions in Appendix A.1.

Applying the square root construction from Section 3.2 to the estimated diagonal metric gives the coordinate scales

$$s_i = \sqrt{F_i} \approx \sqrt{[G_0]_{ii}}. \tag{10}$$

These scales realize the estimated diagonal measurement,

$$\|s \odot \delta\|_2^2 = \sum_{i=1}^{P} F_i \delta_i^2. \tag{11}$$

For a matrix block W, we reshape the corresponding scales into a matrix S of the same shape. The coordinate scaling is then $S \odot W$.

We add a small damping term to each F_i, take the square root, and normalize the scales as specified in Appendix A.2. Let $\bar{S}$ denote the stabilized scale matrix. Since arbitrary elementwise scaling need not preserve matrix rank, we fit positive row and column factors $\bar{S} \approx r c^\top$ and define

$$T(W) = (r c^\top) \odot W = D_r W D_c, \tag{12}$$

where D_r and D_c are diagonal matrices with entries r and c, respectively. The positive factors make T invertible and preserve the rank of W. The rank one approximation applies to the scale matrix $\bar{S}$. The resulting transform rescales each entry of W without discarding any component of the task update. Figure 2(a) illustrates the connection between the pretrained output response and the resulting transform T.

The row and column factorization introduces a structured approximation to the stabilized diagonal measurement. Appendix A.2 derives the changes to the measurement and proves invertibility and rank preservation. Section 5.2 evaluates the effect on response ranking. With the feature definition and estimation setup fixed, we compute the sensitivity estimate and transform once per checkpoint and reuse them across task sets and mergers.

3.4 MERGING WITH PAL

PAL applies the transform from Section 3.3 to each task vector, giving $T(\tau_t)$. If the base merger also uses bases, projections, or inner products in parameter space, these quantities are adapted consistently to the transformed coordinates. For example, a parameter basis is transformed together with the task vector before projection, so both are compared under the same measurement. Let M_T denote the base merger evaluated in these coordinates, including the required adaptations. Mapping its output $T(\tau_{\mathrm{PAL}})$ back gives the merged update

$$\tau_{\mathrm{PAL}} = T^{-1}\big[M_T\big(T(\tau_1), \ldots, T(\tau_K)\big)\big]. \tag{13}$$

Adding this update to the pretrained checkpoint gives the merged model,

$$\theta_{\mathrm{PAL}} = \theta_0 + \tau_{\mathrm{PAL}}. \tag{14}$$

Figure 2(b) illustrates this procedure.

PAL changes the parameter measurement while retaining the base merger's scoring, weighting, selection, and composition rules. The transformed inputs can change the numerical scores and selected components produced by these rules. For a linear sum with coefficients held fixed, the transform and its inverse cancel, leaving the merged update unchanged. PAL therefore affects merging through operations that depend on the parameter measurement. Appendix A.3 proves this property and shows that PAL recovers the base merger when $T = I$.

The transformed measurement also affects operations that do not explicitly compute a norm. For a matrix block W, truncating the SVD of $T(W)$ to k components and mapping the result back gives a reconstruction $\widehat{W}$ that minimizes

$$\|T(W - \widehat{W})\|_{\mathrm{F}}^2$$

among matrices of rank at most k. This follows from the invertibility and rank preservation of $T(W) = D_r W D_c$. At a fixed rank, this objective assigns larger penalties to reconstruction errors in coordinates that receive larger scales. Appendix A.3 proves this equivalence and derives the corresponding basis projection. For TSV-M (Gargiulo et al., 2025), Iso-C, and Iso-CTS (Marczak et al., 2025), PAL applies the original spectral and composition rules to transformed task matrices. ESM (Li et al., 2026) also evaluates parameter bases and projections in PAL coordinates while retaining its procedure for activation statistics and semantic weighting. WUDI-Merging (Cheng et al., 2025) uses the transformed norms and inner products within its original objective. Appendix A.4 records the adaptations used by the base mergers.

4 EXPERIMENTS

We evaluate whether pretrained sensitivity improves existing merging methods across image classification, language understanding, and LLM capability merging. The vision experiments cover different backbone sizes and task sets. The language experiments examine the merging of task experts for language understanding and the combination of LLM capabilities across domains. Each comparison pairs a base merger with its PAL adaptation using the same pretrained checkpoint and task experts. We report aggregate performance to assess the consistency of the gains and individual task or domain scores to examine where PAL helps and where tradeoffs remain.

4.1 EXPERIMENTAL SETUP

Models and tasks. Vision experiments use CLIP (Radford et al., 2021) ViT-B/32, ViT-B/16, and ViT-L/14 (Dosovitskiy, 2020) on nested sets of 8, 14, and 20 image classification tasks. Language experiments use RoBERTa-base (Liu et al., 2019) on eight GLUE tasks (Wang et al., 2018) and Llama-3.2-3B (Grattafiori et al., 2024) on five MergeBench domains (He et al., 2026). Appendix A.4.1 lists the vision and GLUE tasks and the Llama evaluation domains.

Base mergers. Vision and GLUE experiments include TSV-M (Gargiulo et al., 2025), Iso-C and Iso-CTS (Marczak et al., 2025), DC-Merge (Zhang et al., 2026), and ESM (Li et al., 2026). GLUE additionally includes WUDI-Merging (Cheng et al., 2025). Llama experiments use TSV-M, Iso-C, Iso-CTS, and ESM. The base and PAL variants select their hyperparameters independently under the same validation procedure. Score aggregation and paired evaluation protocols are described in Appendices A.4.2 and A.4.3.

Table 1: Vision merging on three CLIP backbones and 8/14/20-task sets (%). Entries show accuracy followed by expert-normalized accuracy in parentheses; reference rows show accuracy only. Bold marks PAL.

Method	ViT-B/32			ViT-B/16			ViT-L/14		
	8	14	20	8	14	20	8	14	20
Zero-shot	48.26	57.21	56.10	55.34	61.28	59.73	64.70	68.20	65.23
Fine-tuned	92.83	90.88	91.37	94.64	92.76	93.17	95.81	94.29	94.73
Weight Avg.	66.34 (72.13)	64.34 (71.12)	61.04 (67.53)	72.22 (76.60)	69.46 (74.82)	65.31 (70.36)	79.56 (83.15)	76.73 (81.10)	71.60 (75.60)
TSV-M	85.86 (92.31)	80.06 (87.88)	77.07 (84.29)	89.01 (93.94)	84.58 (91.01)	80.57 (86.45)	92.98 (96.98)	89.17 (94.43)	87.72 (92.50)
+ PAL	**87.13 (93.68)**	**82.03 (90.01)**	**78.96 (86.23)**	**90.41 (95.41)**	**86.01 (92.56)**	**82.68 (88.68)**	**93.70 (97.74)**	**89.90 (95.23)**	**88.99 (93.86)**
Iso-C	86.30 (92.90)	80.30 (88.10)	75.50 (82.50)	90.60 (95.60)	84.80 (91.10)	79.60 (85.40)	94.20 (98.30)	89.30 (94.50)	87.60 (92.20)
+ PAL	**88.32 (95.08)**	**81.84 (89.79)**	**77.76 (84.93)**	**91.91 (97.02)**	**86.27 (92.70)**	**81.81 (87.66)**	**94.79 (98.92)**	**90.17 (95.42)**	**88.86 (93.60)**
Iso-CTS	86.20 (92.80)	81.70 (89.70)	78.10 (85.50)	91.10 (96.10)	86.40 (92.80)	82.40 (88.40)	94.70 (98.80)	91.00 (96.30)	90.10 (94.90)
+ PAL	**88.04 (94.75)**	**82.66 (90.70)**	**79.58 (86.86)**	**92.31 (97.49)**	**87.23 (93.72)**	**84.15 (90.15)**	**95.11 (99.25)**	**91.17 (96.49)**	**90.65 (95.52)**
DC-Merge	87.05 (93.55)	82.52 (90.62)	80.58 (88.18)	90.78 (95.83)	87.06 (93.70)	84.57 (90.76)	94.31 (98.38)	91.01 (96.43)	90.51 (95.43)
+ PAL	**87.77 (94.34)**	**83.86 (92.03)**	**82.26 (89.93)**	**91.54 (96.65)**	**87.81 (94.54)**	**86.14 (92.41)**	**94.44 (98.54)**	**91.70 (97.18)**	**91.39 (96.43)**
ESM	88.40 (95.30)	83.70 (92.00)	81.30 (88.90)	91.80 (97.00)	87.40 (94.10)	84.90 (91.10)	94.80 (98.90)	91.30 (96.80)	90.40 (95.30)
+ PAL	**89.88 (96.84)**	**85.13 (93.52)**	**83.17 (90.85)**	**92.40 (97.60)**	**88.50 (95.30)**	**85.86 (92.07)**	**95.13 (99.26)**	**91.62 (97.09)**	**91.45 (96.47)**

Table 2: Average normalized accuracy over the nine settings in Table 1 (%). Red values show PAL gains in percentage points.

Merger	TSV-M	Iso-C	Iso-CTS	DC-Merge	ESM
Base	91.09	91.18	92.81	93.65	94.38
+ PAL	**92.60** +1.51	**92.79** +1.61	**93.88** +1.07	**94.67** +1.02	**95.45** +1.07

4.2 Vision Model Merging

We evaluate PAL across different model sizes and task compositions. For each CLIP backbone, we estimate pretrained sensitivity from 80 unlabeled ImageNet-1k training images (Deng et al., 2009). The resulting transform is reused across all mergers and task sets for that backbone. We compare each base merger with its PAL adaptation using accuracy and normalized accuracy. Normalized accuracy is computed relative to the fine-tuned expert for each task and then averaged across tasks.

PAL improves normalized accuracy in all 45 vision comparisons. Table 1 shows a mean gain of 1.26 percentage points across the evaluated configurations. The gains averaged over the nine settings for each merger range from 1.02 points for DC-Merge to 1.61 points for Iso-C (Table 2). ESM, the strongest base merger by this average, improves from 94.38 to 95.45. Gains are generally larger on the two ViT-B backbones, while remaining positive across all settings on ViT-L/14. Averaged over backbones and mergers, the gains are 1.06, 1.12, and 1.59 points for the 8-, 14-, and 20-task sets, respectively, with the largest gain on the evaluated 20-task set.

4.3 Language Understanding

We evaluate how PAL affects individual GLUE task scores and their normalized average. For RoBERTa, we estimate pretrained sensitivity from 80 unlabeled WikiText-2 training texts (Merity et al., 2016) (Generic80). The resulting transform is reused across all mergers. Each task score is normalized by the corresponding single task expert score before averaging.

PAL improves the average normalized GLUE score for all six base mergers. Table 4 shows gains of 0.84–6.81 percentage points, averaging 2.72 points across mergers. ESM, the strongest base merger by this average, improves from 89.89 to 90.94, while Iso-CTS achieves the largest gain of 6.81 points. Iso-CTS improves on seven of the eight tasks, including gains of 12.5 points on STS-B and 8.2 points on QNLI. Table 3 also shows differences across individual tasks. For example, PAL increases the CoLA score of Iso-C from 28.9 to 43.3, while its STS-B score decreases from 59.7 to 38.2.

4.4 LLM Capability Merging

For Llama, we use the same 80 unlabeled WikiText-2 texts (Merity et al., 2016) (Generic80) as for RoBERTa. Sensitivity is estimated independently on the pretrained Llama model, producing

Table 3: RoBERTa-base merging on eight GLUE tasks (%). Entries show task scores followed by expert-normalized scores in parentheses; reference rows show task scores only. Bold marks the better paired score, and shading marks PAL.

Method	CoLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE
Zero-shot	0.0	49.1	15.8	15.0	41.1	34.2	52.4	53.4
Fine-tuned	56.5	94.7	88.0	86.4	89.7	87.0	91.7	66.4
Iso-C	28.9 (51.1)	89.9 (94.9)	**78.3 (89.0)**	**59.7 (69.1)**	**83.7 (93.3)**	77.6 (89.2)	84.3 (91.9)	61.0 (91.8)
+ PAL	**43.3 (76.6)**	**91.4 (96.5)**	78.2 (88.9)	38.2 (44.2)	82.8 (92.3)	**79.6 (91.5)**	**85.2 (92.9)**	**62.5 (94.0)**
TSV-M	22.2 (39.3)	92.8 (97.9)	78.7 (89.4)	50.4 (58.3)	**83.3 (92.8)**	73.1 (84.0)	84.4 (92.0)	**51.6 (77.7)**
+ PAL	**22.4 (39.6)**	92.8 (97.9)	**81.4 (92.5)**	**69.2 (80.1)**	82.3 (91.8)	**78.5 (90.2)**	**84.6 (92.3)**	46.6 (70.1)
Iso-CTS	40.0 (70.8)	85.0 (89.7)	78.3 (89.0)	56.8 (65.8)	**80.4 (89.7)**	68.0 (78.2)	78.4 (85.5)	58.5 (88.0)
+ PAL	**43.5 (77.0)**	**90.1 (95.2)**	**81.9 (93.1)**	**69.3 (80.3)**	80.3 (89.5)	**73.9 (85.0)**	**86.6 (94.4)**	**64.3 (96.7)**
DC-Merge	**29.5 (52.3)**	92.5 (97.7)	79.4 (90.3)	47.0 (54.5)	**84.9 (94.6)**	80.0 (92.0)	85.1 (92.8)	47.3 (71.2)
+ PAL	29.5 (52.2)	**93.0 (98.2)**	**79.5 (90.3)**	**53.2 (61.6)**	83.3 (92.8)	**80.4 (92.4)**	**86.6 (94.4)**	**50.5 (76.1)**
ESM	44.8 (79.3)	90.7 (95.8)	**83.1 (94.4)**	74.8 (86.6)	74.0 (82.4)	**75.7 (87.0)**	**85.3 (93.0)**	**66.8 (100.5)**
+ PAL	**47.0 (83.1)**	**92.1 (97.2)**	81.4 (92.5)	**78.4 (90.8)**	**84.1 (93.7)**	72.3 (83.1)	84.9 (92.6)	62.8 (94.6)
WUDI-Merging	46.2 (81.8)	**93.3 (98.6)**	69.6 (79.0)	52.9 (61.3)	**83.1 (92.7)**	**80.9 (93.0)**	**83.0 (90.5)**	57.0 (85.9)
+ PAL	**48.4 (85.7)**	92.7 (97.8)	**79.4 (90.3)**	**67.6 (78.3)**	82.1 (91.5)	73.4 (84.4)	82.4 (89.9)	**59.9 (90.2)**

Table 4: Average normalized score over the eight tasks in Table 3 (%). Red values show PAL gains in percentage points.

Merger	Iso-C	TSV-M	Iso-CTS	DC-Merge	ESM	WUDI-Merging
Base	83.79	78.94	82.09	80.66	89.89	85.34
+ PAL	**84.63** +0.84	**81.82** +2.88	**88.90** +6.81	**82.26** +1.61	**90.94** +1.05	**88.50** +3.16

Table 5: Llama-3.2-3B merging on five MergeBench domains (%). Avg. is the unweighted mean across domains. Bold marks the better paired score; red values show PAL gains in percentage points.

Method	Instruction	Math	Coding	Multilingual	Safety	Avg.
TSV-M	30.24	54.59	**41.49**	**47.88**	48.90	44.62
+ PAL	**33.84**	**54.81**	41.40	47.38	**51.83**	**45.85** +1.23
Iso-C	**11.83**	53.53	**38.30**	**47.72**	41.21	38.52
+ PAL	11.65	**56.03**	35.55	46.97	**47.57**	**39.55** +1.04
Iso-CTS	8.69	52.77	**37.89**	**47.28**	42.55	37.83
+ PAL	**9.43**	**54.81**	34.81	47.00	**46.91**	**38.59** +0.76
ESM	**16.88**	51.78	**40.06**	**47.69**	43.55	39.99
+ PAL	14.96	**55.88**	37.86	47.05	**51.56**	**41.46** +1.47

a separate metric and transform. The transform is reused across all Llama mergers. We evaluate individual domain scores and their unweighted mean to assess both capability changes and aggregate performance.

PAL improves the average score across the five MergeBench domains for all four base mergers. Table 5 shows gains of 0.76–1.47 percentage points, averaging 1.12 points across mergers. TSV-M with PAL achieves the highest average score of 45.85, while ESM gains the most at 1.47 points. For ESM, gains of 8.01 points in safety and 4.10 points in math offset declines in the other three domains. Math and safety scores improve for all four mergers, while coding and multilingual scores decline for all four. Instruction following improves for two mergers and declines for the other two.

5 Analysis

PAL uses pretrained sensitivity to guide parameter operations. We examine whether this helps preserve output features and whether the practical transform retains the response information needed for these operations. We also study how the estimation inputs and their number affect the results, and measure the cost of constructing the transform.

5.1 Effect of PAL on Low Rank Reconstruction

We test whether applying SVD through PAL better preserves task expert outputs at the same reconstruction rank. For each of three CLIP backbones, we reconstruct 18 task vector matrices indi-

Table 6: SVD and SVD + PAL perform low-rank reconstruction on CLIP ViT-B/32, ViT-B/16, and ViT-L/14, with retained ranks set to 20%, 40%, 60%, 80%, and 90% of the maximum rank. We report relative parameter reconstruction error and relative output-feature error.

	Parameter error		Feature error ($\times 10^{-6}$)	
Rank	**SVD**	**+ PAL**	**SVD**	**+ PAL**
20%	**0.115**	0.150	14.24	**6.93**
40%	**0.0485**	0.0641	3.18	**1.27**
60%	**0.0186**	0.0286	0.821	**0.342**
80%	**0.00488**	0.00876	0.209	**0.0719**
90%	**0.00176**	0.00330	0.0838	**0.0236**

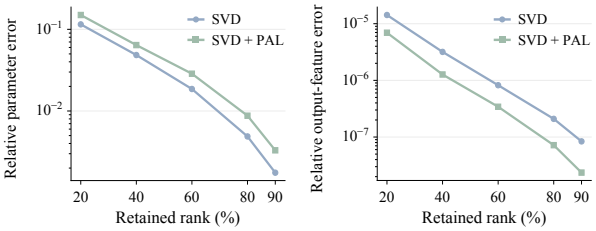

Table 7: Diagonal sensitivity and the PAL transform are evaluated for pretrained output response prediction on ViT-B/32 and RoBERTa-base. We report Spearman's ρ with the measured response, the change in ρ from diagonal sensitivity to the PAL transform, and the Spearman's ρ between the two; higher ρ indicates stronger ranking agreement.

Model	**Diagonal sensitivity**	**PAL transform**	**Change after fitting**	**PAL vs. diagonal**
	Spearman's ρ with measured response $\uparrow$		**Δ Spearman's ρ**	**Spearman's $\rho \uparrow$**
ViT-B/32	0.677	0.655	-0.022	**0.977**
RoBERTa-base	0.869	0.864	-0.005	**0.997**

vidually, keeping the remaining expert parameters unchanged. The retained rank ranges from 20% to 90% of the maximum rank. Feature evaluation uses 40 unlabeled ImageNet-1k images held out from sensitivity estimation. Parameter error is the squared Frobenius reconstruction error divided by the squared norm of the original matrix. Feature error measures the mean squared difference between the reconstructed and full experts' normalized output features, relative to the full expert's mean squared feature change from the pretrained model. Table 6 reports the median errors across the 54 reconstructions at each rank.

SVD + PAL achieves lower median feature error at every reported rank despite higher parameter error. At 80% rank, median feature error decreases from 0.209×10^{-6} to 0.0719×10^{-6}, while median parameter error increases from 0.00488 to 0.00876. The curves in Table 6 show the same tradeoff across the reported ranks. Although the feature errors are numerically small, their consistent ordering shows that lower parameter reconstruction error need not imply better preservation of expert outputs.

5.2 Fidelity of the Practical Transform

We examine how much response information is lost when the stabilized diagonal sensitivity is approximated by row and column factors. Constructing the full sensitivity matrix is impractical, so this comparison starts from the diagonal estimate used in PAL. On ViT-B/32 and RoBERTa-base, we compare predictions before and after fitting with measured output responses to 149 and 97 task vector updates, respectively. Updates have the same norm within each model. Response evaluation uses 40 inputs held out from sensitivity estimation, averaging the responses to each update and its negative.

The row and column approximation largely preserves the response rankings of the diagonal estimate. Predictions before and after fitting have Spearman correlations of 0.977 for ViT-B/32 and 0.997 for RoBERTa-base (Table 7). Their correlations with measured responses decrease by only 0.022 and 0.005, respectively. The practical transform therefore retains most of the ranking information provided by diagonal sensitivity. Because update norms are fixed within each model, this information cannot be obtained from update magnitude alone.

5.3 Data for Sensitivity Estimation

We examine whether sensitivity estimation requires particular input data or a carefully chosen sample count. Using the response evaluation in Section 5.2, we compare generic data, downstream data, and random noise with 80 inputs per source. Generic inputs are ImageNet-1k images or WikiText-2

Table 8: For PAL sensitivity estimation on ViT-B/32 and RoBERTa-base, generic data, downstream data, and random noise are used as inputs for pretrained output response prediction. We report Spearman's ρ with the measured response; higher ρ indicates stronger ranking agreement.

Input source	**Spearman's ρ with measured response ↑**	
	ViT-B/32	**RoBERTa-base**
Generic data	0.655	0.864
Downstream data	0.689	0.495
Random noise	0.674	0.820

Table 9: For Iso-C + PAL on the ViT-B/32 8-task and 20-task settings and RoBERTa-base GLUE, sensitivity is estimated using different numbers of generic inputs. We report expert-normalized scores on the validation sets; 100% corresponds to task expert performance.

Number of inputs	**Expert-normalized score (%) ↑**		
	ViT-B/32 (8 tasks)	**ViT-B/32 (20 tasks)**	**RoBERTa-base (GLUE)**
5	97.86	85.96	84.13
20	97.82	85.93	84.59
80	97.68	85.77	84.63
160	97.72	85.76	84.73

texts; noise consists of random pixels or token IDs. We vary the generic input count from 5 to 160 for Iso-C + PAL in two vision settings and GLUE, keeping merger settings fixed. Results are averaged over five input sets, except for the single shared estimate with 80 inputs.

Generic inputs provide useful response information, and merging performance changes little across the tested sample counts. All three input sources produce positive correlations with measured responses (Table 8). Downstream data perform slightly better on ViT-B/32 but worse on RoBERTa-base, showing no consistent advantage over generic data. Across 5–160 inputs, normalized scores vary by at most 0.20 points in both vision settings. GLUE improves from 84.13 to 84.73, with only 0.10 points gained by increasing the count from 80 to 160 (Table 9). These results support using a small set of generic inputs without requiring downstream data or precise tuning of the sample count.

5.4 Computational Cost

We measure the time required to construct and save the PAL transform from 80 generic inputs on NVIDIA RTX 4090 D GPUs. Timing covers sensitivity estimation and the subsequent steps that produce the transform, excluding model and input loading. We report median times over five runs for ViT-B/32 and RoBERTa-base and three runs for Llama-3.2-3B.

PAL constructs the transform once per pretrained checkpoint and adds no inference overhead. Construction takes 5.77 seconds for ViT-B/32, 3.70 seconds for RoBERTa-base, and approximately 3.66 minutes for Llama-3.2-3B. The transform is then reused across task sets and mergers sharing that checkpoint. The resulting merged model retains the original architecture, with no additional parameters or forward operations.

6 Conclusion

We introduced PAL to incorporate pretrained output response information into existing model merging methods. PAL estimates pretrained sensitivity from generic unlabeled inputs and uses a reusable parameter transform while preserving the original merging rules. Experiments on CLIP vision models, RoBERTa on GLUE, and Llama-3.2-3B show substantial improvements over recent state of the art mergers, with consistent gains in aggregate performance. These results support using the pretrained model's response to guide how task vectors are combined.

AI USE STATEMENT

We used generative AI tools to assist with drafting portions of the manuscript and improving the clarity, readability, and presentation of the writing. We also used generative AI tools to support research ideation and aspects of research execution, as well as to assist in developing mathematical arguments and proofs. All AI-assisted research ideas, technical content, and mathematical derivations were critically reviewed and independently verified by the authors. In particular, all mathematical claims and proofs were checked by the authors for correctness. The authors take full responsibility for the final content, claims, and results of this work.

ETHICS STATEMENT

All datasets used in this work are publicly available and are used in accordance with their respective licenses and terms of use. This work does not involve the collection of new human-subject data or personally identifiable information.

REPRODUCIBILITY STATEMENT

The main paper and appendix describe pretrained sensitivity estimation, the adaptations of the base mergers, the experimental settings, and the evaluation protocols. Code and instructions for reproducing the main experimental results will be made publicly available upon acceptance.

REFERENCES

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pp. 446–461. Springer, 2014.

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)*, pp. 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.

Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. *arXiv preprint arXiv:2503.08099*, 2025.

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*, pp. 3606–3613, 2014.

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

Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pp. 215–223. JMLR Workshop and Conference Proceedings, 2011.

Gregory Cohen, Saeed Afshar, Jonathan Tapson, and Andre Van Schaik. Emnist: Extending mnist to handwritten letters. In *2017 international joint conference on neural networks (IJCNN)*, pp. 2921–2926. IEEE, 2017.

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

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

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.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodola. Task singular vectors: Reducing task interference in model merging. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 18695–18705. IEEE, 2025.

Ian J Goodfellow, Dumitru Erhan, Pierre Luc Carrier, Aaron Courville, Mehdi Mirza, Ben Hamner, Will Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, et al. Challenges in representation learning: A report on three machine learning contests. In *International conference on neural information processing*, pp. 117–124. Springer, 2013.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Yifei He, Siqi Zeng, Yuzheng Hu, Rui Yang, Tong Zhang, and Han Zhao. Mergebench: A benchmark for merging domain-specialized llms. *Advances in Neural Information Processing Systems*, 38, 2026.

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, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. *arXiv preprint arXiv:2212.04089*, 2022.

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*, pp. 554–561, 2013.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

Longhua Li, Lei Qi, Qi Tian, and Xin Geng. Model merging in the essential subspace. *arXiv preprint arXiv:2602.20208*, 2026.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Juanwu Lu, Anand Bhaskar, Brian Axelrod, Ekaterina Tolstaya, and Tristan Emrich. Model merging on loss landscapes: A geometric perspective. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 7644–7653, June 2026.

Daniel Marczak, Simone Magistri, Sebastian Cygert, Bartłomiej Twardowski, Andrew D Bagdanov, and Joost Van De Weijer. No task left behind: Isotropic model merging with common and task-specific subspaces. *arXiv preprint arXiv:2502.04959*, 2025.

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

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

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, pp. 4. Granada, 2011.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pp. 722–729. IEEE, 2008.

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*, 36:66727–66754, 2023.

Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pp. 3498–3505. IEEE, 2012.

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*, pp. 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*, pp. 2383–2392, 2016.

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*, pp. 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*, pp. 1453–1460. IEEE, 2011.

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

Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant cnns for digital pathology. In *International Conference on Medical image computing and computer-assisted intervention*, pp. 210–218. Springer, 2018.

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

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

Adina Williams, Nikita Nangia, and Samuel R 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)*, pp. 1112–1122, 2018.

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

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pp. 3485–3492. IEEE, 2010.

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.

Linyan Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. Language models are super mario: Absorbing abilities from homologous models as a free lunch. In *Icml*, volume 2, pp. 21, 2024.

Han-Chen Zhang, Zi-Hao Zhou, Mao-Lin Luo, Shimin Di, Min-Ling Zhang, and Tong Wei. Dc-merge: Improving model merging with directional consistency. *arXiv preprint arXiv:2603.06242*, 2026.

Jeffrey Zhou, Tianjian Lu, Swaroop Mishra, Siddhartha Brahma, Sujoy Basu, Yi Luan, Denny Zhou, and Le Hou. Instruction-following evaluation for large language models. *arXiv preprint arXiv:2311.07911*, 2023.

A Appendix

A.1 Local Approximation and Sensitivity Estimation

Local response approximation. For the response in Eq. (3), assume that the normalized feature map $z_\theta(x)$ is twice continuously differentiable in $B(\theta_0, \rho)$ for almost every x. Let nonnegative functions $a(x)$ and $b(x)$ satisfy

$$\|J_0(x)\|_{\mathrm{op}} \le a(x), \qquad \sup_{\theta \in B(\theta_0,\rho)} \|D_\theta^2 z_\theta(x)\|_{\mathrm{op}} \le b(x), \tag{15}$$

with $\mathbb{E}[a(x)^2 + b(x)^2] < \infty$. The second norm is the operator norm of the vector-valued bilinear map. For $\|\delta\|_2 < \rho$, Taylor's formula gives

$$\Delta z(x, \delta) = J_0(x)\delta + r_x(\delta), \qquad \|r_x(\delta)\|_2 \le \tfrac{1}{2} b(x) \|\delta\|_2^2. \tag{16}$$

Subtracting $\|J_0(x)\delta\|_2^2$ from $\|\Delta z(x,\delta)\|_2^2$ leaves $2\langle J_0(x)\delta, r_x(\delta)\rangle + \|r_x(\delta)\|_2^2$. Bounding these two terms and taking expectations yields

$$\left|R_0(\delta) - \delta^\top G_0 \delta\right| \le \mathbb{E}[a(x)b(x)]\|\delta\|_2^3 + \tfrac{1}{4}\mathbb{E}[b(x)^2]\|\delta\|_2^4. \tag{17}$$

Both expectations are finite by the assumptions and the Cauchy–Schwarz inequality, so $R_0(\delta) = \delta^\top G_0 \delta + O(\|\delta\|_2^3)$ as $\delta \to 0$. This bound is local and need not remain accurate for an arbitrary fine-tuning update. The smoothness assumption includes feature normalization.

Unbiased sensitivity estimation. Use g_{nq} and F_i from Eqs. (8) and (9). Assume that the probes are independent of the inputs and of one another, with $\mathbb{E}[v_{nq} v_{nq}^\top] = I_d$. For a fixed input x_n,

$$\mathbb{E}_v[g_{nq} g_{nq}^\top \mid x_n] = J_0(x_n)^\top \mathbb{E}_v[v_{nq} v_{nq}^\top \mid x_n] J_0(x_n) = J_0(x_n)^\top J_0(x_n). \tag{18}$$

Taking the ith diagonal entry and averaging over the inputs gives

$$\mathbb{E}_v[F_i \mid x_{1:N}] = \left[\frac{1}{N}\sum_{n=1}^{N} J_0(x_n)^\top J_0(x_n)\right]_{ii}. \tag{19}$$

Thus F_i is unbiased for the empirical diagonal entry. If the inputs are independent samples from $\mathcal{D}_g$ and $\mathbb{E}_x\|J_0(x)\|_{\mathrm{F}}^2 < \infty$, taking expectation over the inputs also gives $\mathbb{E}_{x,v}[F_i] = [G_0]_{ii}$. Standard Gaussian probes and probes with independent Rademacher entries satisfy the second-moment condition; unit-norm probes require rescaling.

A.2 Properties of the Practical Transform

Changes to the measurement. Subtracting the estimated diagonal measurement in Eq. (11) from the full quadratic form gives

$$\delta^\top G_0 \delta - \sum_i F_i \delta_i^2 = \sum_i \big([G_0]_{ii} - F_i\big)\delta_i^2 + \sum_{i \ne j} [G_0]_{ij}\delta_i\delta_j. \tag{20}$$

The first term is the diagonal estimation error. The second contains omitted interactions between coordinates and need not be small for every update direction.

Our implementation stabilizes the coordinate scales from Section 3.3 using the damping and normalization defined below. Let $\bar{F} = P^{-1}\sum_i F_i$ and $\lambda > 0$. The stabilized scales are

$$\bar{s}_i = \frac{\sqrt{F_i + \lambda\bar{F}}}{\alpha_F}, \qquad \alpha_F = \left[\frac{1}{P}\sum_{j=1}^{P}(F_j + \lambda\bar{F})\right]^{1/2}. \tag{21}$$

Expanding their squared norm gives

$$\|\bar{s} \odot \delta\|_2^2 = \frac{\sum_i F_i\delta_i^2 + \lambda\bar{F}\|\delta\|_2^2}{\alpha_F^2}. \tag{22}$$

Damping adds an isotropic term, and normalization sets the overall scale. When $\bar{F} > 0$, all scales are positive and their root mean square is one. Equation (21) is undefined when $\bar{F} = 0$. Without damping, a zero F_i makes the diagonal scaling singular.

For the stabilized matrix $\bar{S}$ and positive factors r, c in Section 3.3, direct expansion of Eq. (12) gives

$$\|T(W)\|_{\mathrm{F}}^2 - \|\bar{S} \odot W\|_{\mathrm{F}}^2 = \sum_{i,j} \left[(r_i c_j)^2 - \bar{S}_{ij}^2\right] W_{ij}^2. \tag{23}$$

These identities describe how diagonal approximation, stabilization, and the fitted scales change the measurement from $\delta^\top G_0 \delta$ to $\delta^\top T^\top T \delta$.

Invertibility and rank preservation. Positive entries of r and c make D_r and D_c invertible. For any matrix W of the block's shape,

$$T^{-1}(W) = D_r^{-1} W D_c^{-1}, \qquad \operatorname{rank}(T(W)) = \operatorname{rank}(W). \tag{24}$$

The rank equality follows because left and right multiplication by invertible matrices leaves rank unchanged. Although the scale matrix $rc^\top$ has rank one, its action on vectorized parameters is a diagonal map with positive entries $r_i c_j$, so the parameter transform is invertible.

A.3 Properties of PAL Merging

Consistency. With $T = I$, all quantities retain their original coordinates and $M_T = M$, so Eq. (13) gives $\tau_{\mathrm{PAL}} = \tau_M$. For a linear merger with coefficients a_t held fixed in both coordinate systems, linearity of T gives

$$\tau_{\mathrm{PAL}} = T^{-1}\left[\sum_{t=1}^{K} a_t T(\tau_t)\right] = \sum_{t=1}^{K} a_t \tau_t. \tag{25}$$

Low rank reconstruction. By Appendix A.2, T is a bijection that preserves matrix rank. Thus $T(\widehat{W})$ ranges over all matrices of rank at most k as $\widehat{W}$ does. The truncated SVD $[T(W)]_k$ minimizes reconstruction error over this transformed feasible set by the Eckart–Young theorem. Mapping it back therefore gives

$$\begin{aligned} \widehat{W}_{\mathrm{PAL}} &= T^{-1}\big([T(W)]_k\big), \\ \widehat{W}_{\mathrm{PAL}} &\in \operatorname*{arg\,min}_{\operatorname{rank}(\widehat{W}) \le k} \|T(W - \widehat{W})\|_{\mathrm{F}}^2. \end{aligned} \tag{26}$$

This optimum is with respect to the metric induced by the practical transform. It does not imply optimality for the full G_0 or the nonlinear response R_0.

Consistent basis projection. For invertible T, write $\langle u, v\rangle_{\mathrm{PAL}} = \langle T(u), T(v)\rangle_2$ and $\|u\|_{\mathrm{PAL}} = \|T(u)\|_2$. For a parameter basis B with linearly independent columns, the closest vector Ba to u in this norm minimizes $\|T(u - Ba)\|_2^2$. Differentiating with respect to a gives $B^\top T^\top T B a = B^\top T^\top T u$. The coefficient matrix is positive definite because T is invertible and B has independent columns. Solving for a yields

$$\begin{aligned} \Pi_{B,T} u &= B(B^\top T^\top T B)^{-1} B^\top T^\top T u \\ &= T^{-1}\left[\Pi_{TB}(T(u))\right], \end{aligned} \tag{27}$$

where Π_{TB} is the Euclidean projector onto the span of TB. The second equality follows from the Euclidean projector formula for the transformed basis TB.

A.4 Merger Adaptations and Experimental Details

TSV-M (Gargiulo et al., 2025), Iso-C and Iso-CTS (Marczak et al., 2025) apply their original spectral and composition rules to transformed task tensors. DC-Merge (Zhang et al., 2026) also forms its decomposition quantities in PAL coordinates. ESM (Li et al., 2026) follows its original procedure for activation statistics, semantic weights, and rank selection, with task matrices, parameter bases, projections, and polar or Procrustes operations evaluated in PAL coordinates. The semantic weights act on the corresponding transformed factors. WUDI-Merging (Cheng et al., 2025) retains the structure of its objective and evaluates its parameter inner products and norms under T.

Output features. For CLIP, $z_\theta(x)$ is the ℓ_2-normalized final image embedding. For RoBERTa-base, it is the ℓ_2-normalized final-layer hidden state of the first token. For Llama-3.2-3B sensitivity construction, it is the ℓ_2-normalized final decoder hidden state at the last non-padding token.

A.4.1 Task Sets

The CLIP experiments use three nested image-classification task sets. Table 10 lists the eight-task core and the additions used to form the 14- and 20-task sets. All three visual backbones use these same task sets. The RoBERTa-base experiment merges experts (Wang et al., 2018) for CoLA (Warstadt et al., 2019), SST-2 (Socher et al., 2013), MRPC (Dolan & Brockett, 2005), STS-B (Cer et al., 2017), QQP, MNLI (Williams et al., 2018), QNLI (Rajpurkar et al., 2016), and RTE (Dagan et al., 2005). For Llama-3.2-3B, the five MergeBench domains (He et al., 2026) are instruction following, math, coding, multilingual, and safety.

Table 10: **Nested vision task sets.** Each larger setting contains all tasks from the preceding rows.

Setting	Tasks introduced
8 tasks	Cars (Krause et al., 2013), DTD (Cimpoi et al., 2014), EuroSAT (Helber et al., 2019), GTSRB (Stallkamp et al., 2011), MNIST (LeCun et al., 1998), RESISC45 (Cheng et al., 2017), SVHN (Netzer et al., 2011), SUN397 (Xiao et al., 2010)
14 tasks	STL10 (Coates et al., 2011), Oxford-IIIT Pet (Parkhi et al., 2012), Flowers102 (Nilsback & Zisserman, 2008), CIFAR100 (Krizhevsky et al., 2009), PCAM (Veeling et al., 2018), FER2013 (Goodfellow et al., 2013)
20 tasks	CIFAR10 (Krizhevsky et al., 2009), Food101 (Bossard et al., 2014), FashionMNIST (Xiao et al., 2017), Rendered SST-2 (Radford et al., 2021), EMNIST (Cohen et al., 2017), KMNIST (Clanuwat et al., 2018)

A.4.2 Reported Scores

In the vision table, absolute accuracy is averaged over the tasks in a setting. Normalization is performed for each task using its corresponding fine-tuned expert, before averaging over tasks. The two means are therefore not generally related by dividing mean merged accuracy by mean expert accuracy. Table 2 averages the nine normalized setting scores for each merger.

For GLUE, we use Matthews correlation for CoLA; accuracy for SST-2, MNLI-matched, QNLI, and RTE; the mean of accuracy and F1 for MRPC and QQP; and the mean of Pearson and Spearman correlations for STS-B. Each task score is normalized by its fixed task-expert reference score before averaging across tasks. The raw task scores are also shown because a positive normalized mean can conceal individual task regressions. The public final evaluation uses the same evaluator for each base/PAL pair and is distinct from a private hidden-test submission.

The language-model table uses a different aggregation. Its final column averages the five reported domain scores without expert normalization. The instruction-following score averages the loose instruction-level and loose prompt-level IFEval scores (Zhou et al., 2023). This reporting convention is stated explicitly because aggregate values depend on the choice of domain metrics and weights. All reported means and paired gains are computed before display rounding.

A.4.3 Shared Sensitivity and Paired Comparisons

For the main experiments in Section 4, the generic input set, sensitivity estimate, and practical PAL transform are fixed across the evaluated mergers sharing a pretrained checkpoint. CLIP uses 80 unlabeled ImageNet-1k training images (Deng et al., 2009). RoBERTa and Llama share exactly the same 80 raw WikiText-2 training texts (Merity et al., 2016) (Generic80), without labels. Sensitivity is estimated independently on each pretrained model, yielding separate metrics G_0 and transforms for the two language models.

Each paired comparison uses the same pretrained checkpoint, expert bank, task order, and final evaluator. The base and PAL versions retain the same merger-specific validation procedure. They may select different numerical hyperparameters. Vision configurations are selected on validation

data before evaluation on the test split. GLUE configurations are selected before the final public evaluation. For Llama, all mergers keep the tied token embedding/output head fixed at the pretrained checkpoint; base and PAL variants are otherwise selected independently under the same validation-only protocol.

Input-count analysis. For input counts of 5, 20, and 160, we average results over five deterministic input sets. The 80-input setting uses the single canonical Generic80 estimate shared with the main experiments. Merger configurations are fixed across input counts and are not re-tuned for each setting.