MERGE BEHAVIOR, NOT JUST WEIGHTS: DATA-FREE MODEL MERGING VIA SUBSPACE RECONSTRUCTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Model merging combines fine-tuned models without retraining, but success ultimately depends on whether the merged model behaves similarly to each expert on its own task. Existing methods largely operate on task-vector weights or their singular subspaces, but without task inputs, they lack effecttive constraints on the merged model's output behavior. We therefore ask how to preserve expert responses without using task inputs during merging? Starting from the observation that matching responses within an expert's output subspace reconstructs its projected output behavior, we construct data-free, layer-local response targets from task vectors and propose Subspace Behavior Reconstruction (SBR). SBR reconstructs each expert's self-response within its task-specific input and output spaces while suppressing unintended cross-task responses. To account for capabilities that may be shared across tasks, we further introduce SBR-R, which preserves cross-task responses on which the two experts agree. Extensive experiments demonstrate that SBR-R more closely preserves expert predictive distributions in CLIP diagnostics and consistently outperforms prior state-of-the-art data-free methods in average performance on CLIP, Qwen-14B LoRA, and InternVL2.5.

1 INTRODUCTION

Model merging combines task-specialized models into one, retaining their capabilities without separately running every expert. It reuses existing fine-tuned models, avoiding retraining and added inference cost. A common way to merge these models without task data is to combine their *task vectors*, the weight updates from pretraining (Ilharco et al., 2023). However, directly combining these task vectors can cause the merged model's behavior to deviate from that of each expert on its own task. To mitigate this deviation, existing methods adjust how task vectors are combined through trimming (Yadav et al., 2023), dropping (Yu et al., 2024), alignment (Ainsworth et al., 2023), or tangent-space analysis (Ortiz-Jimenez et al., 2023). The lack of task data makes it difficult for existing methods to directly optimize behavioral consistency between the merged model and each expert, raising the question: how can we approximate this consistency in a data-free setting?

Our starting point is that aligning the merged model's outputs with an expert's in the expert's output subspace can help preserve that expert's behavior. To make this concrete, we use each task vector's leading singular directions, which capture its dominant fine-tuning changes, to define important input and output spaces. We then define a data-free *self-response loss* that encourages the merged model to match each expert's responses within its input and output subspaces.

Yet self-response alone is not enough. It only checks whether the merged model matches expert i in its own output space. On the same input, the merged model may also activate the output space of another expert $j \neq i$. We measure these activations as a *cross-response interference surrogate*. During task-specific inference, excessive cross responses mix task-irrelevant components into the hidden representation and can severely impair predictions.

Therefore, to preserve expert behavior, we combine self-response reconstruction with cross-task response control in our proposed *Subspace Behavior Reconstruction* (SBR) method. Since experts may share useful behavior, we further extend SBR to SBR-R, which adaptively suppresses cross-task responses while retaining components supported by agreement between experts. To evaluate how well this reconstruction preserves expert behavior, we use real inputs after merging to measure both local responses and final predictions. The *core response error* measures the difference between

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.

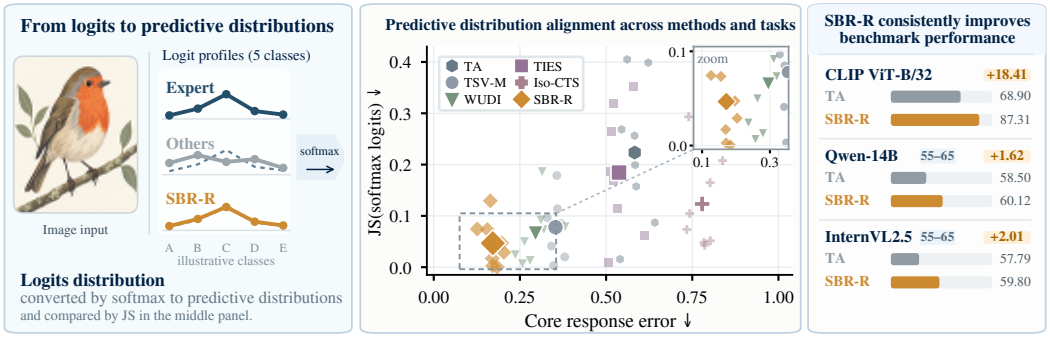


Figure 1: **Left:** a schematic showing that SBR-R's logit distributions more closely match the expert's during inference. **Middle:** each small point is one method on one CLIP task. Left means more expert-like internal responses; down means more expert-like class probabilities under the same classifier. Large points average eight tasks, SBR-R is nearest the bottom-left. The inset enlarges that region. **Right:** SBR-R achieve SOTA performance on CLIP, Qwen-14B, and InternVL2.5.

merged and expert responses in the expert's output subspace, using inputs projected into its input subspace and normalizing by its response magnitude. The Jensen–Shannon (JS) divergence measures the difference between their predictive distributions under the same classification head. Both metrics are lower when the merged model more closely matches the expert. In the middle panel of Figure 1, SBR-R achieves lower averages on both metrics than the compared baselines.

Our main contributions are:

- We show how matching responses in expert output subspaces can help preserve expert behavior, and use task vectors to approximate this alignment without task data.
- We propose SBR to reconstruct each expert's self-response while controlling cross-task responses, and extend it to SBR-R to retain responses supported by expert agreement.
- Extensive experiments demonstrate SBR-R consistently outperforms the SOTA data-free baselines in average performance across CLIP backbones, Qwen-14B LoRA, InternVL2.5, and BERT.

2 Related Work

Task vectors and weight-space conflict resolution. Ilharco et al. (2023) introduce task vectors, with weight averaging as the simplest aggregation (Wortsman et al., 2022), (Zimmer et al., 2024), and (Matena & Raffel, 2022). TIES resolves sign conflicts by trimming magnitudes (Yadav et al., 2023). DARE drops and rescales redundant entries (Yu et al., 2024). CAT removes conflict-prone components (Sun et al., 2025a). Other lines align neurons via permutation symmetries (Ainsworth et al., 2023), tie shared SVD modes (Stoica et al., 2024b) and (Stoica et al., 2024a), or localize task-specific updates (Jin et al., 2023), (He et al., 2024), and (Wang et al., 2024a). Further variants learn per-layer coefficients (Yang et al., 2024b), correct representations (Yang et al., 2024a), or constrain updates in a trust region (Sun et al., 2025b). These methods modify *how* task vectors are aggregated. The SBR family constructs merged task vectors by reconstructing expert responses.

Subspace and spectral methods. Ortiz-Jimenez et al. (2023) analyze task arithmetic in the tangent space. TSV-M orthogonalizes task-specific singular vectors (Gargiulo et al., 2025). STAR truncates and rescales spectra (Lee et al., 2025). Iso-CTS aligns task-specific and merged subspaces (Marczak et al., 2025). Subspace-Boosting counteracts rank collapse (Skorobogat et al., 2025). DC-Merge preserves directional geometry in a shared basis (Zhang et al., 2026). Singular Value Calibration (SVC) corrects shared-direction over-accumulation (Li et al., 2026b). APGD performs data-free

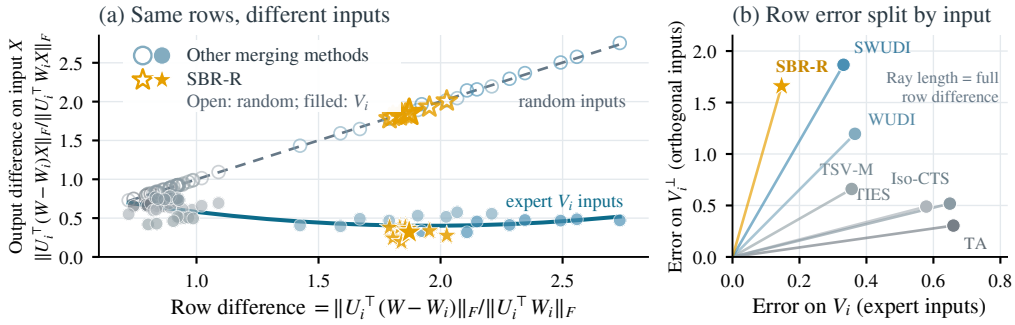


Figure 2: Output-space response matching on ViT-B/32 with rank-32 expert subspaces. **(a)** At the first MLP layer, row error is compared with output error within $\mathbf{U}_i$, using 1,024 Gaussian random inputs (open markers) or input directions $\mathbf{V}_i$ (filled markers). **(b)** Across eight experts and 24 MLP layers, RMS row error splits into components within $\mathbf{V}_i$ (horizontal) and $\mathbf{V}_i^\perp$ (vertical). SBR-R achieves the lowest average response error on the truly important expert-input directions.

projected optimization in a shared subspace (Wei et al., 2025), while WUDI uses task-vector row spaces as input proxies (Cheng et al., 2025) and SWUDI interprets the finite optimization trajectory as a spectral filter (Wei et al., 2026b). Concurrent work identifies essential directions from activation shifts (Li et al., 2026a) or optimizes outputs on calibration inputs (Evans et al., 2026). In contrast, both SBR variants derive input and output proxies solely from task vectors, explicitly optimize the paired responses $\mathbf{U}_j^\top \mathbf{W}\mathbf{V}_i$, and reconstruct the complementary input subspace. The technical supplement gives an equation-level comparison.

3 Subspace Behavior Analysis

Task vectors and subspace structure. A pretrained linear layer has weight $\mathbf{W}_0 \in \mathbb{R}^{m\times n}$. Fine-tuning on task k yields $\mathbf{W}_k^{\text{ft}}$. The *task vector* is $\Delta\mathbf{W}_k = \mathbf{W}_k^{\text{ft}} - \mathbf{W}_0$. We write $\mathbf{W}_k \equiv \Delta\mathbf{W}_k$ and fuse K task vectors into one $\mathbf{W}$ that is added back to $\mathbf{W}_0$. Each $\mathbf{W}_k$ has top-r SVD $\mathbf{W}_k \approx \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^\top$ with $\mathbf{U}_k \in \mathbb{R}^{m\times r}$, $\mathbf{V}_k \in \mathbb{R}^{n\times r}$. Thus, $\mathbf{V}_k$ and $\mathbf{U}_k$ span the dominant input and output directions of expert k's task update, which we use as proxies for its task-specific behavior. Define

$$\mathbf{Q} = \text{orth}([\mathbf{U}_1 \cdots \mathbf{U}_K]), \quad \mathbf{Z} = \text{orth}([\mathbf{V}_1 \cdots \mathbf{V}_K]), \tag{1}$$

the joint orthonormal output/input bases. Let $\mathbf{P}_\mathbf{Q} = \mathbf{Q}\mathbf{Q}^\top$, $\mathbf{P}_{\mathbf{Q}^\perp} = \mathbf{I}_m - \mathbf{P}_\mathbf{Q}$, $\mathbf{P}_\mathbf{Z} = \mathbf{Z}\mathbf{Z}^\top$, $\mathbf{P}_{\mathbf{Z}^\perp} = \mathbf{I}_n - \mathbf{P}_\mathbf{Z}$ be the corresponding orthogonal projectors.

Throughout, "response" denotes the layer-local response of a task-update matrix to the same input vector. We therefore take a layer-wise view, reconstructing expert behavior independently at each layer. The technical supplement further evaluates this layer-wise view using real activations and full-network behavior.

The same two-panel audit on Qwen-14B LoRA and InternVL2.5-1B is reported in Appendix E.2.

Why output-subspace behavior is necessary. The first object to preserve is the expert's output inside its output subspace. For an orthonormal basis $\mathbf{U}$ and projector $\mathbf{P}_\mathbf{U} = \mathbf{U}\mathbf{U}^\top$, matching $\mathbf{U}^\top\mathbf{W}$ is exactly the condition that the merged model and the expert produce the same output after projection onto that subspace.

Lemma 1 (Output-subspace behavior)**.** *Let $\mathbf{W}_m, \mathbf{W}_k \in \mathbb{R}^{m\times n}$ and let $\mathbf{U} \in \mathbb{R}^{m\times r}$ have orthonormal columns. Then*

$$\mathbf{U}^\top\mathbf{W}_m = \mathbf{U}^\top\mathbf{W}_k \iff \mathbf{P}_\mathbf{U}\mathbf{W}_m x = \mathbf{P}_\mathbf{U}\mathbf{W}_k x, \quad \forall x \in \mathbb{R}^n. \tag{2}$$

Equivalently, if $y_m = \mathbf{W}_m x$ and $y_k = \mathbf{W}_k x$, then

$$\mathbf{P}_\mathbf{U} y_m = \mathbf{P}_\mathbf{U} y_k, \qquad \forall x \in \mathbb{R}^n. \tag{3}$$

Thus $\mathbf{U}^\top \mathbf{W}_m = \mathbf{U}^\top \mathbf{W}_k$ *guarantees that the merged model and expert* k *have identical outputs inside the output subspace* $\mathrm{span}(\mathbf{U})$.

Lemma 1 makes output-side matching a natural starting point: $\mathbf{U}_i^\top \mathbf{W}$ controls what the merged layer produces inside expert i's output subspace. The question is whether the whole row operator $\mathbf{U}_i^\top \mathbf{W}$ is the right object to match.

Why row matching over-constrains output behavior. The relevant quantity for expert i is not its whole output row, but the response produced on the inputs to which expert i is sensitive. The right singular vectors $\mathbf{V}_i$ of the task vector span this expert-input subspace, so $\mathbf{U}_i^\top \mathbf{W}\mathbf{V}_i$ captures the response of a candidate layer $\mathbf{W}$ on expert i's inputs and inside expert i's output subspace.

Row matching ignores this input conditioning. Let $\mathbf{V}_i^\perp$ be any orthonormal complement of $\mathbf{V}_i$. Then

$$\|\mathbf{U}_i^\top(\mathbf{W} - \mathbf{W}_i)\|_F^2 = \|\mathbf{U}_i^\top(\mathbf{W} - \mathbf{W}_i)\mathbf{V}_i\|_F^2 + \|\mathbf{U}_i^\top(\mathbf{W} - \mathbf{W}_i)\mathbf{V}_i^\perp\|_F^2. \tag{4}$$

The first term is the response on expert i's input space. The second term measures off-input directions. A row objective can therefore reduce full-row error by sacrificing the response that matters for task i.

Proposition 1 (Row matching can miss expert-input response)**.** *Assume* $\mathrm{span}(\mathbf{V}_i)$ *has a nonempty orthogonal complement. There exist two merged weights* $\mathbf{W}_a, \mathbf{W}_b$ *such that*

$$\begin{aligned} &\|\mathbf{U}_i^\top(\mathbf{W}_a - \mathbf{W}_i)\mathbf{V}_i\|_F < \|\mathbf{U}_i^\top(\mathbf{W}_b - \mathbf{W}_i)\mathbf{V}_i\|_F, \\ \textit{but} \qquad &\|\mathbf{U}_i^\top(\mathbf{W}_a - \mathbf{W}_i)\|_F > \|\mathbf{U}_i^\top(\mathbf{W}_b - \mathbf{W}_i)\|_F. \end{aligned} \tag{5}$$

Thus a smaller row/output loss can correspond to a worse expert-input response.

This motivates measuring the self-response

$$R_{i\leftarrow i}(\mathbf{W}) = \mathbf{U}_i^\top \mathbf{W}\mathbf{V}_i, \tag{6}$$

with target $\mathbf{U}_i^\top \mathbf{W}_i \mathbf{V}_i$. In short, the expert's own response should be reconstructed on expert inputs, not as a full output row. Figure 2 illustrates both points. In panel (a), output differences within $\mathbf{U}_i$ on Gaussian inputs track the discrepancy in $\mathbf{U}_i^\top \mathbf{W}$, illustrating how output-space matching controls projected responses. For expert behavior reconstruction, we focus on responses to the expert-input directions $\mathbf{V}_i$, shown by the filled markers and lower fitted curve, rather than errors across the full input space. Panel (b) separates this discrepancy into components within $\mathbf{V}_i$ and $\mathbf{V}_i^\perp$, showing why a smaller full-row error need not mean better response matching on expert-input directions.

Why output matching misses cross-task response. Matching expert i's self-response requires

$$\mathbf{U}_i^\top \mathbf{W}\mathbf{V}_i = \mathbf{U}_i^\top \mathbf{W}_i \mathbf{V}_i. \tag{7}$$

This constrains the merged layer only on expert i's input space $\mathbf{V}_i$ and inside its output space $\mathbf{U}_i$. During inference, however, the response is evaluated in the joint output space $\mathbf{Q}$ shared by all experts. The self-response does not constrain the remaining output directions $\mathbf{U}_j^\top \mathbf{W}\mathbf{V}_i$ for $j \neq i$.

Proposition 2 (Self-response matching does not control joint-space response)**.** *Assume* $\mathrm{col}(\mathbf{Q})$ *strictly contains* $\mathrm{span}(\mathbf{U}_i)$ *and* $\mathrm{span}(\mathbf{V}_i)$ *is nonempty. Then there exists a merged weight* $\mathbf{W}_m$ *such that*

$$\mathbf{U}_i^\top \mathbf{W}_m \mathbf{V}_i = \mathbf{U}_i^\top \mathbf{W}_i \mathbf{V}_i, \qquad \textit{but} \qquad \mathbf{P}_\mathbf{Q} \mathbf{W}_m \mathbf{V}_i \neq \mathbf{P}_\mathbf{Q} \mathbf{W}_i \mathbf{V}_i. \tag{8}$$

Thus, the discrepancy can appear on task i*'s input directions while remaining invisible to the self-response constraint.*

These unconstrained components motivate the cross response

$$R_{j\leftarrow i}(\mathbf{W}) = \mathbf{U}_j^\top \mathbf{W}\mathbf{V}_i, \qquad j \neq i. \tag{9}$$

Its magnitude can indicate interference, but a nonzero cross response is not necessarily harmful. Indeed, $\mathbf{W}_i\mathbf{V}_i = \mathbf{U}_i\mathbf{\Sigma}_i$ implies $\mathbf{U}_j^\top \mathbf{W}_i \mathbf{V}_i = \mathbf{U}_j^\top \mathbf{U}_i \mathbf{\Sigma}_i$, which can be nonzero when the expert output subspaces $\mathbf{U}_i$ and $\mathbf{U}_j$ overlap. Weight-only evidence cannot determine whether such a component represents shared behavior, but forcing every cross target to zero can discard it. This motivates a conservative rule that preserves each expert's self-response while retaining a cross-task response only when the two experts respond consistently on the same input directions. The Methodology section constructs this rule directly from the expert weights.

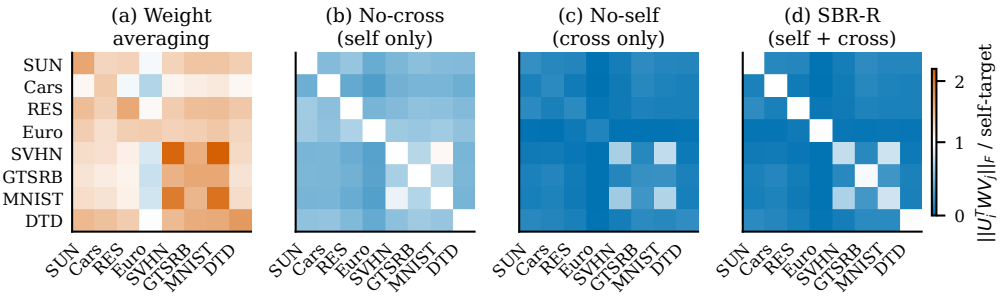


Figure 3: Component ablation on ViT-B/32's `visual.proj` layer. Rows index output expert i and columns index input expert j. Cells show $R_{i\leftarrow j} = \|\mathbf{U}_i^\top \mathbf{W} \mathbf{V}_j\|_F$, normalized by output expert i's self target. Diagonal values near one indicate preserved self-response, while off-diagonal values near zero indicate stronger cross-task suppression. The shared component-ablation color scale spans 0–2.2: white denotes 1, blue denotes below 1, and orange denotes above 1. **No-cross** preserves self-response but leaves cross response uncontrolled; **No-self** controls cross response without explicitly preserving self-response; **SBR-R** combines both terms to adaptively retain supported responses.

Why exact self matching can conflict. The responses above identify what the merged layer should control: the self-responses $\mathbf{U}_i^\top \mathbf{W} \mathbf{V}_i$ preserve expert behavior, while the cross responses $\mathbf{U}_j^\top \mathbf{W} \mathbf{V}_i$ provide an interference surrogate. However, exact equality on every self-response can be too strong. If two experts overlap in both input and output directions but demand different responses, no single merged layer can satisfy both experts exactly.

Proposition 3 (Exact self-behavior matching can conflict)**.** *If there exist experts $i \neq j$, an input $z \in \mathrm{span}(\mathbf{V}_i) \cap \mathrm{span}(\mathbf{V}_j)$, and an output direction $g \in \mathrm{span}(\mathbf{U}_i) \cap \mathrm{span}(\mathbf{U}_j)$ such that*

$$g^\top \mathbf{W}_i z \neq g^\top \mathbf{W}_j z, \tag{10}$$

then no single merged weight $\mathbf{W}$ can satisfy both exact self-behavior constraints $\mathbf{U}_i^\top \mathbf{W} \mathbf{V}_i = \mathbf{U}_i^\top \mathbf{W}_i \mathbf{V}_i$ and $\mathbf{U}_j^\top \mathbf{W} \mathbf{V}_j = \mathbf{U}_j^\top \mathbf{W}_j \mathbf{V}_j$. Consequently, the full set of exact self-behavior constraints can have no zero-loss solution.

The final question is whether these local responses control the actual expert-input response inside the joint output space $\mathbf{Q}$. The following theorem gives the desired link for chosen cross-response targets $\{\mathbf{T}_{ji}\}_{j\neq i}$, where $\mathbf{T}_{ji}$ specifies the desired response from expert i's input space to expert j's output space. Accordingly, the bound captures errors both when the merged model misses this target and when the target itself differs from the original expert response.

Theorem 1 (Self-response and cross-response losses bound joint error)**.** *Let $\mathbf{A} = [\mathbf{U}_1 \cdots \mathbf{U}_K]$ and $\lambda_{\mathbf{Q}} = \lambda_{\min}(\mathbf{Q}^\top \mathbf{A} \mathbf{A}^\top \mathbf{Q}) > 0$. Define the target discrepancy*

$$\mathcal{D}_{\mathbf{Q}}(\mathbf{W}) = \sum_i \|\mathbf{P}_{\mathbf{Q}}(\mathbf{W} - \mathbf{W}_i)\mathbf{V}_i\|_F^2, \tag{11}$$

and the self-response, cross-response, and target-mismatch terms

$$\mathcal{D}_s(\mathbf{W}) = \sum_i \|\mathbf{U}_i^\top (\mathbf{W} - \mathbf{W}_i)\mathbf{V}_i\|_F^2, \qquad \mathcal{D}_c^{\mathbf{T}}(\mathbf{W}) = \sum_i \sum_{j\neq i} \|\mathbf{U}_j^\top \mathbf{W} \mathbf{V}_i - \mathbf{T}_{ji}\|_F^2,$$
$$\mathcal{D}_o^{\mathbf{T}} = \sum_i \sum_{j\neq i} \|\mathbf{T}_{ji} - \mathbf{U}_j^\top \mathbf{W}_i \mathbf{V}_i\|_F^2. \tag{12}$$

Then for any merged weight $\mathbf{W}$,

$$\mathcal{D}_{\mathbf{Q}}(\mathbf{W}) \leq \frac{\mathcal{D}_s(\mathbf{W}) + 2\mathcal{D}_c^{\mathbf{T}}(\mathbf{W}) + 2\mathcal{D}_o^{\mathbf{T}}}{\lambda_{\mathbf{Q}}}. \tag{13}$$

In this bound, setting every cross-response target to zero corresponds to suppressing all cross responses, while retaining part of the original expert response can reduce the fixed mismatch term. Theorem 1 therefore supports reconstructing each expert's self-response and controlling deviations from the chosen cross targets $\mathbf{T}_{ji}$, but it does not identify which targets reflect shared behavior. We construct the targets from expert weights and validate them empirically. Appendix A.1 provides the proof and numerical audit.

Figure 3 shows the roles of the two terms. NO-CROSS reconstructs each expert's own behavior but does not control cross-task responses. NO-SELF controls cross-task responses but no longer explicitly preserves each expert's own behavior. Our method SBR-R combines both terms.

4 METHODOLOGY

The SBR family follows the preceding behavior analysis: retained dominant expert-input directions should be precisely handled in their joint subspace, while the complementary directions should be calibrated separately. For each linear layer, both variants construct a merged task vector from two orthogonal components: a *Specialists Subspace* component for $\mathrm{col}(\mathbf{Z})$ and a *Generalist Subspace* component for $\mathrm{col}(\mathbf{Z})^{\perp}$.

4.1 SPECIALISTS AND GENERALIST SUBSPACES

Let $\mathbf{Z} = \mathrm{orth}([\mathbf{V}_1 \cdots \mathbf{V}_K])$ collect all expert input directions and $\mathbf{Q} = \mathrm{orth}([\mathbf{U}_1 \cdots \mathbf{U}_K])$ collect all expert output directions. Every input decomposes as

$$x = \mathbf{P}_{\mathbf{Z}} x + \mathbf{P}_{\mathbf{Z}^{\perp}} x, \qquad \mathbf{P}_{\mathbf{Z}} = \mathbf{Z}\mathbf{Z}^{\top}, \quad \mathbf{P}_{\mathbf{Z}^{\perp}} = \mathbf{I} - \mathbf{P}_{\mathbf{Z}}. \tag{14}$$

The first term lies in the Specialists Subspace, where retained dominant expert responses and cross responses are jointly controlled. The second lies in the Generalist Subspace outside these dominant directions and may contain weaker expert-update components.

We parameterize the merged task vector as

$$\boxed{\mathbf{W}_{\mathrm{merge}} = \underbrace{\mathbf{P}_{\mathbf{Q}} \mathbf{W}_s \mathbf{P}_{\mathbf{Z}}}_{\text{Specialists Subspace}} + \underbrace{\mathbf{W}_g \mathbf{P}_{\mathbf{Z}^{\perp}}}_{\text{Generalist Subspace}}}. \tag{15}$$

Here $\mathbf{W}_s$ and $\mathbf{W}_g$ are two layer-shaped weights solved for the two orthogonal input regimes. The projector $\mathbf{P}_{\mathbf{Q}}$ routes every specialists response into the joint output space, so

$$\mathbf{P}_{\mathbf{Q}^{\perp}} \mathbf{W}_{\mathrm{merge}} \mathbf{Z} = 0. \tag{16}$$

Thus the Specialists component has no output outside $\mathrm{col}(\mathbf{Q})$ by construction, while its cross responses inside this space are regularized by the objective below.

4.2 SPECIALISTS SUBSPACE: TWO CROSS-RESPONSE TARGETS

Both variants reconstruct expert i's self-response $\mathbf{U}_i^{\top}\mathbf{W}_i\mathbf{V}_i$ and differ only in the target for each cross response $\mathbf{U}_j^{\top}\mathbf{W}_s\mathbf{V}_i$, where $j \neq i$. The original SBR suppresses every cross response toward zero:

$$\mathbf{T}_{ji}^{\mathrm{SBR}} = \mathbf{0}, \qquad j \neq i. \tag{17}$$

Zero can erase jointly supported behavior. To avoid this, SBR-R compares two responses in shared coordinates:

$$\mathbf{A}_{ji} = \mathbf{U}_j^{\top}\mathbf{W}_i\mathbf{V}_i, \qquad \mathbf{B}_{ji} = \mathbf{U}_j^{\top}\mathbf{W}_j\mathbf{V}_i. \tag{18}$$

$\mathbf{A}_{ji}$ is the source response, while $\mathbf{B}_{ji}$ measures support from the peer expert. SBR-R keeps the source direction and scales it by their positive agreement:

$$\mathbf{T}_{ji}^{\text{SBR-R}} = \widehat{\beta}_{ji}\mathbf{A}_{ji}, \qquad \widehat{\beta}_{ji} = \begin{cases} 0, & \mathbf{A}_{ji} = \mathbf{0} \text{ or } \mathbf{B}_{ji} = \mathbf{0}, \\ \dfrac{[\langle \mathbf{A}_{ji}, \mathbf{B}_{ji} \rangle_F]_+}{\|\mathbf{A}_{ji}\|_F \max\{\|\mathbf{A}_{ji}\|_F, \|\mathbf{B}_{ji}\|_F\}}, & \text{otherwise.} \end{cases} \tag{19}$$

The target retains identical responses, removes nonpositive agreement, and never reverses or amplifies the source response. These are weight-only safeguards rather than a semantic test for shared knowledge. The technical supplement derives the rule and states its conditional uniqueness guarantee in Appendix A.2.

Since $\mathbf{P}_{\mathbf{Z}}\mathbf{V}_i = \mathbf{V}_i$ and $\mathbf{U}_i^\top \mathbf{P}_{\mathbf{Q}} = \mathbf{U}_i^\top$, the specialists component in Eq. (15) has exactly the expert-input response measured in the preceding analysis. For $v \in \{\mathrm{SBR}, \mathrm{SBR\text{-}R}\}$, both methods use the response loss

$$\mathcal{L}_s^v(\mathbf{W}_s) = \frac{1}{K}\sum_{i=1}^{K} \left\|\mathbf{U}_i^\top \mathbf{W}_s \mathbf{V}_i - \mathbf{U}_i^\top \mathbf{W}_i \mathbf{V}_i\right\|_F^2 + \frac{1}{\sqrt{K}} \sum_{i=1}^{K}\sum_{j\neq i} \left\|\mathbf{U}_j^\top \mathbf{W}_s \mathbf{V}_i - \mathbf{T}_{ji}^v\right\|_F^2. \tag{20}$$

The first term is the self-response loss. The second suppresses cross responses toward the zero target of SBR or the adaptive target of SBR-R. The fixed coefficients follow a data-free minimax balance and are not selected using downstream accuracy. Appendix A.3 gives the derivation.

For either target, this is a convex least-squares problem in the joint $\mathbf{Q}$ and $\mathbf{Z}$ bases. Dense models use the reported fixed 50-step CG estimator, so the iteration budget is part of the method. The technical supplement provides the reduced solver and stopping audit in Appendix B.4, with architecture-specific details in Appendix C.1.

4.3 Generalist Subspace: Output-Direction Calibration

The Generalist Subspace handles directions in $\mathbf{Z}^\perp$. Since $\mathbf{V}_i^\top \mathbf{P}_{\mathbf{Z}^\perp} = 0$ for every retained expert basis, this component is invisible to the dominant directions used by the Specialists objective. It can nevertheless contain truncated expert-update tails, so both SBR and SBR-R calibrate their output directions coarsely rather than treating them as absent. They use the loss

$$\mathcal{L}_g(\mathbf{W}_g) = \sum_{k=1}^{K} \left\|\mathbf{U}_k^\top (\mathbf{W}_g \mathbf{P}_{\mathbf{Z}^\perp} - \mathbf{W}_k \mathbf{P}_{\mathbf{Z}^\perp})\right\|_F^2 + \frac{1}{\sqrt{K}} \left\|\mathbf{W}_g \mathbf{P}_{\mathbf{Z}^\perp} - \mathbf{S}\mathbf{P}_{\mathbf{Z}^\perp}\right\|_F^2, \tag{21}$$

where $\mathbf{S} = \sum_k \mathbf{W}_k$. The first term keeps the generalist response calibrated on each expert's output directions. The second term anchors it to the aggregated task vector. Let $\mathbf{P}_k = \mathbf{U}_k \mathbf{U}_k^\top$ and

$$\mathbf{H} = \frac{1}{\sqrt{K}}\mathbf{I} + \textstyle\sum_{k=1}^{K} \mathbf{P}_k. \tag{22}$$

Only the product $\mathbf{W}_g \mathbf{P}_{\mathbf{Z}^\perp}$ is needed, and its solution is

$$\mathbf{W}_g^\star \mathbf{P}_{\mathbf{Z}^\perp} = \mathbf{H}^{-1}\Big(\frac{1}{\sqrt{K}}\mathbf{S}\mathbf{P}_{\mathbf{Z}^\perp} + \textstyle\sum_{k=1}^{K} \mathbf{P}_k \mathbf{W}_k \mathbf{P}_{\mathbf{Z}^\perp}\Big). \tag{23}$$

The matrix $\mathbf{H}$ acts only on the output dimension, so Eq. (23) is solved directly without iteration and remains cheap relative to a full-layer solve.

Finally, both variants obtain the merged model by combining their solved Specialists and Generalist components through Eq. (15). The layer-wise pipeline and reduced-system solver details are given in Appendix B.3 and Appendix B.4, respectively.

5 Experiments

5.1 Setup

Models and tasks. We evaluate SBR and SBR-R across vision, language, and multimodal architectures. For vision, we merge $K = 8$ task vectors fine-tuned from three CLIP backbones (Radford et al., 2021), ViT-B/32, ViT-B/16, and ViT-L/14, on SUN397, Cars, RESISC45, EuroSAT, SVHN, GTSRB, MNIST, and DTD. For language, we merge four Qwen-14B LoRA experts trained for MMLU, TruthfulQA, BBQ, and CNN/DailyMail. For multimodal merging, we use InternVL2.5-1B and evaluate ten benchmarks spanning VQA, geometry, chart understanding, OCR, and referring-expression grounding. Due to space constraints, BERT results are reported in Appendix F.3. The technical supplement also provides dataset citations, checkpoint provenance, architecture-specific protocols, and fourteen-expert results.

Table 1: Capability merging results (%) on InternVL2.5-1B. The average is the mean over all ten benchmarks. All merge rows use $s = 0.1$. Best and second-best results are in **bold** and underlined.

	VQA		Geometry		Chart	OCR		Grounding			
Method	VizWiz	GQA	MathVista	MathVision	ChartQA	TextVQA	OCRVQA	RefCOCO	RefCOCO+	RefCOCOg	Avg
InternVL2.5-1B	28.99	54.70	45.40	14.47	69.72	74.21	41.76	74.49	67.89	71.32	54.29
Published and matched training-free baselines											
TA (Ilharco et al., 2023)	30.99	**57.04**	44.80	16.45	62.24	76.06	44.92	86.22	76.45	82.68	57.79
TIES (Yadav et al., 2023)	30.76	56.92	**48.30**	19.41	67.60	75.88	45.96	86.97	77.14	**84.08**	59.30
DARE (Yu et al., 2024)	31.01	57.02	45.20	15.79	62.08	**76.09**	44.69	86.33	76.57	82.82	57.76
TSV-M (Gargiulo et al., 2025)	30.85	56.85	46.60	17.76	68.80	75.91	45.87	87.23	76.92	83.85	59.06
Iso-CTS (Marczak et al., 2025)	31.13	56.70	47.70	20.39	68.72	75.63	46.97	87.00	76.96	83.75	59.50
WUDI (Cheng et al., 2025)	30.89	**57.04**	46.80	16.45	69.08	76.04	46.09	**87.32**	77.30	83.68	59.07
SWUDI (Wei et al., 2026b)	30.96	56.87	46.60	17.76	69.52	75.97	46.09	87.22	**77.31**	83.73	59.20
SWUDI-A (Wei et al., 2026b)	**31.17**	56.96	46.80	16.12	70.20	75.97	46.03	**87.32**	77.30	83.78	59.16
Our methods											
SBR	30.69	56.86	47.30	20.07	70.24	75.42	**47.88**	86.95	76.90	83.79	59.61
SBR-R	30.87	57.00	47.00	**21.38**	**70.52**	75.65	47.59	87.18	77.00	83.80	**59.80**

Table 2: Qwen-14B LoRA merging results. Experts are task-specific models. Small percentages give score/expert score (Avg.: ratio of means). Bold/underline: best/second-best merged results.

	Exact match ↑			*ROUGE-2* ↑	*Mean* ↑
Method	MMLU	TQA	BBQ	CNN/DM	**Avg**
Individual experts	67.95	66.42	94.74	20.48	62.40
TA	67.95	65.41	83.71	16.91	58.50
TIES	**67.96**	63.91	81.45	15.46	57.20
DARE	67.70	**65.66**	82.96	17.15	58.37
TSV-M	65.94	61.40	**90.23**	18.67	59.06
Iso-CTS	67.40	64.66	86.47	20.11	59.66
WUDI	65.74	61.65	89.22	18.28	58.72
SWUDI	66.51	63.66	89.72	19.22	59.78
SWUDI-A	66.92	63.66	87.97	18.53	59.27
SBR	67.55 (99.4%)	63.41 (95.5%)	83.21 (87.8%)	20.30 (99.1%)	58.62 (93.9%)
SBR-R	66.40 (97.7%)	63.41 (95.5%)	88.72 (93.6%)	**21.97** (107.3%)	**60.12** (96.3%)

Configurations. Rank choices, solver settings, and joint-basis construction are provided in the technical supplement. We merge one public checkpoint per expert, so expert-training seed variance is unavailable. We therefore test robustness across model backbones and experimental settings. Details are in the supplement.

5.2 Multimodal Large Language Model Merging

We evaluate multimodal merging on InternVL2.5-1B across ten benchmarks covering VQA, geometry, chart understanding, OCR, and grounding, following the MLLMerging protocol (Wei et al., 2026a). As shown in Table 1, SBR and SBR-R achieve average scores of 59.61 and 59.80, respectively. SBR-R exceeds the strongest listed baseline, Iso-CTS, by 0.30 points, supporting the applicability of response reconstruction to diverse multimodal tasks. Detailed merge and evaluation settings are provided in Appendix C.3.

5.3 Qwen-14B LoRA Merging

We next test whether the same subspace response principle applies to parameter-efficient LLM adaptation. We merge four LoRA experts (Hu et al., 2022) derived from Qwen-14B (Bai et al., 2023) and trained for MMLU, TruthfulQA, BBQ, and CNN/DailyMail, and evaluate the merged adapter with the local HELM/Twin protocol. Table 2 compares both methods with available training-free baselines. Our SBR-R reaches 60.12, exceeding the strongest external baseline, SWUDI, by 0.35

points. This result shows that response reconstruction transfers to large LoRA-adapted language models and is not tied to dense model weights.

5.4 CLIP Model Merging

We finally evaluate both methods on three CLIP backbones. Table 3 compares our SBR and SBR-R with SOTA training-free merging baselines, and both outperform every listed baseline across all three backbones. Against SWUDI, the strongest listed external baseline by average accuracy on each backbone, SBR-R gains 1.31, 1.24, and 0.96 percentage points and wins on 7/8, 8/8, and 8/8 individual tasks, respectively. For context, SWUDI's gains over WUDI on these backbones are 0.91, 0.81, and 0.21 points. These comparisons use fixed expert checkpoints and do not establish statistical significance across independent training runs.

Table 3: Average accuracy (%) on 8 CLIP tasks. Best and second-best results are in **bold** and underlined.

Method	ViT-B/32	ViT-B/16	ViT-L/14
TA	68.9	73.7	84.3
TIES	72.6	76.6	85.6
DARE	65.8	71.5	79.4
TSV-M	84.0	87.3	91.5
Iso-CTS	81.4	86.9	90.9
WUDI	85.1	88.8	92.5
SWUDI	86.0	89.6	92.8
SWUDI-A	85.5	89.5	92.5
SBR	<u>86.5</u>	<u>90.5</u>	<u>93.7</u>
SBR-R	**87.3**	**90.8**	**93.7**

Merge overhead. Table 4 compares one-time merge costs on ViT-B/32 after task vectors are loaded. SBR-R takes 29.9 seconds and 3.0 GB, compared with 6.7 seconds and 1.6 GB for SWUDI, while achieving higher accuracy (87.3% versus 86.0%). The detailed timing and memory protocol is provided in the technical supplement.

Component ablation. Table 5 evaluates each component at rank 32 with the same 50-step solver. Removing the self or cross term lowers accuracy by 18.56 and 24.59 points, respectively. Using the zero cross target of SBR lowers accuracy by 0.84 points, while replacing Generalist calibration with an additive tail lowers it by only 0.11 points.

Table 4: One-time ViT-B/32 merge overhead after task vectors are loaded.

Method	Avg ↑	Time (s) ↓	Mem. (GB) ↓
TA (WA)	68.9	$< \mathbf{0.1}$	$< \mathbf{0.1}$
Iso-CTS	81.4	49.0	9.8
WUDI	85.1	108.1	<u>1.6</u>
SWUDI	86.0	<u>6.7</u>	<u>1.6</u>
SBR	<u>86.5</u>	109.6	3.1
SBR-R	**87.3**	29.9	3.0

Table 5: SBR-R component ablation on ViT-B/32. Δ: accuracy change from SBR-R.

Configuration	Avg ↑	Δ
SBR-R	**87.33**	0.00
zero cross target (SBR)	86.49	−0.84
w/o self-response term	68.77	−18.56
w/o cross-response term	62.74	−24.59
additive Generalist tail	87.22	−0.11

6 Conclusion

We introduced SBR, a data-free model merging framework motivated by the observation that output-subspace alignment can help preserve expert behavior. Using task vectors, SBR approximates this alignment through self-response reconstruction and cross-response control. SBR-R further retains cross-responses supported by expert agreement. Experiments across vision, language, and multimodal models demonstrate the effectiveness of this approach, while real-input CLIP diagnostics provide evidence of improved consistency with expert predictions. These results support self/cross-response modeling as a practical way to approximate behavioral consistency without task data.

AI USE STATEMENT

We used AI tools for language editing, including translation, and improving manuscript clarity. The research ideas, methods, and experimental design were developed by the authors, who take full responsibility for the final content.

REFERENCES

Emanuele Aiello, Lili Yu, Yixin Nie, Armen Aghajanyan, and Barlas Oguz. Jointly training large autoregressive multimodal models, 2023. URL https://arxiv.org/abs/2309.15564.

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git re-basin: Merging models modulo permutation symmetries, 2023. URL https://arxiv.org/abs/2209.04836.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. Qwen technical report, 2023. URL https://arxiv.org/abs/2309.16609.

Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.

Rich Caruana. Multitask learning. *Machine Learning*, 28:41–75, 1997.

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. doi: 10.1109/JPROC.2017.2675998.

Runxi Cheng, Feng Xiong, Yongxian Wei, Wanyun Zhu, and Chun Yuan. Whoever started the interference should end it: Guiding data-free model merging via task vectors. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 10121–10143, 2025. URL https://proceedings.mlr.press/v267/cheng25h.html.

Rajas Chitale, Ankit Vaidya, Aditya Kane, and Archana Ghotkar. Task arithmetic with LoRA for continual learning, 2023. URL https://arxiv.org/abs/2311.02428.

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.

Jasper Dekoninck, Marc Fischer, Luca Beurer-Kellner, and Martin Vechev. Controlled text generation via language model arithmetic, 2024. URL https://arxiv.org/abs/2311.14479.

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*, 2019. URL https://arxiv.org/abs/1810.04805.

Bethan Evans, Benjamin Etheridge, Stephen Roberts, and Jared Tanner. Model merging by output-space projection, 2026. URL https://arxiv.org/abs/2605.29101.

Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *IEEE CVPR Workshop on Generative-Model Based Vision*, 2004.

Antonio Andrea Gargiulo, Donato Crisostomi, Maria Sofia Bucarelli, Simone Scardapane, Fabrizio Silvestri, and Emanuele Rodolà. Task singular vectors: Reducing task interference in model merging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 18695–18705, 2025. URL https://openaccess.thecvf.com/content/CVPR2025/html/Gargiulo_Task_Singular_Vectors_Reducing_Task_Interference_in_Model_Merging_CVPR_2025_paper.html.

Charles Goddard, Shamane Siriwardhana, Malikeh Ehghaghi, Luke Meyers, Vladimir Karpukhin, Brian Benedict, Mark McQuade, and Jacob Solawetz. Arcee's MergeKit: A toolkit for merging large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 477–485. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.emnlp-industry.36. URL https://aclanthology.org/2024.emnlp-industry.36/.

Danna Gurari, Qing Li, Abigale J. Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P. Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018.

Yifei He, Yuzheng Hu, Yong Lin, Tong Zhang, and Han Zhao. Localize-and-stitch: Efficient model merging via sparse task arithmetic, 2024.

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. doi: 10.1109/JSTARS.2019.2918242.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://arxiv.org/abs/2106.09685.

Chenyu Huang, Peng Ye, Tao Chen, Tong He, Xiangyu Yue, and Wanli Ouyang. EMR-Merging: Tuning-free high-performance model merging. In *Advances in Neural Information Processing Systems*, volume 37, pp. 122741–122769. Curran Associates, Inc., 2024. doi: 10.52202/079017-3900. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/dda5cac5272a9bcd4bc73d90bc725ef1-Abstract-Conference.html.

Drew A. Hudson and Christopher D. Manning. GQA: A new dataset for real-world visual reasoning and compositional question answering. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=6t0Kwf8-jrj.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 3(1):79–87, 1991. doi: 10.1162/neco.1991.3.1.79.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models, 2023.

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. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Yann LeCun, Corinna Cortes, and Chris Burges. MNIST handwritten digit database. https://yann.lecun.com/exdb/mnist/, 1998.

Yu-Ang Lee, Ching-Yun Ko, Tejaswini Pedapati, I-Hsin Chung, Mi-Yen Yeh, and Pin-Yu Chen. STAR: Spectral truncation and rescale for model merging. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pp. 496–505. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.naacl-short.42. URL https://aclanthology.org/2025.naacl-short.42/.

Longhua Li, Lei Qi, Xin Geng, and Qi Tian. Essential subspace merging for multi-task learning, 2026a. URL https://arxiv.org/abs/2606.19164.

Yayuan Li, Ze Peng, Jian Zhang, Jintao Guo, Yue Duan, and Yinghuan Shi. When shared knowledge hurts: Spectral over-accumulation in model merging. In *International Conference on Machine Learning*, 2026b. URL https://arxiv.org/abs/2602.05536.

Percy Liang, Rishi Bommasani, Tony Lee, Dimitris Tsipras, Dilara Soylu, Michihiro Yasunaga, Yian Zhang, Deepak Narayanan, Yuhuai Wu, Ananya Kumar, et al. Holistic evaluation of language models. *Transactions on Machine Learning Research*, 2023.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Annual Meeting of the Association for Computational Linguistics*, 2022.

Pan Lu, Hritik Bansal, Tony Xia, Jiacheng Liu, Chunyuan Li, Hannaneh Hajishirzi, Hao Cheng, Kai-Wei Chang, Michel Galley, and Jianfeng Gao. MathVista: Evaluating mathematical reasoning of foundation models in visual contexts. In *International Conference on Learning Representations*, 2024a.

Zhenyi Lu, Chenghao Fan, Wei Wei, Xiaoye Qu, Dangyang Chen, and Yu Cheng. Twin-Merging: Dynamic integration of modular expertise in model merging. In *Advances in Neural Information Processing Systems*, volume 37, pp. 78905–78935. Curran Associates, Inc., 2024b. doi: 10.52202/079017-2505. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/8fcd17eb91bae20d9826786d7d6be799-Abstract-Conference.html.

Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. Generation and comprehension of unambiguous object descriptions. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

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, 2025. URL https://arxiv.org/abs/2502.04959.

Ahmed Masry, Do Xuan Long, Jia Qing Tan, Shafiq Joty, and Enamul Hoque. ChartQA: A benchmark for question answering about charts with visual and logical reasoning. In *Findings of the Association for Computational Linguistics: ACL 2022*, 2022.

Michael S. Matena and Colin A. Raffel. Merging models with Fisher-Weighted Averaging. *Advances in Neural Information Processing Systems*, 35:17703–17716, 2022.

Anand Mishra, Shashank Shekhar, Ajeet Kumar Singh, and Anirban Chakraborty. OCR-VQA: Visual question answering by reading text in images. In *International Conference on Document Analysis and Recognition*, pp. 947–952, 2019.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Conference on Computational Natural Language Learning*, 2016.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y. Ng. Reading digits in natural images with unsupervised feature learning. Technical report, Stanford University, 2011.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing*, 2008.

Guillermo Ortiz-Jimenez, Alessandro Favero, and Pascal Frossard. Task arithmetic in the tangent space: Improved editing of pre-trained models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 66727–66754. Curran Associates, Inc., 2023. doi: 10.52202/075280-2913. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/d28077e5ff52034cd35b4aa15320caea-Abstract.html.

Bo Pang and Lillian Lee. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Annual Meeting of the Association for Computational Linguistics*, 2005.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *IEEE Conference on Computer Vision and Pattern Recognition*, 2012.

Alicia Parrish, Angelica Chen, Nikita Nangia, Vishakh Padmakumar, Jason Phang, Jana Thompson, Phu Mon Htut, and Samuel R. Bowman. BBQ: A hand-built bias benchmark for question answering. In *Findings of the Association for Computational Linguistics: ACL 2022*, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8748–8763. PMLR, 2021. URL https://proceedings.mlr.press/v139/radford21a.html.

Amanpreet Singh, Vivek Natarajan, Meet Shah, Yu Jiang, Xinlei Chen, Dhruv Batra, Devi Parikh, and Marcus Rohrbach. Towards VQA models that can read. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019.

Ronald Skorobogat, Karsten Roth, Mariana-Iuliana Georgescu, and Zeynep Akata. Subspace-boosted model merging, 2025. URL https://arxiv.org/abs/2506.16506.

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, 2011. doi: 10.1109/IJCNN.2011.6033395.

George Stoica, Daniel Bolya, Jakob Bjorner, Pratik Ramesh, Taylor Hearn, and Judy Hoffman. Zipit! merging models from different tasks without training, 2024a. URL https://arxiv.org/abs/2305.03053.

George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with SVD to tie the knots, 2024b.

Wenju Sun, Qingyong Li, Yangliao Geng, and Boyang Li. CAT merging: A training-free approach for resolving conflicts in model merging. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 57523–57543. PMLR, 2025a. URL https://proceedings.mlr.press/v267/sun25i.html.

Wenju Sun, Qingyong Li, Wen Wang, Yangliao Geng, and Boyang Li. Task arithmetic in trust region: A training-free model merging approach to navigate knowledge conflicts. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 5178–5187. ACM, 2025b. doi: 10.1145/3746027.3755789. URL https://doi.org/10.1145/3746027.3755789.

Fanqi Wan, Longguang Zhong, Ziyi Yang, Ruijun Chen, and Xiaojun Quan. FuseChat: Knowledge fusion of chat models, 2024. URL https://arxiv.org/abs/2408.07990.

Ke Wang, Nikolaos Dimitriadis, Guillermo Ortiz-Jimenez, François Fleuret, and Pascal Frossard. Localizing task information for improved model merging and compression, 2024a. URL https://arxiv.org/abs/2405.07813.

Ke Wang, Junting Pan, Weikang Shi, Zimu Lu, Mingjie Zhan, and Hongsheng Li. Measuring multimodal mathematical reasoning with MATH-Vision dataset, 2024b.

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 *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 66178–66193. PMLR, 2025. URL https://proceedings.mlr.press/v267/wei25k.html.

Yongxian Wei, Runxi Cheng, Weike Jin, Enneng Yang, Li Shen, Lu Hou, Sinan Du, Chun Yuan, Xiaochun Cao, and Dacheng Tao. OptMerge: Unifying multimodal LLM capabilities and modalities via model merging. In *International Conference on Learning Representations*, 2026a. URL https://openreview.net/forum?id=Me0n0iESJY.

Yongxian Wei, Runxi Cheng, Xingxuan Zhang, Li Shen, Chun Yuan, Peng Cui, and Dacheng Tao. Closed-form spectral regularization for multi-task model merging, 2026b. URL https://arxiv.org/abs/2606.07289.

Mitchell Wortsman, Gabriel Ilharco, Samir Ya Gadre, Rebecca Roelofs, Raphael Gontijo-Lopes, Ari S. Morcos, Hongseok Namkoong, Ali Farhadi, Yair Carmon, Simon Kornblith, and Ludwig Schmidt. Model soups: Averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 23965–23998. PMLR, 2022. URL https://proceedings.mlr.press/v162/wortsman22a.html.

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*, 2016. doi: 10.1007/s11263-014-0748-y.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-Merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 7093–7115. Curran Associates, Inc., 2023. doi: 10.52202/075280-0310. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1644c9af28ab7916874f6fd6228a9bcf-Abstract-Conference.html.

Enneng Yang, Li Shen, Zhenyi Wang, Guibing Guo, Xiaojun Chen, Xingwei Wang, and Dacheng Tao. Representation surgery for multi-task model merging, 2024a. URL https://arxiv.org/abs/2402.02705.

Enneng Yang, Zhenyi Wang, Li Shen, Shiwei Liu, Guibing Guo, Xingwei Wang, and Dacheng Tao. Adamerging: Adaptive model merging for multi-task learning, 2024b. URL https://arxiv.org/abs/2310.02575.

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 *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 57755–57775. PMLR, 2024. URL https://proceedings.mlr.press/v235/yu24p.html.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. Modeling context in referring expressions. In *European Conference on Computer Vision*, 2016.

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, 2026. URL https://arxiv.org/abs/2603.06242.

Xiang Zhang, Junbo Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, 2015.

Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, and Fei Wu. Merging loras like playing lego: Pushing the modularity of lora to extremes through rank-wise clustering. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu (eds.), *International Conference on Learning Representations*, volume 2025, pp. 72896–72913, 2025. URL https://proceedings.iclr.cc/paper_files/paper/2025/file/b54e0146a82945f01e69c2e3309ba925-Paper-Conference.pdf.

Max Zimmer, Christoph Spiegel, and Sebastian Pokutta. Sparse model soups: A recipe for improved pruning via model averaging, 2024. URL https://arxiv.org/abs/2306.16788.

A Theory and Target Derivations

A.1 Proofs of the Analysis Results

Lemma 1. If $\mathbf{U}^\top \mathbf{W}_m = \mathbf{U}^\top \mathbf{W}_k$, left-multiplication by $\mathbf{U}$ gives $\mathbf{P}_\mathbf{U}\mathbf{W}_m = \mathbf{P}_\mathbf{U}\mathbf{W}_k$, and therefore $\mathbf{P}_\mathbf{U} y_m = \mathbf{P}_\mathbf{U} y_k$ for every input x. Conversely, equality of projected outputs for all x implies $\mathbf{P}_\mathbf{U}(\mathbf{W}_m - \mathbf{W}_k) = 0$. Multiplying by $\mathbf{U}^\top$ yields $\mathbf{U}^\top(\mathbf{W}_m - \mathbf{W}_k) = 0$.

Proposition 1. It suffices to construct a rank-one example. Let u be a unit vector in $\mathrm{span}(\mathbf{U}_i)$, let v be a unit vector in $\mathrm{span}(\mathbf{V}_i)$, and let $v_\perp$ be a unit vector orthogonal to $\mathrm{span}(\mathbf{V}_i)$. For $\epsilon > 0$ and $M > \sqrt{3}\epsilon$, define errors

$$\mathbf{E}_a = \epsilon u v^\top + M u v_\perp^\top, \qquad \mathbf{E}_b = 2\epsilon u v^\top, \tag{24}$$

and set $\mathbf{W}_a = \mathbf{W}_i + \mathbf{E}_a$, $\mathbf{W}_b = \mathbf{W}_i + \mathbf{E}_b$. Since $\|\mathbf{U}_i^\top u\|_2 = 1$, the expert-input response errors are $\|\mathbf{U}_i^\top \mathbf{E}_a \mathbf{V}_i\|_F^2 = \epsilon^2$ and $\|\mathbf{U}_i^\top \mathbf{E}_b \mathbf{V}_i\|_F^2 = 4\epsilon^2$, so $\mathbf{W}_a$ is better on expert i's inputs. But the full row errors are $\|\mathbf{U}_i^\top \mathbf{E}_a\|_F^2 = \epsilon^2 + M^2 > 4\epsilon^2 = \|\mathbf{U}_i^\top \mathbf{E}_b\|_F^2$. Hence row matching strictly prefers $\mathbf{W}_b$, even though $\mathbf{W}_b$ is worse on expert i's inputs.

Proposition 2. We prove by construction. The strict containment assumption gives a nonzero vector q satisfying

$$q \in \mathrm{col}(\mathbf{Q}), \qquad \mathbf{U}_i^\top q = 0. \tag{25}$$

Since $\mathrm{span}(\mathbf{V}_i)$ is nonempty, choose a unit vector $v \in \mathrm{span}(\mathbf{V}_i)$ and define

$$\mathbf{W}_m = \mathbf{W}_i + q v^\top. \tag{26}$$

Then

$$\mathbf{U}_i^\top(\mathbf{W}_m - \mathbf{W}_i)\mathbf{V}_i = \mathbf{U}_i^\top q\, v^\top \mathbf{V}_i = 0, \tag{27}$$

so the self-response is matched exactly. However,

$$\mathbf{P}_\mathbf{Q}(\mathbf{W}_m - \mathbf{W}_i)\mathbf{V}_i = q v^\top \mathbf{V}_i \neq 0 \tag{28}$$

because $q \in \mathrm{col}(\mathbf{Q})$ and $v^\top \mathbf{V}_i$ is nonzero. Thus exact agreement on $\mathbf{U}_i^\top \mathbf{W} \mathbf{V}_i$ does not guarantee equality in the whole joint output space $\mathbf{Q}$ on expert i's input directions. If q has a component in another expert output subspace $\mathrm{span}(\mathbf{U}_j)$, this nonzero term contributes to the cross response $\mathbf{U}_j^\top \mathbf{W}_m \mathbf{V}_i$ left uncontrolled by self matching.

Proposition 3. Assume a single $\mathbf{W}$ satisfies both exact self-behavior constraints. Since z lies in both input subspaces and g lies in both output subspaces, the constraints imply $g^\top \mathbf{W} z = g^\top \mathbf{W}_i z$ and $g^\top \mathbf{W} z = g^\top \mathbf{W}_j z$, hence $g^\top \mathbf{W}_i z = g^\top \mathbf{W}_j z$, contradicting the premise. Exact self-behavior matching can therefore be infeasible.

Theorem 1. Let $\mathbf{A} = [\mathbf{U}_1 \cdots \mathbf{U}_K]$. Since $\mathbf{Q}$ spans $\mathrm{col}(\mathbf{A})$, the matrix $\mathbf{Q}^\top \mathbf{A}\mathbf{A}^\top \mathbf{Q}$ is positive definite on the joint output coordinates. Thus for any $\mathbf{Y} \in \mathrm{col}(\mathbf{Q})$,

$$\lambda_{\mathbf{Q}} \|\mathbf{Y}\|_F^2 \le \|\mathbf{A}^\top \mathbf{Y}\|_F^2 = \sum_j \|\mathbf{U}_j^\top \mathbf{Y}\|_F^2. \tag{29}$$

Apply this with $\mathbf{Y}_i = \mathbf{P}_{\mathbf{Q}}(\mathbf{W} - \mathbf{W}_i)\mathbf{V}_i$. Because $\mathbf{U}_j^\top \mathbf{P}_{\mathbf{Q}} = \mathbf{U}_j^\top$, we have

$$\lambda_{\mathbf{Q}} \|\mathbf{Y}_i\|_F^2 \le \sum_j \|\mathbf{U}_j^\top (\mathbf{W} - \mathbf{W}_i)\mathbf{V}_i\|_F^2. \tag{30}$$

The $j = i$ term is the self-response loss. For $j \neq i$,

$$\begin{aligned} \|\mathbf{U}_j^\top (\mathbf{W} - \mathbf{W}_i)\mathbf{V}_i\|_F^2 \le 2\|\mathbf{U}_j^\top \mathbf{W}\mathbf{V}_i - \mathbf{T}_{ji}\|_F^2 \\ + 2\|\mathbf{T}_{ji} - \mathbf{U}_j^\top \mathbf{W}_i \mathbf{V}_i\|_F^2, \end{aligned} \tag{31}$$

by $\|a - b\|^2 \le 2\|a\|^2 + 2\|b\|^2$. Summing over i and dividing by $\lambda_{\mathbf{Q}}$ gives main-paper Eq. (13). The zero-target specialization sets every $\mathbf{T}_{ji} = 0$. More generally, for $\mathbf{T}_{ji} = \beta_{ji}\mathbf{U}_j^\top \mathbf{W}_i \mathbf{V}_i$ with $0 \le \beta_{ji} \le 1$, the corresponding per-pair fixed mismatch is $(1 - \beta_{ji})^2 \|\mathbf{U}_j^\top \mathbf{W}_i \mathbf{V}_i\|_F^2$, no larger than its zero-target value.

Numerical bound check. We evaluate the zero-target specialization of main-paper Eq. (13) on all 49 non-embedding matrix layers of the ViT-B/32 merge. Every observed joint error satisfies the bound. The bound-to-actual ratio has minimum 6.57, median 36.31, 90th percentile 157.72, and maximum $4{,}132.73$. The observed bound is conservative, consistent with the shared-frame eigenvalue and triangle-inequality relaxations.

A.2 SBR-R Target Construction and Proof

Scope and source direction. The main paper defines the source response $\mathbf{A}_{ji}$ and peer response $\mathbf{B}_{ji}$ in main-paper Eq. (18). For each ordered block (j, i), SBR-R restricts the retained target to a scaled copy of $\mathbf{A}_{ji}$. This attenuation-only restriction preserves the observed source direction and uses $\mathbf{B}_{ji}$ only as evidence of peer support. It avoids injecting a direction found only in the peer response. This is a conservative weight-only restriction, not a claim that shared knowledge has a uniquely identifiable semantic direction.

For compactness, write $\mathbf{A} = \mathbf{A}_{ji}$ and $\mathbf{B} = \mathbf{B}_{ji}$. Consider the two response segments

$$\begin{aligned} \mathcal{C}(\mathbf{A}) &= \{\beta \mathbf{A} : 0 \le \beta \le 1\}, \\ \mathcal{C}(\mathbf{B}) &= \{\gamma \mathbf{B} : 0 \le \gamma \le 1\}. \end{aligned} \tag{32}$$

These matrix segments are not intersected. Unless $\mathbf{A}$ and $\mathbf{B}$ are positively collinear, their intersection contains only zero. Instead, SBR-R intersects scalar bounds on the amount that can be retained.

Directed magnitude certificates. Assume first that $\mathbf{A}$ and $\mathbf{B}$ are nonzero; if either is zero, SBR-R defines the target to be zero. Let

$$q = [\langle \mathbf{A}, \mathbf{B} \rangle_F]_+, \qquad a = \|\mathbf{A}\|_F, \qquad b = \|\mathbf{B}\|_F. \tag{33}$$

Projecting each response onto the other's segment gives two retained magnitudes:

$$r_{\mathbf{A}|\mathbf{B}} = \min\{q/a, a\}, \qquad r_{\mathbf{B}|\mathbf{A}} = \min\{q/b, b\}. \tag{34}$$

They define scalar intervals

$$\mathcal{I}_{\mathbf{A}|\mathbf{B}} = [0, r_{\mathbf{A}|\mathbf{B}}], \qquad \mathcal{I}_{\mathbf{B}|\mathbf{A}} = [0, r_{\mathbf{B}|\mathbf{A}}]. \tag{35}$$

Their intersection in $\mathbb{R}_{\ge 0}$ is

$$\begin{aligned} \mathcal{I}(\mathbf{A}, \mathbf{B}) &= [0, m^\star], \\ m^\star &= \min\{r_{\mathbf{A}|\mathbf{B}}, r_{\mathbf{B}|\mathbf{A}}\} = \frac{q}{\max\{a, b\}}. \end{aligned} \tag{36}$$

Orienting this magnitude along the source response gives the radial set

$$\mathcal{R}(\mathbf{A} \mid \mathbf{B}) = \{(m/a)\mathbf{A} : m \in \mathcal{I}(\mathbf{A}, \mathbf{B})\}. \tag{37}$$

Its endpoint is exactly the SBR-R target in main-paper Eq. (19).

Proposition A.4 (Conditional radial optimum)**.** *Fix nonzero* $\mathbf{A}, \mathbf{B}$*. Among source-radial targets* $\mathbf{T} = (m/a)\mathbf{A}$ *whose magnitude* m *lies in both intervals in Eq.* (35)*, the SBR-R target uniquely maximizes* $\|\mathbf{T}\|_F$ *and, equivalently, uniquely minimizes* $\|\mathbf{T} - \mathbf{A}\|_F^2$*. Exchanging the two response views preserves the certified magnitude:*

$$\|\mathbf{T}(\mathbf{A} \mid \mathbf{B})\|_F = \|\mathbf{T}(\mathbf{B} \mid \mathbf{A})\|_F = m^\star. \tag{38}$$

The two targets generally retain different directions.

This proposition compares only attenuation along the chosen source direction. It does not rank peer-directed, mixed-direction, or other target classes, and it does not prove that the retained response is semantically shared. Exchanging $\mathbf{A}$ and $\mathbf{B}$ is a magnitude-symmetry check within one ordered block, not the reverse expert target $\mathbf{T}_{ij}$. Positive response agreement does not guarantee downstream benefit. For example, SBR-R improves the InternVL average over SBR but scores slightly lower on MathVista and OCRVQA (Table 1); these aggregate comparisons do not isolate the effect of individual retained responses.

Directed certificates. For nonzero $\mathbf{A}$, the projection of $\mathbf{B}$ onto $\mathcal{C}(\mathbf{A})$ has coefficient $\Pi_{[0,1]}(\langle \mathbf{A}, \mathbf{B}\rangle_F / a^2)$. Its magnitude is therefore $r_{\mathbf{A}|\mathbf{B}} = \min\{q/a, a\}$. Exchanging the two views gives $r_{\mathbf{B}|\mathbf{A}} = \min\{q/b, b\}$, proving Eq. (34).

Common magnitude and uniqueness. Every admissible source-radial target is represented by one magnitude satisfying

$$0 \le m \le r_{\mathbf{A}|\mathbf{B}}, \qquad 0 \le m \le r_{\mathbf{B}|\mathbf{A}}. \tag{39}$$

Maximum retention therefore uniquely gives the smaller of the two scalar certificates. Since $m^\star \le a$, the same endpoint uniquely minimizes $\|(m/a)\mathbf{A} - \mathbf{A}\|_F^2 = (a-m)^2$ over the derived radial set. Suppose $a \ge b$. Cauchy–Schwarz gives $q/a \le b \le a$, hence $r_{\mathbf{A}|\mathbf{B}} = q/a$. Moreover, $q/b \ge q/a$ and $b \ge q/a$, so $r_{\mathbf{B}|\mathbf{A}} \ge q/a$. Thus $m^\star = q/a = q/\max\{a, b\}$. The case $b \ge a$ is symmetric. Orienting $m^\star$ along $\mathbf{A}$ gives $\mathbf{T} = (m^\star/a)\mathbf{A}$, which is exactly main-paper Eq. (19). Orienting the same magnitude along $\mathbf{B}$ proves Eq. (38). Unless the normalized views coincide, the two oriented matrices differ even though their norms agree. If either view is zero or their Frobenius inner product is nonpositive, both scalar certificates admit only zero magnitude.

Limitation of cross-response target selection. Shared output directions need not imply shared responses to the same input. Consequently, agreement-based retention can suppress a valid source-expert response when the peer response is weak or zero. For example, let $\mathbf{W}_1 = [1\ 0]$ and $\mathbf{W}_2 = [0\ 1]$. Their retained input directions are orthogonal, but their output space is identical. The sum $[1\ 1]$ exactly preserves both experts on their respective inputs, whereas both directed cross blocks have $\mathbf{A}_{ji} = 1$ and $\mathbf{B}_{ji} = 0$. Both SBR and SBR-R therefore set the cross targets to zero. For $\mathbf{W} = [a\ b]$, Eq. (20) becomes

$$\mathcal{L}_s(\mathbf{W}) = \tfrac{1}{2}[(a-1)^2 + (b-1)^2] + \tfrac{1}{\sqrt{2}}(a^2 + b^2), \tag{40}$$

whose minimizer is $a = b = 1/(1+\sqrt{2}) \approx 0.414$. Thus the selected cross targets can conflict with otherwise compatible expert responses. The joint input space is complete in this example, so the Generalist component cannot compensate for this attenuation. Theorem 1 remains valid because it explicitly includes target mismatch; it does not guarantee that our target rule preserves every compatible expert response. This example establishes a limitation of the surrogate objective, not a measured downstream accuracy loss. Future work will investigate source-referenced cross-response targets that distinguish a peer's absent response from genuine conflict, and evaluate them under controlled subspace overlap and on downstream tasks.

A.3 Coherence-Robust Task-Count Scaling

Let $\mathbf{E}_{ji}(\mathbf{W}) = \mathbf{U}_j^\top \mathbf{W} \mathbf{V}_i - \mathbf{T}_{ji}$, with the diagonal target defined by the corresponding expert. Under a uniform task prior, the self-response loss is

$$L_s(\mathbf{W}) = \frac{1}{K} \sum_i \|\mathbf{E}_{ii}(\mathbf{W})\|_F^2. \tag{41}$$

The task vectors identify every cross-response error, but without downstream activations they do not identify how the K task-conditioned cross systems are aggregated. If they are alternatives under a uniform task prior, the cross coefficient is $1/K$. If their constraints are jointly persistent in the merged model, it is 1. We expose this missing aggregation semantics as the ambiguity class

$$\mathfrak{R}_K = \{L_s + \omega L_c : \ \omega \in [1/K, 1]\}, \tag{42}$$

where

$$L_c = \sum_i \sum_{j \neq i} \|\mathbf{E}_{ji}\|_F^2. \tag{43}$$

The interval is an explicit robust-modeling assumption about unresolved aggregation, not an estimate of a latent coherence value.

Choose one coefficient w without downstream data. Since task count changes the admissible scale multiplicatively, define worst-case coefficient distortion

$$\mathcal{D}_K(w) = \sup_{\omega \in [1/K, 1]} \max\left\{\frac{w}{\omega}, \frac{\omega}{w}\right\} = \max\{Kw, 1/w\}. \tag{44}$$

The unique minimizer equalizes the endpoint distortions:

$$w^\star = K^{-1/2}, \qquad \mathcal{D}_K(w^\star) = \sqrt{K}. \tag{45}$$

Thus no other fixed coefficient approximates every risk in Eq. (42) within a smaller multiplicative factor. This is a minimax statement about coefficient distortion, not a claim that the unknown downstream loss itself is minimized.

The same choice has an axiomatic characterization. If $g(K)$ is required to compose across independent task groups, $g(KM) = g(K)g(M)$, and to be self-dual between average and total aggregation, $g(K)^2 = (1/K) \cdot 1$, then the unique positive solution is $g(K) = K^{-1/2}$. Hence the coefficient in Main-paper Eq. (20) uses the only multiplicatively consistent endpoint-neutral normalization. We therefore use it unchanged across every architecture rather than selecting a coefficient from downstream results.

For the Generalist objective, the tail anchor $\mathbf{SP}_{\mathbf{Z}^\perp}$ has the same unresolved average-versus-total aggregation semantics. Applying Eq. (44) again gives its $1/\sqrt{K}$ coefficient, while the observable per-expert output-calibration terms retain unit weight. No validation data enter either objective coefficient. Rank remains a separate protocol configuration.

B Method and Implementation Details

B.1 Boundary with WUDI and SWUDI

WUDI and the SBR family both use task vectors as data-free surrogates for unavailable activations, but they optimize different objects. WUDI observes that the row space of a task vector approximates the corresponding layer-input space and substitutes task vectors into a single layer-wise interference objective, which is optimized iteratively with Adam (Cheng et al., 2025). Finite-step optimization on such quadratic proxies can itself act as a spectral filter, as analyzed explicitly by SWUDI (Wei et al., 2026b). We likewise treat the fixed dense CG trajectory as part of the estimator and do not attribute the dense-model gains to exact minimization of the response loss.

SBR instead defines a paired response for every ordered expert pair,

$$\mathbf{B}_{j \leftarrow i}(\mathbf{W}) = \mathbf{U}_j^\top \mathbf{W} \mathbf{V}_i, \tag{46}$$

uses the explicit expert target $\mathbf{B}_{i \leftarrow i}(\mathbf{W}_i)$ to reconstruct the self-response, and regularizes $\mathbf{B}_{j \leftarrow i}(\mathbf{W})$ toward zero in SBR or toward the bidirectionally certified radial target in SBR-R. Both impose $\mathbf{P}_{\mathbf{Q}^\perp} \mathbf{W} \mathbf{Z} = 0$ and solve a distinct Generalist problem on $\mathbf{P}_{\mathbf{Z}^\perp}$. These ingredients change the reconstruction target and the domain decomposition. They cannot be obtained merely by applying a different inverse or spectral filter to the WUDI normal equation. Table S1 summarizes the distinction.

Table S1: Method boundary relative to WUDI and SWUDI. The distinction is the behavior target and orthogonal input-space decomposition, not simply a different optimizer for the same objective.

Aspect	WUDI	SWUDI	SBR / SBR-R (ours)
Data-free proxy	Task-vector row space for layer inputs	Task-vector row-space Gram operator	Paired top input/output spaces $(\mathbf{V}_i, \mathbf{U}_j)$
Optimized target	Layer-wise interference surrogate	WUDI endpoint under truncated spectral filtering	Self targets $\mathbf{U}_i^\top \mathbf{W}_i \mathbf{V}_i$ and zero / certified radial cross targets
Structural constraint	No SBR-style hard routing constraint	No SBR-style hard routing constraint	$\mathbf{P}_{\mathbf{Q}^\perp}\mathbf{W}\mathbf{Z} = 0$ by construction
Complementary inputs	Not separately reconstructed	Not separately reconstructed	Separate Generalist solve on $\mathbf{P}_{\mathbf{Z}^\perp}$
Numerical solution	Iterative Adam optimization	Analytic eigenspace filter	Fixed 50-step reduced CG for dense Specialists and main-paper Eq. (23) for Generalist

B.2 Comparison with Related Merging Methods

Our starting point is that matching a layer's outputs in an expert's output subspace reconstructs its projected behavior. Without task data, we approximate this alignment using the paired responses $\mathbf{U}_j^\top \mathbf{W}\mathbf{V}_i$: self-responses have explicit expert targets, while cross-responses are suppressed in SBR or retained according to expert agreement in SBR-R. The distinction concerns these response targets; SVD, subspace construction, and the goal of preserving expert behavior are established ideas. Appendix B.1 gives the separate comparison with WUDI and SWUDI.

Spectral and subspace merging. TSV-M (Gargiulo et al., 2025) compresses task matrices and whitens their concatenated singular vectors to reduce interference. Iso-CTS (Marczak et al., 2025) combines common directions from the summed update with additional task-specific directions and equalizes their singular values. These methods construct the merged update by transforming its directions and spectrum. Our objectives instead specify the response that the merged update should produce for each input/output expert pair. DC-Merge (Zhang et al., 2026) also uses both left and right singular spaces: it smooths task spectra, constructs a shared cover basis, and applies a merging rule such as TA or TIES in that basis before masked reconstruction. Thus, using two-sided projections alone does not distinguish our method from DC-Merge. Our distinction is matching each expert's self-response and assigning explicit, potentially nonzero targets to cross-responses in the original expert-specific bases.

LoRA alignment and recombination. KnOTS (Stoica et al., 2024b) jointly decomposes LoRA updates to provide aligned coordinates in which existing merging rules can operate. LoRA-LEGO (Zhao et al., 2025) decomposes adapters into rank-wise parameter units, clusters these units, and constructs a merged adapter from the cluster centroids with scale adjustment. These approaches address alignment and composition of low-rank updates. Our LoRA implementation reconstructs responses of the effective updates, including their LoRA scaling, using the same self/cross objectives as for dense layers (Appendix B.5). Its objective is not defined by clustering adapter factors or by applying an existing aggregation rule after alignment.

Data-free optimization and expert preservation. APGD (Wei et al., 2025) already motivates merging by reducing the gap between the merged model and each expert. It approximates task-loss differences using task vectors and projects optimization updates away from a shared subspace to preserve common knowledge. We share this motivation. Our approximation instead uses matrix-valued layer responses on paired input/output spaces and separates self reconstruction from cross-response target selection. OptMerge (Wei et al., 2026a) uses low-rank task-vector approximations in a layer-wise interference objective, with tailored optimization for full and LoRA fine-tuning. Its dense formulation matches responses on singular-value-weighted input directions after centering task vectors; it does not separately prescribe the self/cross targets used here. We inherit its MLLMerging checkpoints and evaluation protocol, not its merging objective.

Representation correction with data. Representation Surgery (Yang et al., 2024a) shares the goal of recovering expert-like representations, but learns task-specific correction modules from expert and merged features on unlabeled inputs. AdaMerging (Yang et al., 2024b) instead learns merging coefficients from unlabeled data. Our response targets are constructed from task vectors, with no task inputs or learned correction modules required at merge time.

Dynamic merging and task-specific recovery. Twin-Merging (Lu et al., 2024b) separates shared and exclusive knowledge and combines them according to the test input. EMR-Merging (Huang et al., 2024) uses task-specific masks and rescaling to recover experts from a unified model; TALL-masks (Wang et al., 2024a) similarly recover task-specific updates by masking a merged vector. The latter paper also proposes Consensus Merging, a static rule that removes undesirable weights. Task-specific recovery should therefore be distinguished from input-dependent routing and from static merging. Our Specialists and Generalist are components of one fixed update, assembled before inference; they require neither a router nor selection of task-specific weight masks.

Broader applications and scope. MergeKit (Goddard et al., 2024) provides infrastructure for model merging. FuseChat (Wan et al., 2024) first transfers knowledge between heterogeneous chat models through fine-tuning and then merges compatible target models. Language model arithmetic (Dekoninck et al., 2024) composes models during generation, while LoRA task arithmetic for continual learning (Chitale et al., 2023) and joint autoregressive multimodal training (Aiello et al., 2023) include learning from task or multimodal data. These settings connect to the broader goals of multitask learning and expert combination (Caruana, 1997; Jacobs et al., 1991). Our scope is narrower: constructing one static model from compatible experts without accessing their task data. We evaluate how self/cross-response reconstruction supports this setting, without claiming to replace heterogeneous knowledge transfer, continual training, or inference-time composition.

B.3 Layer-Wise Pseudocode

The complete layer-wise procedure is given in Algorithm 1. The next subsection expands its reduced Specialists solver and Generalist evaluation.

Algorithm 1 SBR / SBR-R for one linear layer.

Input Task vectors $\{\mathbf{W}_k\}_{k=1}^{K}$, rank r, variant v
Output Merged task vector $\mathbf{W}_{\text{merge}}$

Step 1: Per-expert SVDs & joint bases
$\mathbf{W}_k \approx \mathbf{U}_k\mathbf{\Sigma}_k\mathbf{V}_k^\top \quad \forall k, \quad \mathbf{S} \leftarrow \sum_k \mathbf{W}_k$
$\mathbf{Q} \leftarrow \mathrm{orth}([\mathbf{U}_1 \cdots \mathbf{U}_K]), \quad \mathbf{Z} \leftarrow \mathrm{orth}([\mathbf{V}_1 \cdots \mathbf{V}_K])$
$\mathbf{P}_\mathbf{Q} \leftarrow \mathbf{Q}\mathbf{Q}^\top, \quad \mathbf{P}_\mathbf{Z} \leftarrow \mathbf{Z}\mathbf{Z}^\top, \quad \mathbf{P}_{\mathbf{Z}^\perp} \leftarrow \mathbf{I} - \mathbf{P}_\mathbf{Z}$
$\mathbf{P}_k \leftarrow \mathbf{U}_k\mathbf{U}_k^\top \quad \forall k$
Step 2: Specialists Subspace
$\mathbf{T}_{ji}^v \leftarrow \mathbf{0}$ if $v = \mathrm{SBR}$; otherwise compute $\mathbf{A}_{ji}, \mathbf{B}_{ji}$
  set $\mathbf{T}_{ji}^v \leftarrow \mathbf{0}$ if either response is zero; otherwise use Eq. (19)
$\mathcal{L}_s^v(\mathbf{W}_s) \leftarrow \frac{1}{K}\sum_i \|\mathbf{U}_i^\top\mathbf{W}_s\mathbf{V}_i - \mathbf{U}_i^\top\mathbf{W}_i\mathbf{V}_i\|_F^2$
  $+\frac{1}{\sqrt{K}}\sum_i\sum_{j\neq i}\|\mathbf{U}_j^\top\mathbf{W}_s\mathbf{V}_i - \mathbf{T}_{ji}^v\|_F^2$
  run reduced CG (50 steps for dense models)
Step 3: Generalist Subspace
$\mathcal{L}_g(\mathbf{W}_g) \leftarrow \sum_k \|\mathbf{U}_k^\top(\mathbf{W}_g\mathbf{P}_{\mathbf{Z}^\perp} - \mathbf{W}_k\mathbf{P}_{\mathbf{Z}^\perp})\|_F^2$
  $+\frac{1}{\sqrt{K}}\|\mathbf{W}_g\mathbf{P}_{\mathbf{Z}^\perp} - \mathbf{S}\mathbf{P}_{\mathbf{Z}^\perp}\|_F^2$
  evaluate Eq. (23)
Step 4: Assemble
$\mathbf{W}_{\text{merge}} \leftarrow \underbrace{\mathbf{P}_\mathbf{Q}\mathbf{W}_s\mathbf{P}_\mathbf{Z}}_{\text{Specialists}} + \underbrace{\mathbf{W}_g\mathbf{P}_{\mathbf{Z}^\perp}}_{\text{Generalist}}$
return $\mathbf{W}_{\text{merge}}$

B.4 LAYER-WISE OPTIMIZATION DETAILS

The Specialists Subspace objective in main-paper Eq. (20) is a convex quadratic in $\mathbf{W}_s$. Its normal equations are

$$\begin{aligned}\sum_{i=1}^{K}\left(\frac{1}{K}\mathbf{U}_i\mathbf{U}_i^\top+\frac{1}{\sqrt{K}}\sum_{j\neq i}\mathbf{U}_j\mathbf{U}_j^\top\right)\mathbf{W}_s\mathbf{V}_i\mathbf{V}_i^\top\\ =\frac{1}{K}\sum_{i=1}^{K}\mathbf{U}_i\mathbf{U}_i^\top\mathbf{W}_i\mathbf{V}_i\mathbf{V}_i^\top\\ +\frac{1}{\sqrt{K}}\sum_{i=1}^{K}\sum_{j\neq i}\mathbf{U}_j\mathbf{T}_{ji}^{v}\mathbf{V}_i^\top.\end{aligned}\tag{47}$$

Eq. (47) is the layer-space form used to derive our solver. We do not run CG over an $m\times n$ iterate. Only $\mathbf{P_Q}\mathbf{W}_s\mathbf{P_Z}$ enters the merged weight, so we parameterize the solve by

$$\mathbf{X}=\mathbf{Q}^\top\mathbf{W}_s\mathbf{Z}\in\mathbb{R}^{q\times z},\qquad \mathbf{P_Q}\mathbf{W}_s\mathbf{P_Z}=\mathbf{Q}\mathbf{X}\mathbf{Z}^\top,\tag{48}$$

where q and z are the column counts of $\mathbf{Q}$ and $\mathbf{Z}$. Define $\widetilde{\mathbf{U}}_i=\mathbf{Q}^\top\mathbf{U}_i$, $\widetilde{\mathbf{V}}_i=\mathbf{Z}^\top\mathbf{V}_i$, $\mathbf{G}_i=\widetilde{\mathbf{U}}_i\widetilde{\mathbf{U}}_i^\top$, and $\mathbf{D}_i=\widetilde{\mathbf{V}}_i\widetilde{\mathbf{V}}_i^\top$. Left-multiplying Eq. (47) by $\mathbf{Q}^\top$ and right-multiplying it by $\mathbf{Z}$ gives the reduced system $\widetilde{\mathcal{A}}(\mathbf{X})=\mathbf{B}$, with

$$\begin{aligned}\widetilde{\mathcal{A}}(\mathbf{X})&=\sum_{i=1}^{K}\left(\frac{1}{K}\mathbf{G}_i+\frac{1}{\sqrt{K}}\sum_{j\neq i}\mathbf{G}_j\right)\mathbf{X}\mathbf{D}_i,\\ \mathbf{B}&=\frac{1}{K}\sum_{i=1}^{K}\mathbf{G}_i(\mathbf{Q}^\top\mathbf{W}_i\mathbf{Z})\mathbf{D}_i\\ &\quad+\frac{1}{\sqrt{K}}\sum_{i=1}^{K}\sum_{j\neq i}\widetilde{\mathbf{U}}_j\mathbf{T}_{ji}^{v}\widetilde{\mathbf{V}}_i^\top.\end{aligned}\tag{49}$$

Thus every CG iterate is a $q\times z$ matrix, and the matrix-vector product is evaluated by the sums in Eq. (49). Neither the full Kronecker normal matrix nor a full layer-shaped CG variable is formed.

For reproducibility, all task updates, bases, reduced operators, and Krylov iterates are converted to `float32`, and the right-hand side is normalized before solving. Dense CLIP, BERT, and InternVL layers use unpreconditioned CG initialized at the normalized backprojection $\mathbf{X}_0=\mathbf{B}/\|\mathbf{B}\|_F$ for exactly 50 iterations, except when the machine-precision residual floor is reached earlier. The resulting 50-step iterate is the reported dense Specialists estimator. It is not presented as the unconstrained minimizer of main-paper Eq. (20). The fixed budget is an iterative regularizer that limits the gain assigned to weak-curvature modes of the reduced normal operator, and it is held constant across every dense layer, model, and main benchmark. Qwen LoRA uses the corresponding SBR or SBR-R target in the same response loss but applies Jacobi-preconditioned CG from zero until

$$\frac{\|\mathbf{B}-\widetilde{\mathcal{A}}(\mathbf{X}_t)\|_F}{\|\mathbf{B}\|_F}\le\epsilon,\qquad \epsilon=10^{-6},\tag{50}$$

with a maximum of 200 iterations. The LoRA effective-update systems have $q=z=128$ in this experiment, and all 200 audited SBR-R layers reach the criterion within the maximum budget. This numerical configuration changes neither the response target nor any loss coefficient and uses no downstream data. Afterward, $\mathbf{Q}\mathbf{X}\mathbf{Z}^\top$ is cast back to the layer's original dtype.

Controlled solver-depth audit. To distinguish the response target from finite-iteration regularization, we freeze the ViT-B/32 checkpoints, task order, rank $r=32$, arithmetic, initialization, merge scale, and eight-task evaluation, and vary only the dense CG stopping budget. All rows in Table S2 call the same solver helper. The finite rows follow the same normalized, unpreconditioned CG trajectory. The converged row continues that trajectory, using residual replacement only after the largest finite budget, until every one of the 52 merged two-dimensional tensor systems (including three embedding tables) has relative residual at most 10^{-4}. This is the uniformly attainable

Table S2: ViT-B/32 solver-depth audit. Accuracy is the unweighted mean over the same eight tasks as main-paper Table 3. Residuals summarize all 52 merged two-dimensional tensor systems. The SBR control uses the same solver implementation as SBR-R, and both targets call the same Generalist helper.

Target	CG stop	Avg. ↑	Med. rel. res.	Max rel. res.
SBR-R	10	87.16	3.31×10^{-1}	5.70×10^{-1}
SBR-R	25	**87.72**	1.29×10^{-1}	2.52×10^{-1}
SBR-R	50	87.33	5.22×10^{-2}	1.10×10^{-1}
SBR-R	100	86.58	1.49×10^{-2}	4.37×10^{-2}
SBR-R	converged	82.01	9.78×10^{-5}	1.00×10^{-4}
SBR	50	86.49	1.19×10^{-1}	2.86×10^{-1}

Table S3: BERT SBR-R accuracy (%) under different CG budgets. Avg. is the unweighted mean over the three merged tasks; all other settings are fixed. Bold indicates the highest average in this scan.

CG budget	AG News	Rotten Tomatoes	CoLA	Avg.
10	98.05	49.62	68.74	72.14
25	98.74	53.00	70.18	73.97
50	98.63	62.76	70.76	**77.38**
100	97.37	50.19	31.16	59.57

`float32` certificate for these ill-conditioned systems. We also rerun the original zero-target SBR at 50 steps through exactly the same helper.

The reported median and maximum residuals decrease monotonically with the stopping budget, whereas downstream accuracy peaks at 25 steps and then falls. The residual-certified endpoint is about 5.7 points below that peak. Thus the reported dense estimator is genuinely an iterative spectral regularizer, and downstream performance cannot be attributed to merely solving the quadratic more accurately. At the common 50-step budget, SBR-R remains 0.84 points above the same-helper SBR control. Because the two targets also call the exact same Generalist implementation, this comparison isolates an additional gain from the response-retaining cross target. At the same time, the roughly 5.7-point peak-to-converged spread is much larger than this matched-budget target gain, so the finite optimization path is a quantitatively important method component rather than an incidental solver detail. Because this sensitivity audit was conducted after fixing the main protocol and uses the test suite itself, we do not retrospectively select 25 steps for main-paper Table 3. All primary dense results retain the fixed 50-step configuration.

BERT solver-depth sensitivity. We also vary only the CG budget for SBR-R on BERT, keeping the expert checkpoints, rank ratio 0.35, and evaluation protocol in Appendix C.5 fixed. Table S3 reports accuracy on the three merged tasks. The 50-step run reproduces the reported result and has the highest average among these budgets; increasing the budget to 100 steps reduces accuracy, particularly on CoLA. This post hoc diagnostic supports treating the finite solver budget as part of the dense estimator, rather than assuming that more iterations improve downstream performance.

The reduced parameterization also fixes the otherwise irrelevant full-space null directions: the Specialists component is zero on input directions in $\mathbf{Z}^{\perp}$ and has no output component in $\mathbf{Q}^{\perp}$. The former is handled separately by the Generalist solve, while the latter enforces the hard routing constraint. Every returned dense iterate is therefore a deterministic degree-50 matrix polynomial applied to the reduced normal operator and right-hand side. This finite trajectory, together with its backprojection initialization, is part of the method definition. The Qwen result is instead the relative-residual-certified reduced solution in Eq. (50).

For reference, the equivalent layer-shaped operator from which the reduced system is derived is

$$\mathcal{A}(\mathbf{R}) = \sum_{i=1}^{K} \left(\frac{1}{K}\mathbf{U}_i\mathbf{U}_i^\top + \frac{1}{\sqrt{K}} \sum_{j \neq i} \mathbf{U}_j\mathbf{U}_j^\top \right) \mathbf{R}\mathbf{V}_i\mathbf{V}_i^\top. \tag{51}$$

For the Generalist Subspace, only $\mathbf{W}_g\mathbf{P}_{\mathbf{Z}^\perp}$ is needed. Let $\mathbf{P}_k = \mathbf{U}_k\mathbf{U}_k^\top$, $\mathbf{S} = \sum_k \mathbf{W}_k$, and $\mathbf{H} = \frac{1}{\sqrt{K}}\mathbf{I} + \sum_k \mathbf{P}_k$. The solution of main-paper Eq. (21) is

$$\mathbf{W}_g^\star\mathbf{P}_{\mathbf{Z}^\perp} = \mathbf{H}^{-1} \left(\frac{1}{\sqrt{K}}\mathbf{S}\mathbf{P}_{\mathbf{Z}^\perp} + \sum_{k=1}^{K} \mathbf{P}_k\mathbf{W}_k\mathbf{P}_{\mathbf{Z}^\perp} \right). \tag{52}$$

We do not form or invert the dense $m \times m$ matrix $\mathbf{H}$. With $a = K^{-1/2}$ and $\mathbf{U} = [\mathbf{U}_1 \cdots \mathbf{U}_K] \in \mathbb{R}^{m \times Kr}$, Woodbury gives

$$(a\mathbf{I} + \mathbf{U}\mathbf{U}^\top)^{-1}\mathbf{Y} = a^{-1}\mathbf{Y} - a^{-2}\mathbf{U}(\mathbf{I} + a^{-1}\mathbf{U}^\top\mathbf{U})^{-1}\mathbf{U}^\top\mathbf{Y}. \tag{53}$$

Thus the only factorization is at most $Kr \times Kr$ (or the numerical joint rank if smaller), and it is shared across all columns of the layer RHS. This is the Woodbury solve included in the overhead measurement.

B.5 Qwen-14B LoRA Reconstruction Details

The Qwen experiment uses LoRA experts rather than dense task vectors. For each LoRA layer and expert i, the effective task update is

$$\Delta\mathbf{W}_i = \frac{\alpha_{\text{LoRA}}}{r_{\text{LoRA}}}\mathbf{B}_i\mathbf{A}_i. \tag{54}$$

Applying either method independently to $\mathbf{A}_i$ and $\mathbf{B}_i$ would ignore this product structure, so we instead reconstruct behavior in the effective update space. The output subspace is the column space of $\mathbf{B}_i$, and the input subspace is the column space of $\mathbf{A}_i^\top$. We run the corresponding SBR or SBR-R response loss on these subspaces for each LoRA layer, producing a merged effective update $\Delta\mathbf{W}$. Its 128×128 reduced system uses Jacobi-PCG with relative tolerance 10^{-6} and a maximum budget of 200 iterations. The merged update is then factorized by SVD back into a LoRA adapter $(\mathbf{B}, \mathbf{A})$. The implementation infers output capacity as the maximum layer-wise sum of input adapter ranks. Here this is $4 \times 32 = 128$. It sets output LoRA alpha equal to that capacity, so the PEFT scaling is one. This algebraic bound represents every possible summed input update without truncation and removes output rank as a separately chosen Qwen hyperparameter.

C Experimental Protocols

C.1 Experimental Configurations

For each architecture and protocol, we use one fixed standalone configuration throughout. The per-layer rank is $r = 32$ on the 8-task CLIP suite, $r = 24$ on the 14-task suite, $r = 56$ with $s = 0.1$ on InternVL2.5-1B, and a rank ratio of 0.35 on BERT-base. For Qwen-14B LoRA, the implementation automatically sets the common PEFT output capacity to the largest layer-wise sum of the input adapter ranks. With four rank-32 adapters this evaluates to 128 and can represent every direct-sum merged update without rank truncation. It is not selected from downstream scores. Dense architectures use the fixed 50-step Krylov estimator. Qwen uses the relative-residual-certified PCG solve in Eq. (50). In every case, $\mathbf{Q}$ and $\mathbf{Z}$ are the orthogonalized joint bases of the per-expert truncated singular vectors. SBR and SBR-R share every setting and differ only in main-paper Eqs. (17) and (19). The main results apply no SVC or other post-hoc value scaling.

Runtime and memory measurement. The ViT-B/32 overhead was measured on one NVIDIA A100-SXM4 GPU with 80 GB memory in a dual-socket host with two AMD EPYC 7473X 24-core CPUs (48 physical cores and 96 hardware threads total). The system ran Ubuntu 22.04.5, Python 3.13.0, PyTorch 2.6.0 compiled for CUDA 12.4, and NVIDIA driver 580.159.03. Task vectors and

merge computations used FP32 without automatic mixed precision. For SBR-R, the timed region includes all per-layer truncated and joint-basis SVDs, the reduced Specialists CG solve, the Generalist Woodbury solve, and assembly of the merged task vector. Checkpoint loading, construction of the initial mean task vector, and downstream evaluation are outside the timed region. The SBR row uses its previously recorded measurement.

For the newly measured rows, we first run one untimed warm-up, then interleave three measured repetitions to reduce order and initialization effects. Immediately before each repetition, we invoke Python garbage collection, clear the PyTorch CUDA cache, reset its peak allocator statistics, and synchronize the device. Wall-clock time is measured with Python's `perf_counter()` between CUDA synchronizations. Main-paper Table 4 reports the median. Peak memory is read from PyTorch's `max_memory_allocated()` statistic, and the table reports the maximum over repetitions. It excludes reserved-but-unused CUDA cache, host memory, checkpoint storage, and evaluation memory. The SBR measurement was recorded in an earlier run and is not included in this interleaved timing comparison.

C.2 CLIP Dataset Details

We evaluate on SUN397 (Xiao et al., 2016), Stanford Cars (Krause et al., 2013), RESISC45 (Cheng et al., 2017), EuroSAT (Helber et al., 2019), SVHN (Netzer et al., 2011), GTSRB (Stallkamp et al., 2011), MNIST (LeCun et al., 1998), and DTD (Cimpoi et al., 2014). These are the standard CLIP model-merging benchmarks used by prior work (Ilharco et al., 2023) and (Li et al., 2026b). We use the same checkpoints and splits, merge $K = 8$ task vectors, and report per-task and average top-1 accuracy. The 14-task scalability experiment in Appendix F.2 adds Caltech-101 (Fei-Fei et al., 2004), CIFAR-100 (Krizhevsky, 2009), FGVC-Aircraft (Maji et al., 2013), Flowers-102 (Nilsback & Zisserman, 2008), Food-101 (Bossard et al., 2014), and Oxford-IIIT Pets (Parkhi et al., 2012) using the same pretrained checkpoint and fine-tuning recipe.

C.3 InternVL2.5 Experiment Details

Checkpoints and expert training. We use the public full-parameter capability experts released with OptMerge/MLLMerging (Wei et al., 2026a). The common base is `OpenGVLab/InternVL2_5-1B`, and $K = 5$ experts are `yongxianwei/InternVL2_5-1B_VQA`, `..._Geometry`, `..._Chart`, `..._OCR`, and `..._Grounding`. We did not retrain these checkpoints. Their upstream construction fine-tunes the base for one epoch with a cosine schedule, learning rate 4×10^{-5}, and warm-up ratio 0.03. The five instruction-tuning corpora contain, respectively, 588K VQA examples (GQA, VQAv2, OK-VQA, LLaVA-Instruct, and CogVLM single- and multi-round data), 190K geometry examples (GeoQA+ and G-LLaVA), 218K chart examples (ChartQA and DVQA), 238K OCR examples (OCRVQA, TextCaps, SynthDoG, LLaVAR, ST-VQA, TextVQA, DocVQA, DeepForm, Kleister Charity, and TabFact), and 135K grounding examples (RefCOCO and Visual Genome). This data composition is inherited from the released experts, and none of these training data are accessed by either of our methods.

Pipeline version and merge configuration. Our evaluation checkout is based on the official MLLMerging repository at commit `8c46d1f4f72beb859c8099b836528dfbfc968bd0` (Wei et al., 2026a). We use its InternVL and VLMEvalKit paths with PyTorch 2.5.1+cu124, Transformers 4.45.2, and FP16 model loading. Merging excludes the vision encoder, language-model head, normalization layers, token embeddings, and bias vectors. SBR or SBR-R is applied to the remaining matrix-valued language-model updates. We fix the global coefficient $s = 0.1$ across every reported merge row and use rank 56 for both methods. The coefficient was chosen after preliminary comparisons on these benchmarks, without a separate validation set; all ten-task rows use the same $s = 0.1$ checkpoints.

Benchmarks and scoring. The ten reported columns are VizWiz (Gurari et al., 2018), GQA TestDev Balanced (Hudson & Manning, 2019), MathVista-MINI (Lu et al., 2024a), MathVision-MINI (Wang et al., 2024b), ChartQA test (Masry et al., 2022), TextVQA validation (Singh et al., 2019), OCRVQA test-core (Mishra et al., 2019), and the validation splits of RefCOCO, RefCOCO+, and RefCOCOg (Yu et al., 2016) and (Mao et al., 2016). The first seven follow the fixed VLMEvalKit dataset identifiers in the checkout. MathVista-MINI and MathVision-MINI use the

pipeline's `deepseek-v4-flash` answer-extraction judge. The three grounding sets use the native InternVL evaluator and report precision at one with IoU at least 0.5. Main-paper Table 1 reports the unweighted mean of these ten percentage scores.

For the baselines, Table S4 gives the settings recovered from the merge records and implementation. Main-paper Table 1 uses $s = 0.1$ for every merge row, with the same excluded parameters and evaluation protocol. For an included matrix, the final checkpoint stores $\theta_0 + s\Delta W$; for SBR and SBR-R, $\Delta W = W_s + W_g$. Thus s scales the complete reconstructed update, not only the Specialists component. Excluded parameters retain their pretrained values.

C.4 Qwen-14B Protocol and Checkpoints

Checkpoints and training provenance. The exact base is `Qwen/Qwen-14B` (Bai et al., 2023). We use the four public Twin-Merging adapters (Lu et al., 2024b): `lu-vae/qwen-dolly` (the released MMLU expert), `lu-vae/qwen-truthfulqa-merged`, `lu-vae/qwen-bbq-merged`, and `lu-vae/qwen-cnn-merged`. All four target `c_attn`, `c_proj`, `w1`, and `w2`, with rank 32 and dropout 0.05. The released MMLU and CNN adapters use LoRA alpha 16, whereas TruthfulQA and BBQ use alpha 32. We merge the effective updates using each adapter's own scaling. The TA, TIES, DARE, WUDI, SWUDI, and SWUDI-A runs first express all adapters with common PEFT scale 0.5 while preserving every expert's effective update exactly. Their output adapters retain rank 32 and alpha 16.

We use the released adapters without additional fine-tuning. The individual-expert reference evaluates each adapter on its corresponding task with its native LoRA scaling; it is not a single merged model or a strict performance upper bound.

Evaluation protocol. We reproduce the local Twin-Merging protocol (Lu et al., 2024b) with HELM (Liang et al., 2023) at Twin-Merging commit `649faedf09c5659c9918bfda05937a60c0e51f1e` and HELM-Extended-Local commit `a306d8a7a3b4b485cc0db7e12f91c3f42ac018ec`. MMLU (Hendrycks et al., 2021) evaluates the test portions of the 55 subjects explicitly listed by the released `run_moe2_mmlu.conf`, using per-subject caps of 19, 29, or 39 examples and five in-context examples drawn from its training-designated auxiliary/dev pool. TruthfulQA (Lin et al., 2022) uses the original 817-question CSV, the HELM `mc_single` construction, its deterministic 20/80 train/validation partition, and the first 399 validation instances. BBQ (Parrish et al., 2022) pools all released categories and evaluates 399 instances. These three use the HELM joint multiple-choice template, which contains the task instruction, `Question` or `Passage`, lettered options, and `Answer:`. Evaluation uses five demonstrations, greedy decoding, and exact match. CNN/DailyMail (Nallapati et al., 2016) uses the non-anonymized HELM bundle, a fixed 99-example sample (53 test and 46 validation examples), five demonstrations, articles truncated to 512 words, the prompt `Summarize the above article in 3 sentences.`, at most 128 generated tokens, temperature 0.3, and test-split ROUGE-2. Main-paper Table 2 is thus a reproduction of this fixed sampled protocol, not a claim of full-dataset benchmark evaluation.

C.5 BERT Merging Protocol

We use `bert-base-uncased` (Hugging Face revision `86b5e0934494bd15c9632b12f734a8a67f723594`) with a two-class `BertForSequenceClassification` head (Devlin et al., 2019). Three experts are fine-tuned on the training splits of AG News (Zhang et al., 2015), Rotten Tomatoes (Pang & Lee, 2005), and CoLA (Warstadt et al., 2019). AG News is made binary by retaining labels 0 and 1. The other two datasets use their native binary labels. The recorded expert recipe uses seed 42, one epoch, batch size 16, 500 warm-up steps, weight decay 0.01, and the Transformers default AdamW learning rate 5×10^{-5}. It tokenizes with the base uncased tokenizer, truncation, and maximum-length padding. The full classifier state is merged, including both the encoder and the two-class head.

Evaluation uses AG News test, Rotten Tomatoes test, and CoLA validation with maximum length 128, batch size 128, and accuracy. We report CoLA accuracy, rather than its conventional Matthews correlation, to keep the three task columns on one scale. The reported BERT result is the unweighted

average over these three merged tasks. Under the matched solver, SBR reaches 69.78%, while SBR-R reaches 77.38%.

C.6 Baseline Configurations and Selection

Result sources. The eight-task CLIP baseline rows other than WUDI and SWUDI/SWUDI-A are taken from Li et al. (2026b), as specified in Appendix F.1; they are not local retuned baselines. Qwen and BERT scores have local evaluation records. InternVL baseline scores are likewise linked to local merged checkpoints and the common ten-task evaluation. Table S4 summarizes the baseline configurations.

Table S4: Baseline settings. Mask rates denote the fraction discarded. s scales a merged dense update; for Qwen factor methods, the stated coefficient acts separately on the LoRA factors. Here d is the smaller matrix dimension.

Architecture	Method	Configuration
CLIP	WUDI	Adam, 300 steps, learning rate 10^{-5}, $s = 1$.
	SWUDI / SWUDI-A	SWUDI: exponential filter, time 1300, rank ratio 0.65, sum initialization, $s = 1$. SWUDI-A: participation-square-root rank, no exponential filter.
InternVL	TA / TIES / DARE	$s = 0.1$; TIES discards 80% by magnitude; DARE discards 20% randomly, rescales survivors, then applies TA; seed 42.
	TSV-M / Iso-CTS	$s = 0.1$; TSV-M retains $\lfloor d/5 \rfloor$ singular modes per expert; Iso-CTS allocates 80% to the common space, with equal task-specific allocations after rounding.
	WUDI	$s = 0.1$, Adam, 300 steps, learning rate 10^{-5}.
	SWUDI / SWUDI-A	$s = 0.1$; SWUDI: exponential filter, time 300, rank ratio 0.65, sum initialization. SWUDI-A: participation-square-root rank, no exponential filter.
Qwen	TA / TIES / DARE	Factor coefficient 0.3; TIES and DARE discard 70%; DARE rescales survivors before TA.
	TSV-M / Iso-CTS	Effective-update merging; TSV-M keeps all 32 modes per expert; Iso-CTS common fraction 0.8. Output rank/alpha $128/128$.
	WUDI / SWUDI / SWUDI-A	WUDI: 300 steps, learning rate 10^{-5}. SWUDI: time 1300, rank ratio 0.65, exponential filter, sum initialization. SWUDI-A uses participation-square-root rank without the filter.
BERT	WA / TA / TIES / DARE	WA averages updates; TA uses $0.3 \sum_i W_i$; TIES keeps 20% with disjoint sum and final scale 1; DARE drops 80%, rescales, and averages.
	TSV-M / Iso-C / Iso-CTS	TSV-M retains $\lfloor d/3 \rfloor$ modes per expert; Iso-C equalizes all singular values of the summed matrix; Iso-CTS common fraction 0.8.
	SWUDI	Time 1300, rank ratio 0.65, exponential filter, sum initialization.

LoRA representation and capacity. TA, TIES, DARE, WUDI, and SWUDI/SWUDI-A merge the LoRA factors and retain rank/alpha $32/16$. TSV-M, Iso-CTS, SBR, and SBR-R operate on effective updates and store rank/alpha $128/128$. Thus all use the same experts and evaluation, but the table does not impose equal output adapter capacity. The factor-merging implementation normalizes adapters to a common PEFT scale while preserving each expert's effective update.

D HELD-OUT BEHAVIORAL VALIDATION

D.1 DOES THE DATA-FREE PROXY TRACK REAL BEHAVIOR?

To test whether our data-free layer-wise measures reflect real network behavior, we analyze held-out examples after merging. Each model is merged from the same eight task vectors without examples. We then record expert-layer inputs and compare final features from the complete expert and merged encoders. Appendix D.2 gives the protocol and metric definitions.

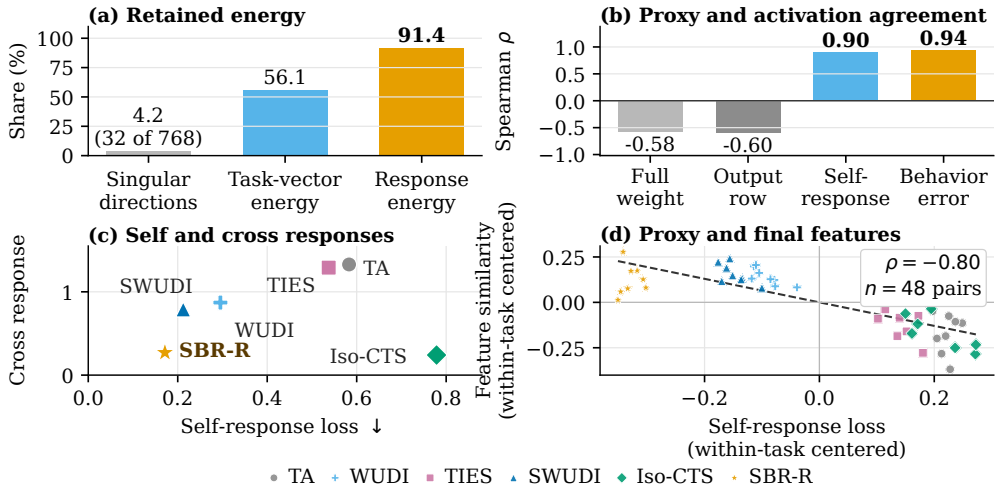


Figure S1: Behavioral evidence. **(a)** Top $32/768$ directions retain 56.1% weight-update and 91.4% held-out response energy. **(b)** Spearman ρ for data-free versus activation-weighted held-out metrics: self-response and behavior error agree; full-weight/output-row errors do not. **(c)** Held-out self-/cross-response losses. **(d)** Lower data-free self-response loss corresponds to higher final-feature similarity over 48 method-task pairs ($\rho = -0.795$). Both axes subtract the mean across methods within each task. Held-out data serve analysis only; feature similarity is not itself a decision metric.

Figure S1(a) shows that the top 32 of 768 directions capture 56.1% of weight-update energy and 91.4% of response energy on held-out inputs, yielding a small but behavior-rich Specialists subspace. Figure S1(b) shows that data-free and held-out measurements agree for self-response loss ($\rho = 0.896$) and the combined response loss ($\rho = 0.935$). Figure S1(c) shows that SBR-R has the lowest self-response loss while keeping cross response below most baselines. Figure S1(d) checks the complete network. Across 48 method-task pairs, lower data-free self-response loss corresponds to more similar final expert and merged features ($\rho = -0.795$ after within-task centering). This pooled similarity correlates with accuracy ($\rho = 0.784$), while directly comparing all local activations does not ($\rho = -0.052$). Yet a closer final feature does not guarantee a closer prediction: on the same 8×64 images, SWUDI has lower head-normalized feature distance than SBR-R (0.610 versus 0.662), but SBR-R has lower centered-logit error (0.338 versus 0.371) and lower predictive JS divergence (0.047 versus 0.055). The fixed head is more sensitive to the directions in SWUDI's feature error. Thus the layer proxy and feature correlation are useful diagnostics, while the head output supplies the direct decision-level check.

D.2 HELD-OUT CLIP BEHAVIOR: PROTOCOL AND METRICS

This analysis tests the task-vector proxy using images, but the images are never used to construct or tune a merge. We construct TA, TIES, Iso-CTS, WUDI, SWUDI, and SBR-R from the same eight ViT-B/32 task vectors. Table S5 additionally reports TSV-M and the existing SBR diagnostic. Both variants use rank 32. Separately, we reserve the first 32 test images from each domain for this post hoc analysis and run them through the corresponding fine-tuned expert. Forward pre-hooks record the inputs to the `c_fc` and `c_proj` maps in all 12 transformer blocks. We deterministically retain

256 evenly spaced token rows per layer, yielding $8 \times 24 = 192$ expert-layer cases. The same cached inputs diagnose the six methods in this held-out audit, for 1,152 combinations of methods, experts, and layers. The seed is 42 throughout.

For the spectral-coverage analysis, both MLP task-update matrices have at most 768 singular directions, so rank 32 retains $32/768 = 4.2\%$ of the available directions. If $\sigma_{i,k}$ denotes the kth singular value of task vector $\mathbf{W}_i$, its rank-r spectral coverage is $\sum_{k=1}^{r} \sigma_{i,k}^2 / \|\mathbf{W}_i\|_F^2$. For recorded examples $\mathbf{X}_i$, the corresponding expert-response coverage is

$$c_i^{\mathrm{upd}} = \frac{\|\mathbf{X}_i \mathbf{V}_i (\mathbf{W}_i \mathbf{V}_i)^\top\|_F^2}{\|\mathbf{X}_i \mathbf{W}_i^\top\|_F^2}. \tag{55}$$

Averaging over all expert and layer cases gives 56.1% task-vector energy and 91.4% expert-response energy.

For each expert i and layer, let $\mathbf{X}_i \in \mathbb{R}^{N \times n}$ contain the recorded input rows and let $\widehat{\mathbf{X}}_i = \mathbf{X}_i \mathbf{V}_i$. The denominator

$$d_i = \|\widehat{\mathbf{X}}_i (\mathbf{U}_i^\top \mathbf{W}_i \mathbf{V}_i)^\top\|_F \tag{56}$$

is the magnitude of the expert's top self-response on those inputs. We define

$$e_{i,\mathbf{X}}^{\mathrm{self}} = \frac{\|\widehat{\mathbf{X}}_i [\mathbf{U}_i^\top (\mathbf{W} - \mathbf{W}_i) \mathbf{V}_i]^\top\|_F}{d_i}, \tag{57}$$

$$c_{i,\mathbf{X}}^{\mathrm{cross}} = \frac{\left(\sum_{j \neq i} \|\widehat{\mathbf{X}}_i (\mathbf{U}_j^\top \mathbf{W} \mathbf{V}_i)^\top\|_F^2\right)^{1/2}}{d_i}, \tag{58}$$

$$\ell_{i,\mathbf{X}}^{\mathrm{route}} = \frac{\|\widehat{\mathbf{X}}_i (\mathbf{P}_{\mathbf{Q}^\perp} \mathbf{W} \mathbf{V}_i)^\top\|_F}{d_i}, \tag{59}$$

$$e_{i,\mathbf{X}}^{\mathrm{beh}} = \left[(e_{i,\mathbf{X}}^{\mathrm{self}})^2 + (c_{i,\mathbf{X}}^{\mathrm{cross}})^2 + (\ell_{i,\mathbf{X}}^{\mathrm{route}})^2\right]^{1/2}. \tag{60}$$

These are activation-weighted versions of the three terms defined by the data-free objective. Across the six methods in this held-out audit, the data-free and activation-weighted self-response losses correlate at $\rho = 0.896$, and the corresponding combined diagnostics correlate at $\rho = 0.935$. We also retain the unrestricted local activation error

$$e_{i,\mathbf{X}}^{\mathrm{full}} = \frac{\|\mathbf{X}_i (\mathbf{W} - \mathbf{W}_i)^\top\|_F}{\|\mathbf{X}_i \mathbf{W}_i^\top\|_F}, \tag{61}$$

rather than excluding it from the audit. It evaluates a different target: every reduction of an off-diagonal expert component and every change in the complementary subspace count as error. Its task-centered correlation with accuracy is only $\rho = -0.052$. Neither the full local error nor the cross diagnostic independently identifies which off-diagonal components affect task predictions.

Table S5 reports the mean of each per-case normalized quantity. TA and TIES have substantial cross response. TSV-M has moderate self-response loss but retains substantial cross response and route leakage. Iso-CTS lowers cross response but loses most of the self-response. WUDI and SWUDI reduce self-response loss but retain cross response. SBR gives the strongest suppression, while SBR-R trades some of it for lower self-response loss. Their shared hard routing constraint makes leakage numerically negligible.

For the independent whole-network check, each raw analysis image is passed through the complete fine-tuned expert and complete merged encoder. Let $f_i(x)$ and $f_{\mathbf{W}}(x)$ denote their final normalized image features. We average $\langle f_i(x), f_{\mathbf{W}}(x) \rangle$ within each method-dataset pair. This comparison allows the two networks to follow different hidden-state trajectories and does not reuse a response block. We average the data-free self-response loss over the 24 analyzed layers, subtract the per-dataset mean from both quantities, and compute Spearman correlation over all $6 \times 8 = 48$ pairs, including SWUDI. The resulting correlation is $\rho = -0.795$. The same centered final-feature cosine correlates with centered accuracy at $\rho = 0.784$. By contrast, the unrestricted local activation error in Eq. (61) correlates with accuracy at only -0.052. The activation-weighted combined behavior diagnostic has correlation -0.827, but because it deliberately contains the cross term, we treat that value only as a regularizer diagnostic and not as independent semantic validation. All correlations are descriptive associations over one fixed expert checkpoint set.

Table S5: Behavior diagnostics on test images reserved for post hoc analysis, averaged over 192 expert-layer cases. Every merge is constructed without these examples. Self-response loss measures failure to reproduce the intended expert response. Cross measures the interference surrogate inside the joint expert-output space, whereas leak measures routed response outside it. Best and second-best values are in **bold** and underlined.

Method	Self loss ↓	Cross ↓	Leak ↓	Behavior ↓
TA	0.583	1.328	0.0913	1.510
TIES	0.537	1.290	0.2855	1.466
TSV-M	0.353	0.951	0.3464	1.082
Iso-CTS	0.779	<u>0.241</u>	0.3009	0.882
WUDI	0.295	0.871	0.1159	0.939
SWUDI	0.212	0.789	0.0308	0.836
SBR	<u>0.208</u>	**0.091**	<u>0.0055</u>	**0.230**
SBR-R	**0.171**	0.270	**0.0052**	<u>0.331</u>

D.3 Fixed-Head Decision Audit

For a separate check on the first 64 test images per task, we run the complete expert, SWUDI, and SBR-R encoders on identical float32 inputs. The ViT-B/32 visual encoder projects its final CLS token into a 512-dimensional image feature f, and each saved task head normalizes $u = f/\|f\|_2$ before applying the same fixed linear map $z = \mathbf{C}u + b$ to expert and merged features. Hence their logit difference is exactly $\delta z = \mathbf{C}(u_m - u_e)$. With $\bar{c}$ the mean row of $\mathbf{C}$, decompose it as $\delta z = (\mathbf{C} - \mathbf{1}\bar{c}^\top)(u_m - u_e) + \mathbf{1}\bar{c}^\top(u_m - u_e)$. The latter is a common shift of every class score, so it changes neither the predicted class nor the softmax distribution. On eight-task equal averages, SWUDI/SBR-R have normalized-feature L2 $0.610/0.662$, but class-centered relative logit L2 $0.371/0.338$ and predictive JS divergence $0.0554/0.0467$; SBR-R has lower JS on every task. Per unit feature error, the class-contrast logit error is $23.36/20.00$, with SBR-R lower on all eight tasks. This algebraic decomposition explains why the two feature-distance and prediction-distance rankings can differ. It does not identify which earlier layer caused the class-relevant difference. In particular, the unshifted local self-response metric favors SBR-R in 11 of 12 `c_fc` blocks, but favors SWUDI in all 12 `c_proj` blocks; its aggregate advantage should not be read as uniformly better local fidelity near the head.

Figure S2 reports a complementary 48-point audit for six methods, including SWUDI and SBR in place of the classical TA and TIES baselines shown in Fig. 1. The local proxy and predictive JS show an overall descriptive association, but their rankings need not agree for nearby methods: SBR has a slightly smaller mean local-response error than SWUDI (0.208 versus 0.212), yet a slightly larger predictive JS (0.0578 versus 0.0554). This reinforces the distinction between the data-free layer objective and the measured decision behavior.

D.4 Cross-Architecture Decision Audit on Real Inputs

Figures S3 and S4 compare merged models with their task experts on real evaluation inputs. Each method–task point averages 64 inputs; the Qwen inset magnifies its low-error region. Colors and marker shapes follow Fig. 1.

The horizontal axis measures the merged update's relative response error in the expert's leading input and output directions: Qwen uses its first MLP expansion map (rank 32), and InternVL uses language MLP layer 12 of 24 (rank 56). Qwen includes each expert's LoRA α_i/r_i factor. For InternVL, the horizontal axis uses the intrinsic update $(\theta_{\text{eval}} - \theta_0)/0.1$ before the common global evaluation scale; the vertical axis uses the actually evaluated checkpoint at scale 0.1. The two axes therefore describe different stages of the same merge, not two measurements of one checkpoint. The InternVL expert reference applies mergeable language-matrix updates to the common base, holding the vision encoder and other excluded modules fixed.

The vertical axis is Jensen–Shannon (JS) divergence between full-vocabulary next-token distributions at the end of an identical prompt. Qwen uses MMLU, TruthfulQA, BBQ, and CNN/DailyMail;

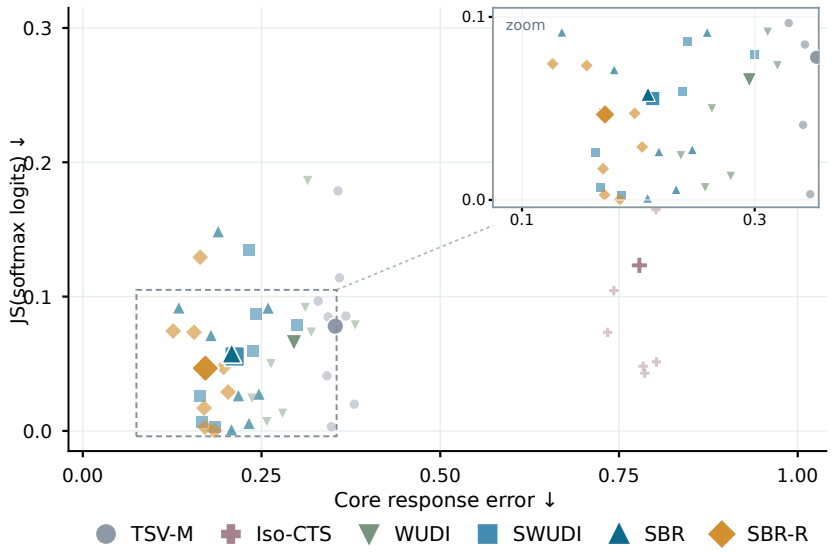


Figure S2: Complementary six-method audit of 48 method–task observations, including SWUDI and SBR; Fig. 1 instead shows six methods including TA and TIES. The horizontal axis is the unshifted local rank-32 MLP response error from 32 expert test images per task; the vertical axis is the expert–merged predictive JS on 64 test images under the same fixed head. Large symbols are method means; the inset magnifies the outlined dense region. The association is descriptive and does not identify a causal layer or direction.

InternVL uses GQA, MathVista, ChartQA, TextVQA, and RefCOCO, with the original image tiling and conversation template. This measures one token position, not the complete answer.

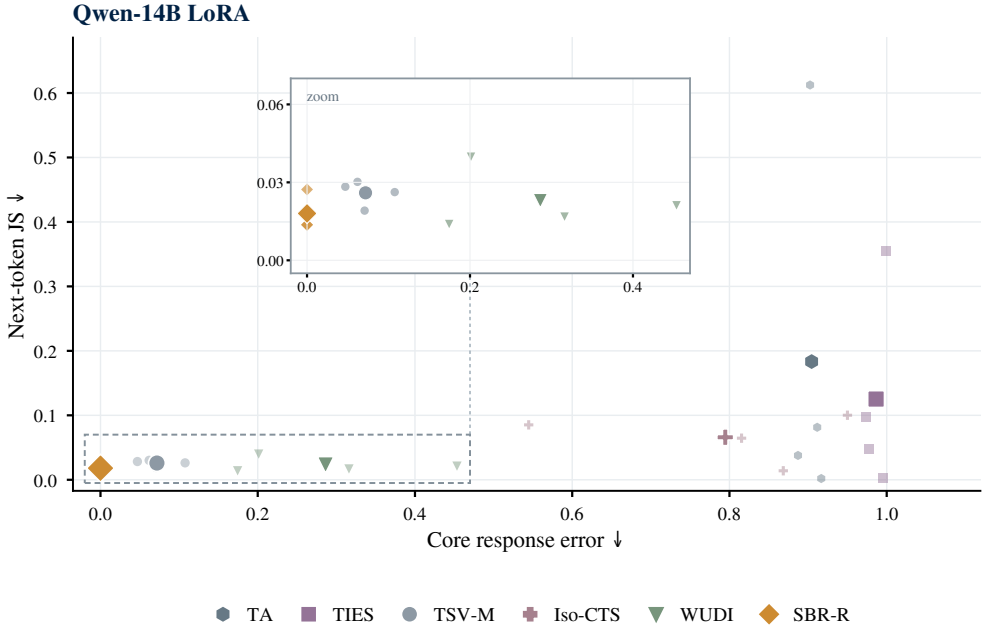


Figure S3: Qwen-14B LoRA. Small points are task means; large points average four tasks. Lower is better on both axes.

On Qwen, SBR-R has the lowest four-task mean next-token JS (0.018), compared with 0.023 for WUDI and 0.026 for TSV-M; its next-token expert agreement is 90.2%. Its first-layer core response error is below 2×10^{-6} for every task. This near-zero horizontal value is expected because the converged rank-128 LoRA solver directly fits those projected self responses; the held-out next-token distribution is the separate behavioral check.

Figure S4 shows that SBR-R has the lowest pre-scale core response error on all five tasks: its layer-12 mean is 0.100, versus 0.417 for TA. The same ordering holds at layer 0 (0.120 versus 0.384), supporting a genuine local reconstruction advantage rather than a fortunate layer choice. Yet TA has slightly lower mean evaluated next-token JS (0.300 versus 0.311), while SBR-R obtains higher ten-benchmark accuracy (59.80% versus 57.79%). Grounding JS is near its $\ln 2$ upper bound for every method, and none shares the expert's first-token argmax on those 64 inputs. Geometry agreement is only 3.1–4.7%; one token cannot summarize multi-token answers or localization quality.

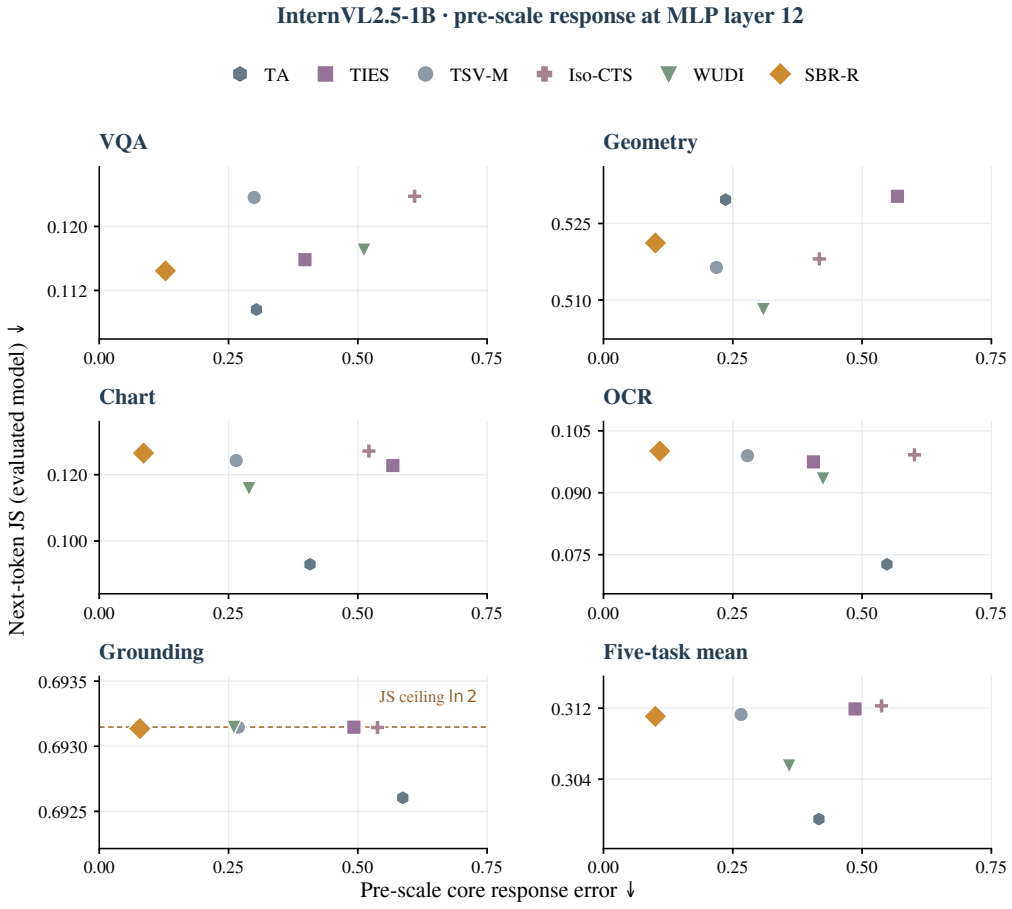


Figure S4: InternVL2.5-1B. The horizontal axis measures pre-scale core response error at language MLP layer 12; the vertical axis measures next-token JS of the scale-0.1 evaluated models. Tasks are separated because absolute JS varies strongly by task. Vertical ranges differ; the final panel is the five-task mean. Lower is better on both axes.

The common 0.1 scale explains why the evaluated-checkpoint local diagnostic reverses this ordering: at layer 12 its errors are 0.902 for SBR-R and 0.873 for TA. At layer 0, a separate rank-56 weight-only check gives 0.021 versus 0.246 before scaling but 0.900 versus 0.890 after it; TA's summed raw update has a larger component along the expert target. The final scorer instead evaluates complete answers. On all 2,500 ChartQA questions, SBR-R alone is correct on 274 cases,

versus 67 for TA; on the first 64 diagnostic questions, both score 53.1% despite TA's lower next-token JS (0.093 versus 0.127). TA appends a percent sign to 197 answers versus 26 for SBR-R; the scorer reads `22%` as 0.22, not 22. Diagnostic rescoring after stripping only TA's trailing percent signs raises its ChartQA score from 62.24% to 68.08%, still below SBR-R's official 70.52%.

D.5 BERT Activation Audit

Separately, we audited local activation responses for the existing fixed TA, SBR, and SBR-R BERT merges. This is *not* the four-way ablation in Figure S8. We recorded inputs to the first and last MLP expansion maps on the first 24 held-out evaluation examples per task (test for AG News and Rotten Tomatoes, validation for CoLA), retaining at most 256 evenly spaced valid token rows per task and layer. No held-out text was used to construct or tune the merges. With the same rank ratio 0.35, SBR lowers mean first-layer activation-weighted self loss across three tasks from 0.731 (TA) to 0.439, and mean cross response from 1.009 to 0.416; at the last layer, the respective values change from 0.671 to 0.293 and from 0.416 to 0.160. However, mean final CLS-feature relative error against the corresponding expert is 0.875 for TA, 0.929 for SBR and 0.924 for SBR-R. Thus this BERT analysis supports the narrow claim that the local proxy tracks some local activation responses, but it does *not* establish better whole-network fidelity or that suppressed cross responses were harmful. The script, sampled protocol and per-task measurements are retained under `experiments/cross_arch_behavior/`.

E Weight-Space Diagnostics and Sensitivity

E.1 Weight-Space Self-Response Gap

Figure S5 provides a separate weight-only, single-layer view of the CLIP self-response gaps discussed above.

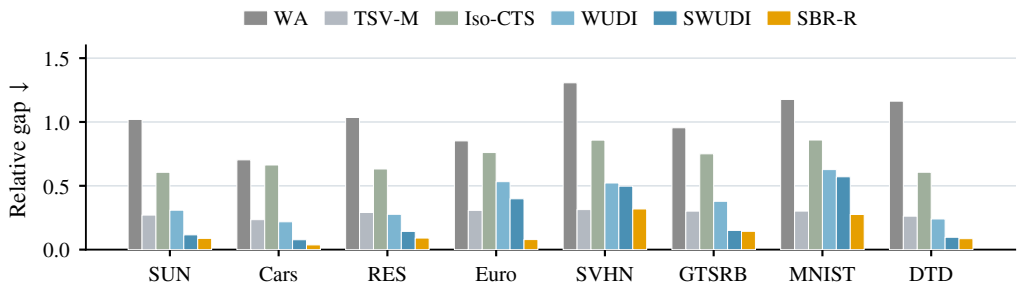


Figure S5: Relative self-response gap $\|\mathbf{U}_k^\top(\mathbf{W} - \mathbf{W}_k)\mathbf{V}_k\|_F / \|\mathbf{U}_k^\top \mathbf{W}_k \mathbf{V}_k\|_F$ on the `visual.proj` layer of ViT-B/32. SBR-R gives the lowest gap on most experts. This weight-only diagnostic does not measure the complete network's predictions.

E.2 Output-Row Analysis on Qwen and InternVL

Figures S6 and S7 repeat the two measurements in Fig. 2, with the same seven merging methods and 1,024 shared Gaussian probes. For each expert, the first MLP expansion map supplies the top input/output singular spaces; panel (a) compares its normalized output-row difference with the response difference on random inputs or on the expert input basis $\mathbf{V}_i$. Panel (b) splits the squared row difference into the $\mathbf{V}_i$ and $\mathbf{V}_i^\perp$ terms of Eq. (4). These are weight-only, layer-local checks; they do not use task examples or measure final predictions.

Qwen uses the effective $\alpha_i/r_i\,\mathbf{B}_i\mathbf{A}_i$ updates from its four released LoRA experts, rank 32, and the same adapter-level merge routines as the reported Qwen experiment. The SBR-R point uses the converged LoRA PCG solver. We evaluate projected products from the exact LoRA factors without

materializing the full 13,696 × 5,120 update. InternVL uses the first language-block `up_proj` matrix, five experts, and rank 56. For this intrinsic comparison, we divide each evaluated merged checkpoint's update by the common global scale 0.1 before measuring it against the raw expert task vectors; the analysis of the scaled evaluated deltas is also retained with the figure data. The raw-space view isolates the merger's response reconstruction from a scale shared by all methods.

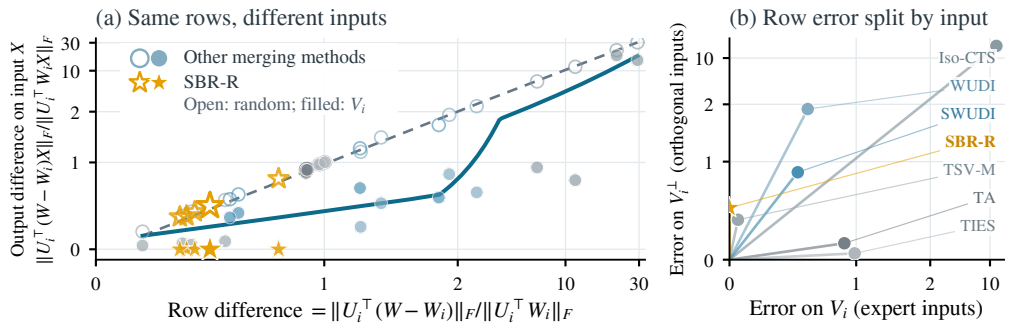


Figure S6: The Fig. 2 audit on Qwen-14B LoRA's first `w1` map (four experts, rank 32). Open markers use the same 1,024 Gaussian probes; filled markers use $\mathbf{V}_i$. Each nonstar marker is another merging method, and panel (b) separates row difference on $\mathbf{V}_i$ from that on $\mathbf{V}_i^\perp$. Symmetric-log spacing beyond 1.5 on both panels keeps the Iso-CTS outliers visible; plotted values are unchanged.

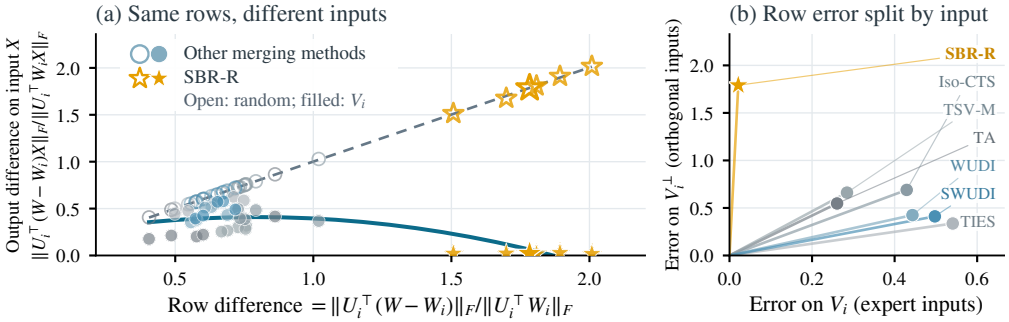


Figure S7: The same audit on InternVL2.5-1B's first language MLP `up_proj` map (five experts, rank 56), using merger updates before the common 0.1 checkpoint scale. SBR-R has a larger full-row difference than several baselines but a smaller response difference on $\mathbf{V}_i$, showing why the two errors should be distinguished. This is a layer-local weight diagnostic, not an evaluation of task accuracy.

E.3 BERT Component Ablation

Figure S8 repeats the four-way component ablation of Figure 3 on the first BERT MLP expansion map for the existing AG News, Rotten Tomatoes, and CoLA experts. It uses the same sum of expert task vectors for weight averaging, the same no-cross and no-self objective ablations, and the same SBR-R reconstruction routine as the ViT-B/32 figure, with the BERT protocol's rank 269 of 768. Each cell is $\|\mathbf{U}_i^\top \mathbf{W}\mathbf{V}_j\|_F$ divided by output expert i's self target, exactly as in Figure 3. The ablation thus measures the same *weight-space* quantity, not held-out activations or task harm.

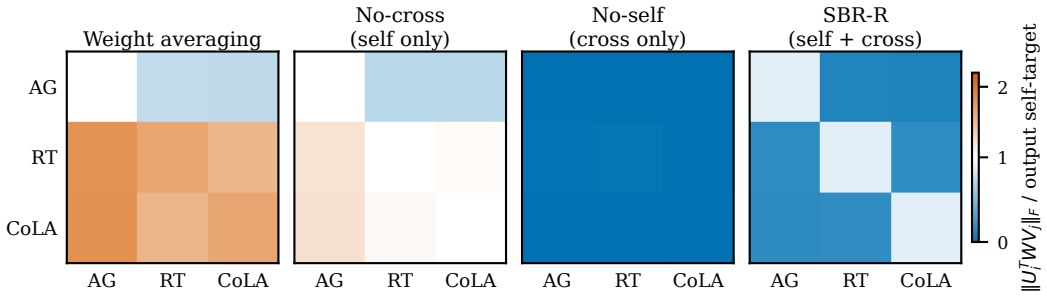


Figure S8: The same four-way component ablation and row-normalized projected weight-response metric as Figure 3, on BERT's first MLP expansion map. AG = AG News; RT = Rotten Tomatoes. The shared component-ablation color scale maps normalized response 1 to white. It shows that no-cross preserves self response but leaves cross response, whereas no-self nearly removes both; SBR-R balances them. These are weight-space surrogates and do not establish that off-diagonal response is harmful.

E.4 Qwen and InternVL Component Ablations

We apply the same four-way weight-space diagnostic to one representative MLP expansion map per architecture: the first-layer `w1` effective LoRA update in Qwen-14B, and the first language-block `up_proj` task vector in InternVL2.5-1B. The four Qwen expert updates are $(\alpha_i/r_i)\mathbf{B}_i\mathbf{A}_i$ with their individual released LoRA scales; we represent these low-rank updates in a joint orthonormal coordinate system that preserves every projected response norm. Their per-expert rank is 32. InternVL uses the five OCR, VQA, geometry, chart, and grounding experts at rank 56. Figures S9 and S10 use exactly the same four objective ablations, output-expert row normalization, and shared 0–2.2 color scale as Figure 3.

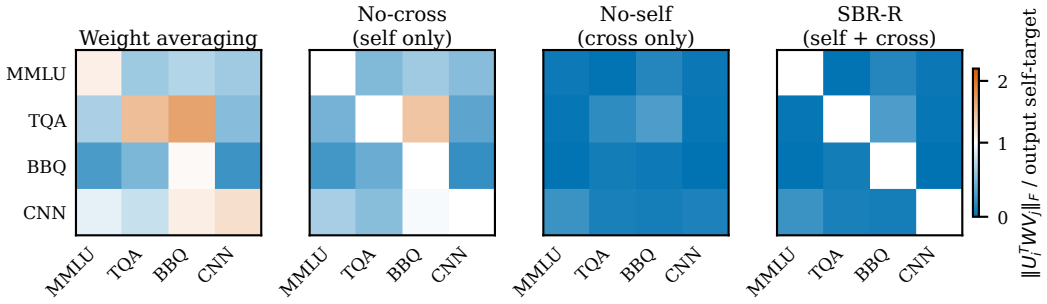


Figure S9: Fig. 3's weight-space component ablation on the effective update of Qwen-14B's first `w1` LoRA map (four experts, rank 32). TQA = TruthfulQA, CNN = CNN/DailyMail. Cells are normalized by the output expert's own self-response. This diagnostic uses the original figure's fixed-budget reconstruction; the reported Qwen accuracy instead uses the converged LoRA PCG solver. No activation or task accuracy is inferred from this heatmap.

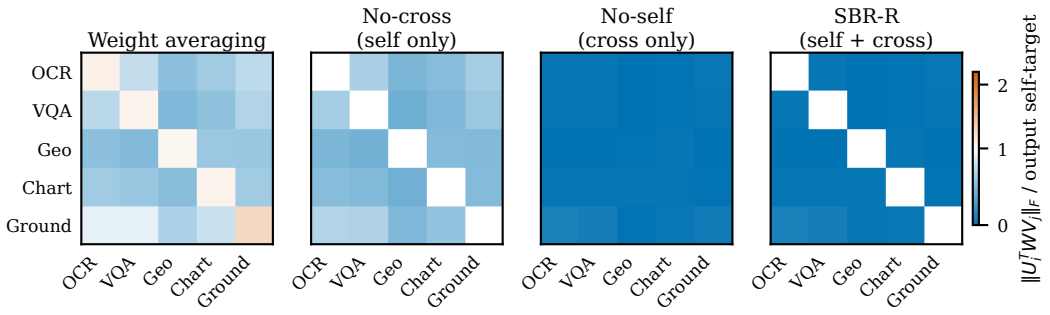


Figure S10: The same four-way projected weight-response ablation on the first language-block `up_proj` map of InternVL2.5-1B (five experts, rank 56). Geo = geometry; Ground = grounding. The plot compares raw task-vector updates before the fixed 0.1 global scale used to build the evaluated checkpoint, matching Fig. 3's intrinsic diagnostic. Off-diagonal magnitude alone does not identify harmful interference.

E.5 Coefficient and Rank Sensitivity

The interval in Eq. (42) is a modeling class, not a consequence of a generative model for downstream activations. We therefore test its two endpoints and a narrower alternative without changing any other setting. In the alternative, the $K-1$ off-task residuals for a uniformly sampled source task have equal second moments and unknown nonnegative exchangeable correlation. The ratio between the energy of their sum and their separate energies then gives the coefficient interval $[1/K, (K-1)/K]$, whose multiplicative center is $\sqrt{K-1}/K$. Table S6 reports all four choices on BERT at the fixed rank ratio 0.35.

Table S6: Task-count coefficient sensitivity for SBR-R on BERT. The average covers AG News, Rotten Tomatoes, and CoLA.

Cross/tail coefficient	Avg.
$1/K$	73.99
$\sqrt{K-1}/K$	76.49
$1/\sqrt{K}$ (reported)	**77.38**
1	74.76

The closest alternative, $\sqrt{K-1}/K$, was also rerun on CLIP ViT-B/32: it obtains 87.3023, versus 87.3297 for $1/\sqrt{K}$. Thus the more specific exchangeable model is not a consistently better replacement. We retain $1/\sqrt{K}$ because it is fixed before evaluation, minimax for the explicitly stated broader class, and is not dominated by the tested alternatives. We do not claim that its ambiguity class is the unique downstream generative model.

For dense-model rank, Figure S11 and Table S7 rerun the final targets and shared solver rather than reusing sweeps from earlier response rules. On the eight-task suite, the fine grid places the SBR optimum at rank 28 (86.5692) and the SBR-R optimum at rank 32 (87.3297). Rank 24 is effectively tied for SBR (-0.0046 points) and is within 0.16 points for SBR-R. The matched solver SBR point at rank 32 is 86.4875, within 0.02 points of the SBR result in Table 3. On the fourteen-task suite, rank 24 is best for both variants; rank 32 remains close for SBR-R but loses 0.99 points for SBR, and rank 48 scores lower for both. Rank is therefore not an inconsequential numerical tolerance, but the useful region is not a single isolated point.

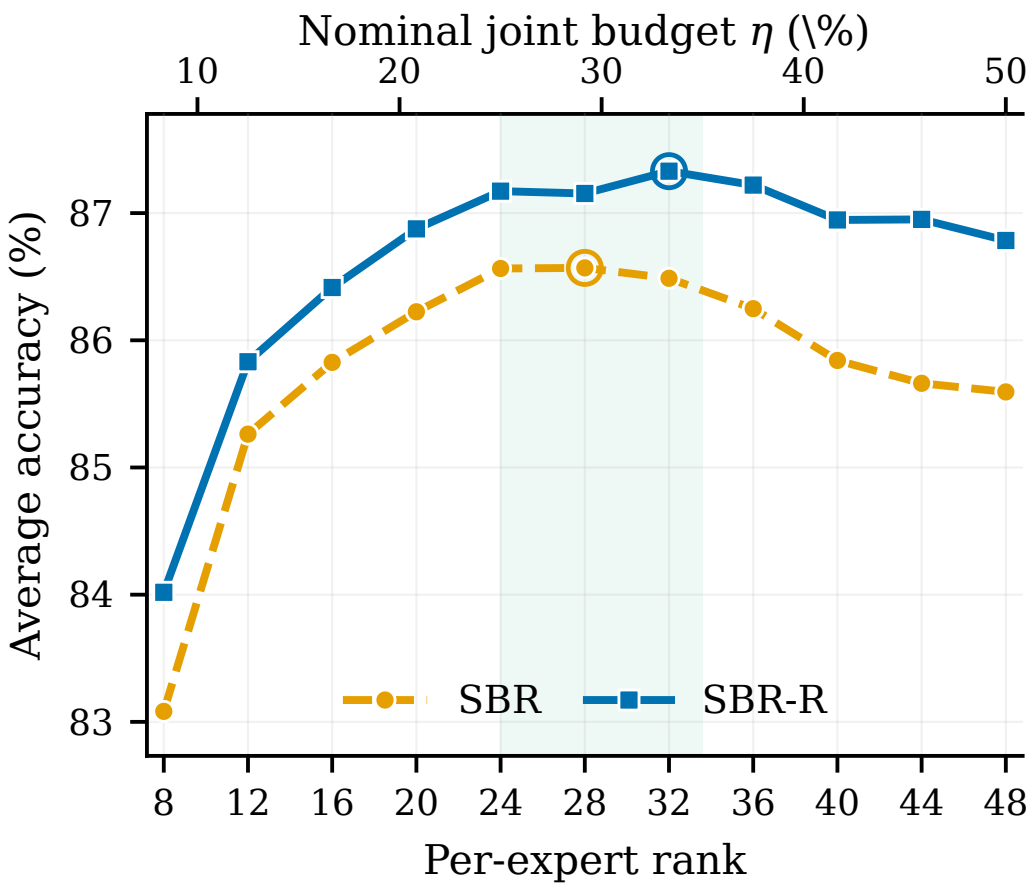


Figure S11: Fine-grained rank sensitivity on the standard eight-task CLIP ViT-B/32 merge, using ranks 8 through 48 in increments of 4 with every other setting fixed. The lower axis gives the per-expert rank, and the upper axis gives the nominal joint-subspace budget $\eta = 8r/768$. The shaded region marks 25–35% nominal capacity; open rings mark the best observed point for each method. SBR peaks at rank 28, whereas SBR-R peaks at rank 32. The sweep uses the complete test sets for post hoc sensitivity analysis and does not retrospectively select the primary results.

Table S7: Rank sensitivity on the fourteen-task ViT-B/32 suite with every other setting fixed. For the 768-dimensional core layers, $\eta = \min(1, Kr/768)$ is the nominal joint-subspace budget before accounting for overlap among expert subspaces. Best and second-best results per method are in **bold** and underlined.

Rank	η (%)	SBR	SBR-R
16	29.2	77.9350	78.1928
17	31.0	<u>78.4597</u>	78.7697
24	43.8	**78.7068**	**79.5902**
32	58.3	77.7154	<u>79.4148</u>
48	87.5	75.7718	76.9787

These test-set sensitivity sweeps do not select the reported ranks. Smaller ranks discard weak spectral tails, keep a larger complementary subspace, and reduce the response-system cost; excessively small ranks can underfit expert behavior, whereas near-saturated joint spaces lose this regularization. In the eight-task fine sweep, the best nominal budgets are 29.2% for SBR and 33.3% for SBR-R; on fourteen tasks, both peak at 43.8%. The useful rank budget therefore varies with task count.

The complementary BERT ratio sweep keeps all other settings fixed. At ratios 0.20, 0.30, 0.35, 0.40, and 0.50, the average accuracies are 74.25, 74.06, $\mathbf{77.38}$, 76.56, and 66.43, respectively. Thus the reported 0.35 lies in a useful neighborhood with 0.40, but the full range is not flat.

We also evaluate a dense-rank rule without an energy threshold. For a task update, stable rank $\mathrm{sr}(\mathbf{W}) = \|\mathbf{W}\|_F^2/\|\mathbf{W}\|_2^2$ is unitarily invariant, scale invariant, continuous in the spectrum, and equals algebraic rank for a flat nonzero spectrum. We set each layer's continuous requirement to either $\max_i \lceil \mathrm{sr}(\mathbf{W}_{i\ell}) \rceil$ or $\lceil \mathrm{sr}(\sum_i \mathbf{W}_{i\ell}) \rceil$, then choose the shared integer r minimizing the squared discrepancy to these requirements subject to each layer's matrix dimension. This rule uses neither labels nor a retained-energy cutoff.

The max-expert rule recovers $r = 32$ on eight-task ViT-B/32 and selects $r = 11$ on BERT. The latter improves the BERT average to 78.30. Applied independently per layer, the same rule lowers ViT-B/32 to 86.61. On the fourteen-task suite, the architecture-wide rule selects $r = 36$ and scores 79.08, below the reported $r = 24$ result of 79.59. It predicts $r = 57$ for InternVL, close to the reported 56, but its lower ViT-B/32 accuracy shows that it does not consistently match the fixed-rank configuration. The aggregate-update rule selects 15 and 19 on the eight- and fourteen-task CLIP suites and also selects ranks below the best observed values. Neither rule consistently matches the fixed-rank results, so dense rank remains a protocol configuration. For Qwen, the output rank is set to the exact upper bound $\mathrm{rank}(\sum_i \mathbf{B}_i\mathbf{A}_i) \le \sum_i \mathrm{rank}(\mathbf{B}_i\mathbf{A}_i)$.

F Extended Benchmark Results

Standalone SBR-R exceeds every listed data-free baseline on all three CLIP backbones, reaching 87.33, 90.80, and 93.71. On Qwen-14B LoRA it reaches 60.12, above SWUDI at 59.78. On InternVL2.5-1B, the complete ten-task averages are 59.61 for SBR and 59.80 for SBR-R, compared with 59.50 for the strongest listed baseline, Iso-CTS. SBR-R also improves over the unmerged InternVL2.5-1B model by 5.51 points. BERT and the fourteen-expert CLIP protocol give the same ranking. All reported SBR-R results are standalone, without SVC or other post-hoc value scaling.

F.1 Full Eight-Task CLIP Results

Tables S8, S9, and S10 report complete per-dataset accuracy. Published baseline rows are reproduced from the identical 8-task CLIP protocol in Li et al. (2026b). WUDI, SWUDI/SWUDI-A, SBR, and SBR-R are evaluated with the matched local pipeline. The SBR values are reused from the original experiments. The ViT-B/32 SBR-R row uses the same shared-solver 50-step run as the solver-depth and component ablations.

Table S8: Full per-dataset top-1 accuracy (%) on ViT-B/32. Individual experts use the corresponding task expert in each column. Best and second-best merged-model results are in **bold** and underlined.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	Avg.
Individual experts	79.23	77.68	96.11	99.78	97.46	98.73	99.69	79.41	91.01
TA	55.10	54.90	66.70	77.20	80.20	69.70	97.30	50.10	68.90
DARE	64.80	63.50	71.80	72.40	63.80	52.40	87.50	50.50	65.80
TIES	59.60	58.60	71.00	81.30	86.10	70.90	98.40	54.70	72.60
TSV-M	69.10	70.70	85.50	94.30	92.00	91.90	99.30	69.20	84.00
Iso-C	**74.80**	74.10	87.90	92.90	83.10	86.00	98.20	67.90	83.10
Iso-CTS	74.40	**74.40**	87.20	90.40	76.80	83.30	97.40	67.00	81.40
STAR	55.90	55.10	67.40	77.70	80.20	68.10	97.20	50.10	69.00
CAT	68.10	65.40	80.50	89.50	85.50	78.50	98.60	60.70	78.40
WUDI	70.82	71.16	85.32	96.00	94.27	94.51	99.47	69.31	85.11
SWUDI	72.31	71.19	87.78	97.44	94.16	94.46	99.50	71.33	86.02
SWUDI-A	69.99	72.81	87.22	95.70	**94.36**	95.00	99.30	69.84	85.53
SBR	71.90	69.80	91.48	98.00	90.95	96.37	99.46	73.83	86.47
SBR-R	72.45	71.51	**91.63**	**98.22**	94.05	**96.88**	**99.58**	**74.31**	**87.33**

Table S9: Full per-dataset top-1 accuracy (%) on ViT-B/16. Best and second-best results are in **bold** and underlined.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	Avg.
TA	61.20	66.00	74.50	74.40	88.10	73.90	98.50	52.70	73.70
DARE	67.60	70.00	76.00	78.60	75.30	59.80	94.40	50.10	71.50
TIES	66.40	70.50	79.80	80.40	89.90	70.30	98.80	57.10	76.60
TSV-M	72.80	80.30	89.10	96.60	93.90	94.00	99.30	72.70	87.30
Iso-C	**78.10**	82.30	91.90	96.90	88.30	91.80	98.80	71.90	87.50
Iso-CTS	77.90	**83.20**	92.00	96.40	84.90	91.30	98.40	71.10	86.90
STAR	63.20	66.30	73.70	79.00	85.60	76.40	98.40	51.80	74.30
CAT	72.90	75.90	83.10	92.80	88.20	82.70	98.80	62.70	82.10
WUDI	75.30	81.67	90.13	98.00	95.29	96.55	99.44	73.62	88.75
SWUDI	76.59	81.93	91.48	98.41	95.59	96.73	99.46	76.33	89.56
SWUDI-A	75.91	81.97	92.81	97.78	95.84	96.86	99.34	75.43	89.49
SBR	77.09	81.86	93.84	**98.96**	95.59	97.63	99.47	79.41	90.48
SBR-R	77.28	82.68	**93.86**	98.70	**96.54**	**98.22**	**99.50**	**79.63**	**90.80**

Table S10: Full per-dataset top-1 accuracy (%) on ViT-L/14. Best and second-best results are in **bold** and underlined.

Method	SUN397	Cars	RESISC45	EuroSAT	SVHN	GTSRB	MNIST	DTD	Avg.
TA	73.90	82.10	86.70	92.70	87.90	86.80	98.90	65.60	84.30
DARE	71.10	81.60	82.60	90.60	78.30	70.80	97.00	63.10	79.40
TIES	76.40	84.20	88.90	95.20	90.00	83.00	99.00	67.90	85.60
TSV-M	79.00	89.80	94.00	98.80	95.30	96.20	99.50	79.10	91.50
Iso-C	81.90	90.90	94.80	98.70	91.40	95.50	99.20	79.20	91.50
Iso-CTS	81.30	91.20	94.70	98.60	89.30	95.30	99.20	77.90	90.90
STAR	74.50	82.00	86.70	93.10	87.80	87.30	98.80	65.00	84.40
CAT	78.70	88.50	91.10	96.30	91.30	95.70	99.40	75.70	89.60
WUDI	81.09	91.01	94.00	99.19	96.21	98.04	99.63	81.17	92.54
SWUDI	81.79	90.96	94.56	99.41	95.84	98.18	99.63	81.60	92.75
SWUDI-A	80.59	91.06	94.40	98.48	97.07	98.25	99.39	80.90	92.52
SBR	83.10	**91.52**	**96.14**	**99.56**	97.42	98.70	99.72	83.46	93.70
SBR-R	**83.15**	91.47	95.94	**99.56**	**97.58**	**98.77**	**99.73**	**83.51**	**93.71**

F.2 Fourteen-Task CLIP Scalability

To test scaling beyond the standard eight-task protocol, we add six more fine-tuned CLIP ViT-B/32 experts (Caltech-101 (Fei-Fei et al., 2004), CIFAR-100 (Krizhevsky, 2009), FGVC-Aircraft (Maji et al., 2013), Flowers-102 (Nilsback & Zisserman, 2008), Food-101 (Bossard et al., 2014), and Oxford-IIIT Pets (Parkhi et al., 2012)) to the original eight, forming a 14-task merging problem with the same pretrained checkpoint. Table S11 reports the complete results: under the matched solver, SBR reaches 78.71%, while SBR-R reaches 79.59%. Both exceed the strongest listed external baseline. We use the fixed configuration specified in Appendix C.1.

Table S11: Full per-dataset top-1 accuracy (%) on 14 CLIP ViT-B/32 tasks. Abbreviations: Cal=Caltech-101, C100=CIFAR-100, FGVC=FGVC-Aircraft, Flow=Flowers-102, GTS=GTSRB. Best and second-best averages are in **bold** and underlined.

Method	SUN	Cars	RES	Euro	SVHN	GTS	MNIST	DTD	Cal	C100	FGVC	Flow	Food	Pets	Avg.
WA	63.5	62.4	67.0	64.9	51.6	44.0	77.0	46.8	91.4	72.0	18.6	73.5	79.0	85.1	64.05
TIES	59.0	55.9	63.5	70.6	55.1	53.0	87.7	47.0	86.6	69.7	30.0	49.0	65.1	69.8	61.58
DARE	63.6	62.4	66.9	65.0	51.9	44.1	76.8	47.2	91.5	71.8	18.3	73.5	78.9	84.8	64.04
TSV-M	64.1	67.1	81.1	94.6	86.9	88.4	99.0	65.6	90.6	74.2	37.1	59.8	73.9	86.7	76.35
Iso-CTS	68.7	68.7	82.8	89.7	82.4	85.8	98.5	64.9	92.7	77.1	39.2	64.7	75.5	83.4	76.72
WUDI	56.8	61.2	73.7	93.4	88.5	88.0	99.1	61.6	85.9	69.4	38.7	53.9	68.9	83.4	73.05
SWUDI	67.9	68.5	83.5	95.6	88.7	89.4	99.1	67.5	90.6	73.9	33.2	66.7	78.6	87.8	77.92
SBR	68.0	64.8	85.9	96.7	85.0	90.3	99.3	69.7	90.3	78.8	35.4	73.5	75.5	88.6	78.71
SBR-R	68.1	66.7	87.2	96.8	92.2	91.4	99.5	70.4	90.5	79.0	37.1	71.6	75.4	88.3	**79.59**

F.3 BERT Merging Results

This experiment merges three BERT-base experts trained on AG News, Rotten Tomatoes, and CoLA, then evaluates the merged classifier on the same three tasks. Checkpoint, training, split, head, and metric details are given in Appendix C.5. SBR-R obtains the best average accuracy. In a component ablation, replacing the Generalist solve by the raw additive tail leaves every retained Specialists response block unchanged, yet lowers the average to 73.8. Complementary-input calibration contributes to the observed BERT result.

Table S12: BERT-base merging accuracy (%). Avg. is the unweighted average over AG News, Rotten Tomatoes, and CoLA. Best and second-best results are in **bold** and underlined.

Method	AG	RT	CoLA	Avg.
WA	95.7	63.8	65.8	75.1
TA	93.4	59.9	66.5	73.3
TIES	98.9	56.1	61.6	72.2
DARE	93.3	52.8	65.5	70.5
TSV-M	44.4	58.5	33.7	45.5
Iso-C	51.3	50.0	69.1	56.8
Iso-CTS	50.5	50.0	69.1	56.5
SWUDI	**99.2**	**66.6**	64.0	76.6
SBR	97.6	57.9	53.9	69.8
SBR-R	98.6	62.8	**70.8**	**77.4**