RESIDUAL GEOMETRY MERGE: STATE-WISE MERGING OF EXPERT LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Model merging combines specialized checkpoints into a single model with multiple capabilities. For large language models, this offers a practical way to integrate expert checkpoints that share a common base without additional training or routing. Existing methods typically combine expert updates directly or transform them before recombination. However, a transformer block is not merely a collection of independent matrices: its projections jointly read from and write to shared residual-stream states. We propose Residual Geometry Merge (RGM), an architecture-aware model merging framework that preserves residual-stream geometry induced by expert checkpoints. Starting from the Transformer Circuits view, RGM uses weight Gram shifts to summarize how each expert changes directions at residual-stream states, aggregates incident shifts at each state, and solves state-wise self-retention equations to derive merge coefficients. The method is coefficient-adaptive and distinct from uniform task-vector addition, routing, and subspace-based merging. Experiments across three shared-base LLM merging settings—Qwen2.5-7B-Instruct with coding, tool-use, and memory experts; DeepSeek-R1-Distill-Qwen-1.5B with coding and mathematics experts; and OpenMath-Nemotron-1.5B with two mathematics experts—show that RGM achieves the strongest overall performance in all three settings.

1 INTRODUCTION

Model merging provides a training-free way to consolidate specialized language-model checkpoints that share a common base. This setting is increasingly relevant to post-training practice: descendants of the same base model may be optimized for coding, tool use, long-context reasoning, mathematical reasoning, search, or other behaviors, while deployed systems often require a single checkpoint rather than a collection of routed experts. Recent industrial post-training pipelines similarly emphasize the importance of integrating multiple specialized experts into one model through multi-teacher on-policy distillation (Xiaomi LLM-Core Team, 2026; DeepSeek-AI, 2026; Kimi Team, 2026). Model merging studies the more constrained version of this consolidation problem, where the available inputs are only the trained checkpoints, without inference-time routing or additional training.

Existing shared-base merging methods primarily decide how to combine expert task vectors in parameter space. Given a base checkpoint θ_0 and expert checkpoints θ_i, they form $\Delta_i = \theta_i - \theta_0$ and combine these deltas through averaging, summation, pruning, sign resolution, random dropping, or other parameter-space transformations (Wortsman et al., 2022; Ilharco et al., 2023; Yadav et al., 2023; Yu et al., 2024). For reinforced experts, this view is especially important because recent work suggests that RL-induced task vectors can be sparse and mutually compatible, making direct summation a strong baseline in some agentic merging settings (Zhu et al., 2026b; Yuan et al., 2026). Another line of work uses SVD to decompose expert updates into singular components, then aligns, selects, or reweights these components before recombination (Stoica et al., 2025; Gargiulo et al., 2025; Skorobogat et al., 2025; Zhang et al., 2026; Zhu et al., 2026a). These approaches have advanced checkpoint merging, but they do not explicitly account for how multiple projections jointly read from or write to the same residual-stream state.

The residual stream offers a different unit of comparison for transformer checkpoint merging. The residual stream is the shared communication channel through which attention and MLP components

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.

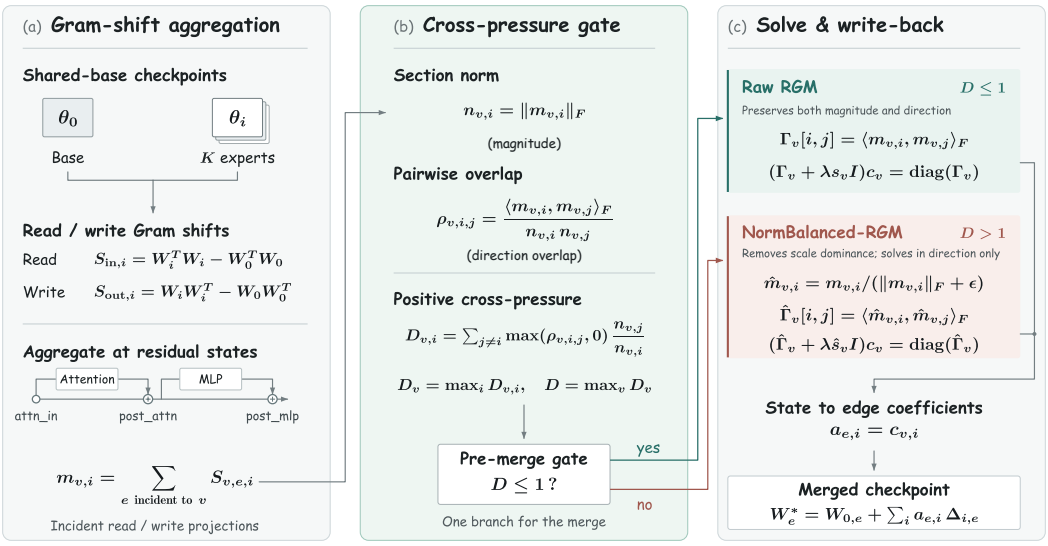

Figure 1: Overview of the Residual Geometry Merge (RGM) Framework.

exchange information (Elhage et al., 2021). Within one transformer block, query, key, and value projections read the pre-attention residual state; the attention output projection writes back to the stream; the MLP input projections read the post-attention state; and the down projection writes the MLP update. Consequently, expert effects should be compared at residual-stream states, where several projections jointly read from or write to the same representation. These states provide the natural units at which the effects of incident projections can be aggregated and compared.

We propose Residual Geometry Merge (RGM), an architecture-aware merging framework that derives state-wise merge coefficients from expert-induced Gram shifts aggregated at residual-stream states. These checkpoint-only quantities encode the geometry: how each expert changes inner products among residual directions. For projections that read a residual state, RGM represents expert i by the weight Gram shift $W_i^T W_i - W_0^T W_0$; for projections that write to a residual state, it uses $W_i W_i^T - W_0 W_0^T$. RGM aggregates the shifts of projections incident to each residual state and solves a self-retention equation, whose target is to preserve each expert's own state-level effect while reducing duplicate retention along overlapping directions. Finally, we apply the resulting state-wise coefficients to the original task-vector deltas for incident projections. Figure 1 gives an overview of the three-stage procedure.

The full RGM procedure contains a pre-specified choice between raw and scale-normalized coefficient estimation. Raw RGM solves self-retention with the unnormalized residual-state Gram shifts, preserving both direction and magnitude. NormBalanced-RGM solves the same equation after normalizing each expert's state effect by its Frobenius norm, preserving overlap direction while removing scale dominance from the coefficient solve. The pre-merge branch-selection rule makes this choice before downstream evaluation using a cross-pressure diagnostic derived from the raw self-retention equation itself. Raw RGM is used when each expert's self-retention row remains dominated by its own effect; NormBalanced-RGM is used when positive overlap amplified by relative effect norms can dominate that row.

We evaluate RGM across three shared-base LLM merging settings: one three-expert 7B merge and two two-expert 1.5B merges. These settings probe complementary interaction regimes: low-overlap heterogeneous experts, high-overlap cross-domain experts, and high-overlap same-domain experts. On a Qwen2.5-7B-Instruct setting that merges RL experts for coding, tool use, and long-context memory, expert-induced residual-state Gram shifts are nearly orthogonal, with negligible pairwise overlap. The branch-selection rule selects Raw RGM, which achieves the best overall average, 63.95, compared with 63.90 for the strongest baselines and 62.09 for full task-vector summation. The 7B ablations further show that the self-retention objective is a strong signal and that the residual-state Gram construction gives the best overall balance among the tested variants.

The two 1.5B settings test the scale-normalized branch of the same method. In the DeepSeek-R1-Distill-Qwen-1.5B setting with code and math experts, 30 of 84 residual nodes exceed the unit cross-pressure threshold, so the branch-selection rule selects NormBalanced-RGM without using evaluation feedback. The selected variant reaches the best overall average, 45.48, compared with 45.45 for the best competing merge and 43.93 for Raw RGM. In the OpenMath-Nemotron-1.5B setting with two math experts, 70 of 84 residual nodes exceed the unit cross-pressure threshold, and the branch-selection rule again selects NormBalanced-RGM. The selected variant reaches 67.07 overall, compared with 66.34 for the best competing merge and 65.14 for full task-vector summation. Together, the three settings support the central claim of this paper: expert merging should be decided at residual-stream states, and coefficient estimation should account for whether raw Gram-shift magnitude is informative or scale-dominating.

Our contributions are:

- We formulate model merging at transformer residual-stream states, where projections jointly read from and write to the network's shared communication channel.
- We represent expert effects by read/write weight Gram shifts and derive a state-wise self-retention equation for merge coefficients.
- We introduce a pre-merge branch-selection rule that chooses Raw RGM or NormBalanced-RGM using a cross-pressure diagnostic computed from residual-state Gram-shift sections.
- We show that RGM achieves the best result on a 7B heterogeneous RL-expert benchmark and that the same RGM pipeline is effective in two shared-base 1.5B RL-expert settings, where the branch-selection rule selects NormBalanced-RGM based on the cross-pressure diagnostic.

2 Related Work

2.1 Task-vector merging

Weight-space averaging and model soups established that independently fine-tuned models can often be combined without increasing inference cost (Wortsman et al., 2022). Task Arithmetic made the expert-base difference $\Delta_i = \theta_i - \theta_0$ the central object for composition (Ilharco et al., 2023). Many later methods modify task vectors before addition. TIES removes small updates and resolves sign conflicts (Yadav et al., 2023); DARE randomly drops delta parameters and rescales survivors (Yu et al., 2024); DELLA, Model Breadcrumbs, and EMR-Merging refine sparsity, masking, or interference control (Deep et al., 2024; Davari & Belilovsky, 2024; Huang et al., 2024). Behavior-knowledge merging in reinforced agentic models and recent analyses of RL coexistence further suggest that RL-induced task vectors may be more compatible than supervised fine-tuning updates (Yuan et al., 2026; Zhu et al., 2026b).

RGM keeps the final realization as a weighted sum of task vectors, but its coefficients are not determined independently at individual parameter coordinates. It derives coefficients from the geometry that each expert induces on residual-stream states. This state-level coefficient field allows RGM to behave close to task-vector summation in the low-overlap 7B setting while still applying state-wise corrections.

2.2 Subspace and spectral merging

A complementary family of methods combines expert updates after projecting them into subspaces or decomposing them into singular components. KnOTS and related methods align or combine update subspaces (Stoica et al., 2025; Skorobogat et al., 2025). Task Singular Vectors and spectral rewiring methods use singular directions to preserve or filter task-relevant components; related RLVR-oriented work also studies spectral inheritance (Gargiulo et al., 2025; Zhang et al., 2026; Zhu et al., 2026a).

RGM takes a different route: it uses residual-stream states as the units for comparing expert effects. It groups projections by the residual-stream states they read from or write to, then solves for expert coefficients at those states.

2.3 Residual streams and weight Gram matrices

Transformer Circuits provides the architectural language used by RGM: the residual stream is the shared channel through which attention and MLP components communicate (Elhage et al., 2021). RGM uses this read/write structure to decide where expert effects should be compared.

Weight Gram matrices are also natural objects in feature-learning analyses. Radhakrishnan et al. (2024) relate layer weight Gram matrices to feature learning, and Cha et al. (2026) study weight-Gram shifts during training. RGM brings the same type of object into data-free model merging. The Gram shift is not merely a post-training analysis tool; it is the representation from which merge coefficients are derived.

3 Residual Geometry Merge

RGM is a three-stage data-free merging procedure. First, RGM converts each expert update into read/write Gram shifts and aggregates the shifts incident to each residual state. Second, it computes the cross-pressure diagnostic from these Gram-shift sections and applies the pre-merge branch-selection rule to select the Raw RGM or NormBalanced-RGM coefficient-estimation branch. Third, it solves the selected state-wise self-retention equations and transfers the resulting coefficients back to the original task-vector deltas. The subsections below describe these stages in execution order.

3.1 Setup

We consider K expert checkpoints $\theta_1, \ldots, \theta_K$ that share a base checkpoint θ_0. For every modeled projection edge e, write:

$$W_{i,e} = W_{0,e} + \Delta_{i,e}. \tag{1}$$

The final checkpoint is realized as a weighted task-vector merge:

$$W_e^* = W_{0,e} + \sum_{i=1}^{K} a_{e,i}\, \Delta_{i,e}. \tag{2}$$

The question is how to choose $a_{e,i}$. RGM derives coefficients at residual-stream states and transfers them back to projection edges. This residual-state view motivates the use of Gram shifts: incident read/write projections may have different parameter shapes, but their induced Gram shifts are residual-state square matrices and can therefore be aggregated before coefficient solving.

For each transformer layer, RGM considers three residual states. The pre-attention state $\mathrm{attn_in}$ is read by W_q, W_k, and W_v. The post-attention state $\mathrm{post_attn}$ receives the attention output through W_o and is read by W_{gate} and W_{up}. The post-MLP state $\mathrm{post_mlp}$ receives the MLP output through W_{down}. This assignment follows the block computation graph rather than parameter names alone.

3.2 Residual-state Gram shifts

For a projection $W \in \mathbb{R}^{d_{\mathrm{out}} \times d_{\mathrm{in}}}$ that reads from a residual state, the input-side Gram matrix $W^T W$ determines inner products after reading residual directions:

$$\langle Wx,\, Wy \rangle = x^T W^T W y. \tag{3}$$

The expert-induced read shift is:

$$S_{\mathrm{in},\,i}(W) = W_i^T W_i - W_0^T W_0 = W_0^T \Delta_i + \Delta_i^T W_0 + \Delta_i^T \Delta_i. \tag{4}$$

For a projection that writes into the residual stream, the corresponding residual-state Gram shift is:

$$S_{\mathrm{out},\,i}(W) = W_i W_i^T - W_0 W_0^T = W_0 \Delta_i^T + \Delta_i W_0^T + \Delta_i \Delta_i^T. \tag{5}$$

At each residual state v, RGM adds the shifts of incident projections:

$$m_{v,i} = \sum_{e \text{ incident to } v} S_{v,e,i}. \tag{6}$$

This sum is principled. If multiple projections read the same residual vector x, their joint read map is:

$$J_v = \bigoplus_{e=1}^{m} W_e, \tag{7}$$

so that:

$$\|J_v x\|^2 = \sum_{e=1}^{m} \|W_e x\|^2 = x^T \Big(\sum_{e=1}^{m} W_e^T W_e \Big) x. \tag{8}$$

Thus the Gram matrix of the joint read map is the sum of the incident Gram matrices. For the resulting state sections, define the expert Gram matrix:

$$\Gamma_v[i,j] = \langle m_{v,i},\, m_{v,j} \rangle_F. \tag{9}$$

The section $m_{v,i}$ is the residual-state representation of expert i, and Γ_v records the pairwise Frobenius inner products among experts at that state. These two objects are passed to the pre-merge cross-pressure gate and the subsequent coefficient solve.

3.3 Pre-merge branch selection

Before solving for merge coefficients, RGM chooses whether self-retention should be estimated from the raw sections $m_{v,i}$ or from norm-balanced sections. The raw branch preserves both the magnitude and direction of each Gram-shift section, while the norm-balanced branch removes relative section scale from the coefficient solve. The choice is made by checking whether positive cross terms in the raw system can dominate an expert's self-retention row. For residual node v, define the section norm and cosine overlap:

$$n_{v,i} = \|m_{v,i}\|_F, \quad \rho_{v,i,j} = \frac{\langle m_{v,i},\, m_{v,j} \rangle_F}{n_{v,i}\, n_{v,j}}. \tag{10}$$

These quantities summarize the scale and directional overlap of the Gram-shift sections used by the cross-pressure diagnostic.

The candidate raw self-retention solve has the unregularized form:

$$\Gamma_v c_v = \mathrm{diag}(\Gamma_v). \tag{11}$$

Dividing its i-th row by $n_{v,i}^2$ gives:

$$c_{v,i} + \sum_{j \neq i} \rho_{v,i,j} \frac{n_{v,j}}{n_{v,i}}\, c_{v,j} = 1. \tag{12}$$

The self term has coefficient one, while each cross term is weighted by the overlap sign and the relative section norm. In particular, positive overlap from a larger section can consume the self-retention budget of a smaller section before the smaller expert's own coefficient is considered. We therefore define the positive cross-pressure on row i as:

$$D_{v,i} = \sum_{j \neq i} \max(\rho_{v,i,j},\, 0)\, \frac{n_{v,j}}{n_{v,i}}, \quad D_v = \max_i D_{v,i}. \tag{13}$$

The pre-merge cross-pressure gate is:

$$\max_v D_v \leq 1 \quad \Rightarrow \quad \text{use raw RGM}, \tag{14}$$

$$\max_v D_v > 1 \quad \Rightarrow \quad \text{use NormBalanced-RGM}. \tag{15}$$

The boundary at one is not tuned on downstream tasks. It is the point at which positive cross-pressure can exceed the unit self-retention budget in the raw equation. For two experts with norms n_1, n_2 and cosine ρ, the unregularized solution is:

$$\begin{cases} c_1 = \dfrac{1 - \rho\, n_2/n_1}{1 - \rho^2}, \\ c_2 = \dfrac{1 - \rho\, n_1/n_2}{1 - \rho^2}. \end{cases} \tag{16}$$

Thus $\rho\, n_2/n_1 > 1$ makes c_1 non-positive, and symmetrically $\rho\, n_1/n_2 > 1$ makes c_2 non-positive. The pre-merge cross-pressure gate therefore selects NormBalanced-RGM only when $\max_v D_v > 1$.

3.4 RAW SELF-RETENTION

When $\max_v D_v \leq 1$, the pre-merge cross-pressure gate selects the raw self-retention solve. RGM represents the merged state section as:

$$m_v^* = \sum_i c_{v,i}\, m_{v,i}. \tag{17}$$

The self-retention of expert i at state v is:

$$\mathrm{ret}_i(v) = \frac{\langle m_v^*,\, m_{v,i}\rangle_F}{\|m_{v,i}\|_F^2} = \frac{(\Gamma_v c_v)_i}{\Gamma_v[i,i]}. \tag{18}$$

Raw RGM sets this quantity to one for every expert:

$$\Gamma_v c_v = \mathrm{diag}(\Gamma_v). \tag{19}$$

In practice we solve:

$$(\Gamma_v + \lambda s_v I)\, c_v = \mathrm{diag}(\Gamma_v), \tag{20}$$

where s_v is the mean positive diagonal energy. Our experiments use $\lambda = 10^{-3}$.

The equation has the desired limiting behavior. If experts are orthogonal at a state, then Γ_v is diagonal and $c_v \approx (1, \ldots, 1)$, so non-overlapping effects are retained rather than averaged away. If two experts induce the same section, the off-diagonal entries prevent duplicated retention of the shared direction.

3.5 NORM-BALANCED VARIANT

When $\max_v D_v > 1$, the pre-merge cross-pressure gate selects the norm-balanced self-retention solve. Raw RGM trusts both the direction and the magnitude of each state section $m_{v,i}$. This is appropriate when Gram-shift magnitude is a meaningful part of the residual effect, but it can be harmful when magnitude acts as a confounding scale factor. In that regime, a larger section with positive overlap can dominate the self-retention solve and suppress the corresponding coefficient of a smaller section.

NormBalanced-RGM keeps the same retention principle but solves it on normalized sections:

$$\hat{m}_{v,i} = \frac{m_{v,i}}{\|m_{v,i}\|_F + \epsilon}. \tag{21}$$

It then forms:

$$\hat{\Gamma}_v[i,j] = \langle \hat{m}_{v,i},\, \hat{m}_{v,j}\rangle_F \tag{22}$$

and solves:

$$(\hat{\Gamma}_v + \lambda \hat{s}_v I)\, c_v = \mathrm{diag}(\hat{\Gamma}_v). \tag{23}$$

The resulting coefficients are still applied to the original task-vector deltas in weight space. Thus NormBalanced-RGM differs from raw RGM only by normalizing the state sections before the coefficient solve; the downstream steps remain the same. It preserves overlap direction and removes raw scale domination.

For either branch, the solved state-wise coefficients are transferred to the projection edges incident to the corresponding residual state:

$$a_{e,i} = c_{v,i} \qquad \text{(e incident to } v\text{)} \tag{24}$$

Substituting these coefficients into the weighted task-vector merge in Eq. (2) produces the final merged checkpoint.

4 EXPERIMENTS

We evaluate RGM in three shared-base checkpoint-merging regimes that stress different forms of expert interaction. The Qwen2.5-7B-Instruct setting tests heterogeneous RL experts with low residual-state overlap, the DeepSeek-R1-Distill-Qwen-1.5B setting tests a higher-overlap two-expert merge, and the OpenMath-Nemotron-1.5B setting isolates a same-domain merge. Additional evaluation details are provided in Appendix A.

Table 1 | **Main results on Qwen2.5-7B-Instruct with 3 RL experts.** Entries are percentages. Coding reports pass accuracy (ACC) and unit-test accuracy (UT); tool use follows BFCL v4; memory uses RULER. Best result per column is in **bold**. Shading marks Experts and Ours.

Method	Coding				Tool Use		Memory				Avg
	LB		LCB v2		Live	Non-Live	HotpotQA		SQuAD		
	ACC	UT	ACC	UT	ACC	ACC	32K	64K	32K	64K	
Domain Expert	35.37±0.37	46.76±0.72	29.81±0.11	44.10±0.18	**76.76**	86.92	78.37±0.58	79.69±0.22	79.98±0.11	78.53±0.53	63.63±0.04
TA-avg	36.54±0.30	47.81±0.36	30.43±0.05	45.42±0.12	76.68	87.37	68.60±0.56	68.82±0.60	68.15±0.52	70.23±0.41	60.00±0.12
TA-sum	35.09±0.15	43.89±0.16	26.65±0.59	36.79±0.77	74.61	85.90	78.63±0.28	79.43±0.38	**80.62±0.17**	79.26±0.54	62.09±0.13
RAM	37.67±1.67	49.20±2.00	30.81±0.67	45.78±0.97	70.83	88.00	78.38±0.27	79.26±0.40	80.04±0.16	79.03±0.26	63.90±0.48
TIES	37.45±2.23	**49.78±1.32**	31.27±0.21	**46.11±0.64**	67.71	88.00	78.63±0.39	79.56±0.24	80.34±0.41	79.12±0.50	63.80±0.13
TSVM	37.57±1.31	49.48±0.71	**31.32±0.73**	45.92±0.47	67.71	**89.00**	77.46±0.43	78.65±0.50	78.29±0.35	77.46±0.32	63.28±0.27
ISO-Merger	37.79±0.18	48.32±0.46	29.91±0.32	42.61±0.37	75.80	86.63	**78.78±0.55**	79.96±0.86	80.52±0.31	78.71±0.34	63.90±0.18
RGM (Ours)	**37.81±0.08**	48.31±0.08	30.27±0.57	42.75±0.31	74.69	86.15	78.71±0.33	**80.11±0.22**	80.57±0.17	**80.11±0.23**	**63.95±0.10**

Table 2 | **Ablation results for Residual Geometry Merge.** The protocol and metrics match Table 1. PM denotes per-matrix estimation, Node denotes residual-state estimation, Delta denotes raw task-vector deltas, and Gram denotes Gram shifts. Best result per column is in **bold**. Shading marks Ours.

Method	Coding				Tool Use		Memory				Avg
	LB		LCB v2		Live	Non-Live	HotpotQA		SQuAD		
	ACC	UT	ACC	UT	ACC	ACC	32K	64K	32K	64K	
PM-Delta	37.00±0.30	47.48±0.31	29.81±0.06	42.64±0.07	**74.69**	**86.38**	**79.07±0.30**	79.39±0.25	80.78±0.47	79.92±0.16	63.72±0.09
PM-Gram	37.04±0.16	47.49±0.10	30.04±0.41	42.60±0.33	74.61	86.12	78.45±0.31	79.98±0.40	**80.84±0.54**	78.97±0.82	63.61±0.07
Node-Delta	**38.09±0.82**	**48.35±0.91**	29.90±0.50	42.44±0.43	74.46	86.02	78.74±0.43	79.65±0.48	80.24±0.24	79.33±0.37	63.72±0.23
RGM (Ours)	37.81±0.08	48.31±0.08	**30.27±0.57**	**42.75±0.31**	**74.69**	86.15	78.71±0.33	**80.11±0.22**	80.57±0.17	**80.11±0.23**	**63.95±0.10**

4.1 QWEN2.5-7B-INSTRUCT HETEROGENEOUS RL-EXPERT SETTING

Setup. We first consider a heterogeneous merge of coding, tool-use, and memory experts. Their expert-induced residual-state sections have a mean pairwise cosine overlap of -2.97×10^{-4}, almost zero. This setting tests whether RGM can preserve complementary capabilities while adapting the merge to state-specific expert interactions, even when direct task-vector summation is already competitive. Starting from Qwen2.5-7B-Instruct, we merge three RL-specialized descendants: CURE for coding (Wang et al., 2025b), ToolRL for tool use (Qian et al., 2025), and MemAgent for long-context memory (Yu et al., 2026). We compare RGM against the domain experts, task-vector averaging, task-vector summation (Ilharco et al., 2023), RAM (Yuan et al., 2026), TIES (Yadav et al., 2023), TSVM (Gargiulo et al., 2025), and ISO-Merger (Zhu et al., 2026a).

Results. Table 1 shows that RGM achieves the best overall average in this setting, reaching 63.95, versus 63.90 for the two strongest baselines and 62.09 for TA-sum. The margin over the strongest baselines is small, but the comparison is informative: RGM remains competitive in a low-overlap regime while applying residual-state corrections that a single global coefficient rule cannot express. Table 2 isolates two design choices: the unit of coefficient estimation (per-matrix versus residual-state) and the representation of expert effects (raw task-vector deltas versus Gram shifts). All three ablations remain competitive, while the full residual-state Gram construction achieves the highest overall average among the variants. Appendix B gives the exact construction of each variant.

4.2 DEEPSEEK-R1-DISTILL-QWEN-1.5B WITH TWO RL EXPERTS

Setup and cross-pressure diagnostic. We next consider a cross-domain two-expert merge with substantial positive cross-pressure. This setting tests whether the branch-selection rule identifies when raw Gram-shift magnitudes distort coefficient estimation and selects NormBalanced-RGM before evaluation. The shared base is DeepSeek-R1-Distill-Qwen-1.5B, and the experts are Archer2.0 for coding (Wang et al., 2025a) and JustRL for mathematics (He et al., 2025). Before reporting downstream accuracy, we apply the cross-pressure diagnostic to all three evaluation settings.

Table 3 separates the low- and high-overlap regimes before any evaluation is run. In the Qwen2.5-7B-Instruct setting, cross-pressure remains below one at every node, with $\max_v D_v = 0.2048$. In contrast, the DeepSeek-R1-Distill-Qwen-1.5B setting has mean pairwise cosine 0.4546, with cross-

Table 3 | **Pre-merge cross-pressure statistics.** D_v is the largest positive cross-pressure at residual node v. Raw RGM is selected only when $\max_v D_v \leq 1$.

Setting	D_v p95	D_v max	$D_v > 1$ nodes	Decision
Qwen2.5-7B-Instruct	0.0344	0.2048	0/84	Raw RGM
DeepSeek-R1-Distill 1.5B	1.6913	1.8537	30/84	NormBalanced-RGM
OpenMath-Nemotron 1.5B	1.7054	2.6791	70/84	NormBalanced-RGM

Table 4 | **Main results on DeepSeek-R1-Distill-Qwen-1.5B with 2 RL experts.** Entries are percentages. Coding uses LiveBench (LB) and LiveCodeBench v5 (LCB v5), reporting pass accuracy (ACC) and unit-test accuracy (UT). Math uses the sampling counts shown under each benchmark. The Domain Expert row uses Archer2.0 for coding and JustRL for math. Best result per column is in **bold**.

Method	Coding				Math					Avg
	LB		LCB v5		AIME 2024	AIME 2025	AMC 2023	Minerva	OlympiadBench	
	ACC	UT	ACC	UT	Avg@32	Avg@32	Avg@8	Avg@4	Avg@4	
Domain Expert	22.41±0.12	24.99±0.13	25.20±0.08	40.87±0.16	51.18±1.19	**37.08±0.52**	90.73±0.78	41.88±0.61	57.96±0.03	43.59±0.16
TA-sum	20.90±0.16	23.45±0.28	17.30±0.26	30.33±0.32	48.40±0.54	34.03±0.62	88.33±0.82	41.24±0.49	57.85±0.81	40.20±0.35
RAM	**28.35±0.40**	**31.60±0.27**	25.49±0.00	44.36±0.00	52.47±0.34	35.03±1.08	89.58±0.64	42.92±0.33	57.94±0.37	45.31±0.13
TSVM	26.19±0.34	29.68±0.37	25.97±0.35	**44.98±0.32**	51.46±0.45	35.24±0.30	89.69±1.55	42.83±0.98	57.78±0.38	44.87±0.20
TIES	27.80±0.61	30.87±0.30	25.54±0.02	43.88±0.00	53.09±0.79	36.08±0.47	89.27±0.39	**43.81±0.54**	**58.68±0.51**	45.45±0.05
ISO-Merger	27.93±0.08	30.62±0.14	25.30±0.11	44.46±0.04	53.40±0.49	36.84±0.43	89.79±0.39	42.62±0.11	58.12±0.14	45.45±0.23
Raw RGM	26.74±0.44	30.84±0.56	22.04±0.00	35.12±0.00	**53.78±1.63**	35.80±0.69	90.31±1.11	42.31±0.68	58.40±0.45	43.93±0.28
RGM-NormBalanced	26.60±0.07	29.74±0.14	**26.20±0.09**	44.74±0.06	53.12±0.53	36.32±0.76	**90.94±0.28**	43.38±0.13	58.25±0.16	**45.48±0.07**

pressure exceeding one at 30 out of 84 residual nodes. The OpenMath-Nemotron-1.5B setting shows an even stronger pattern, with cross-pressure exceeding one at 70 out of 84 residual nodes.

Results. Table 4 tests this prediction: norm-balanced coefficient estimation is the stronger branch in this higher-overlap regime. The Domain Expert baseline achieves an overall average of 43.59 by using Archer2.0 for coding and JustRL for mathematics, while TA-sum reaches 40.20, indicating that unweighted task-vector summation is insufficient here. Raw RGM improves over TA-sum and reaches 43.93, but remains below the best competing merge at 45.45. After normalizing state-section magnitudes before the coefficient solve, NormBalanced-RGM obtains the best overall result in the table, 45.48. The largest difference between the raw and norm-balanced branches occurs on LiveCodeBench v5, where NormBalanced-RGM improves raw RGM from 22.04 to 26.20 ACC and from 35.12 to 44.74 UT.

These results illustrate the role of the pre-merge branch-selection rule rather than an unconditional preference for normalization. The branch-selection rule selects Raw RGM in the Qwen2.5-7B-Instruct setting and NormBalanced-RGM in both 1.5B settings, with all choices made before evaluation. Thus the same procedure instantiates raw self-retention when $\max_v D_v \leq 1$, and scale-normalized self-retention when positive cross-pressure exceeds that threshold.

4.3 OpenMath-Nemotron-1.5B with two mathematics-specialized RL experts

Setup. We finally consider a same-domain merge, where redundancy rather than cross-domain capability transfer is the central concern. This setting tests whether state-wise aggregation can control duplicated expert effects and whether the same branch-selection rule remains effective when both experts target the same capability. The shared base is OpenMath-Nemotron-1.5B, and the two experts are QuestA-Nemotron-1.5B and JustRL-Nemotron-1.5B, both specialized for mathematical reasoning.

Results. Table 5 tests the same principle under same-domain redundancy: the branch-selection rule selects NormBalanced-RGM based on the cross-pressure diagnostic. NormBalanced-RGM achieves the best overall average, 67.07 ± 0.06, outperforming the best competing merge at 66.34 ± 0.27 and TA-sum at 65.14 ± 0.28. It also exceeds both unmerged experts on the aggregate score, indicating that the merged checkpoint is not simply selecting one specialist. Together with the DeepSeek-R1-Distill-Qwen-1.5B results, this setting supports the use of scale-normalized coefficient estimation when the cross-pressure diagnostic reports substantial cross-pressure.

Table 5 | **Main results on OpenMath-Nemotron-1.5B with 2 math RL experts.** Entries are percentages; expert rows are unmerged checkpoints. Best result per column is in **bold**.

Method	Math					Avg
	AIME 2024	AIME 2025	AMC 2023	Minerva	OlympiadBench	
QuestA-Nemotron	72.08±0.10	60.99±0.68	94.84±0.78	31.57±0.41	72.30±0.44	66.36±0.13
JustRL-Nemotron	68.70±0.05	60.73±0.31	95.62±0.62	32.03±0.69	70.81±0.07	65.58±0.05
TA-sum	70.94±0.52	**61.72±0.16**	93.44±0.62	30.70±0.92	68.91±0.20	65.14±0.28
ISO-Merger	69.64±0.99	61.35±0.31	**96.88±0.94**	32.49±1.15	71.33±0.04	66.34±0.27
RGM-NormBalanced	**72.50±0.42**	61.61±0.36	95.78±0.78	**32.81±0.09**	**72.65±0.39**	**67.07±0.06**

5 Analysis

5.1 Relationship to task-vector summation

If expert effects are non-overlapping, task-vector summation is a natural baseline. RGM agrees with that intuition but formulates it at the residual-state level. When Γ_v is diagonal, the self-retention solution is $c_v = (1, \ldots, 1)$, which is exactly the retention behavior of summation at that residual state. When overlap is present, the same equation reduces duplicate retention. Thus RGM recovers task-vector summation in the orthogonal limit and introduces state-wise coefficient corrections when residual-state Gram sections overlap.

The 7B results fit this reading. TA-sum is strong because the expert-induced residual-state sections are nearly orthogonal on average, but it is still worse than RGM because a single global coefficient pattern cannot respond to state-specific read/write overlap.

5.2 Role of norm balancing

Raw RGM uses $\Gamma_v[i, j] = \langle m_{v,i}, m_{v,j} \rangle_F$. This contains both direction and magnitude. If one expert's effect has larger norm but little positive overlap with the others, the positive cross-pressure $D_{v,i}$ remains below one and the gate continues to select Raw RGM; this is the 7B case. If the larger effect also has positive overlap with a smaller effect, the term $\rho\, n_j / n_i$ enters the normalized raw equation and can make $D_{v,i} > 1$; this is the DeepSeek and OpenMath 1.5B case.

NormBalanced-RGM removes the magnitude ratio from the coefficient solve while preserving the overlap directions. It should therefore be used when the cross-pressure diagnostic indicates that $\max_v D_v > 1$. It should not be the unconditional default, because raw magnitude can be informative when it does not dominate the self-retention row. This motivates making the branch-selection rule part of the method rather than a post hoc adjustment.

6 Conclusion

This paper presents Residual Geometry Merge (RGM), an architecture-aware method for merging expert checkpoints that share a common base. RGM compares expert effects at block-local residual-stream states, represents them with read/write Gram shifts, and solves state-wise self-retention equations whose coefficient-estimation branch is chosen before downstream evaluation. It achieves competitive performance across one 7B merge and two 1.5B merges, which supports residual-stream states as useful units for deciding how expert updates should be combined.

RGM is a heuristic abstraction of expert interactions rather than an exact model of Transformer forward dynamics. At the block-local granularity used here, the state partition follows the within-block structure of a Transformer, treats states across block boundaries as distinct, and abstracts away the post-attention normalization. Under these assumptions, the pre-attention, post-attention, and post-MLP states form the natural three-state partition of a Transformer block. Future work could test a variant that also abstracts away the pre-attention normalization of layer $l + 1$, thereby merging the post-MLP state of layer l with the attention-input state of layer $l + 1$. Alternatively, future work could model these normalization maps explicitly and compare different residual-state partitions. Extending RGM beyond shared-base checkpoints with matched Transformer architectures is another important direction.

REFERENCES

Taehun Cha, Daniel Beaglehole, Adityanarayanan Radhakrishnan, and Donghun Lee. The weight Gram matrix captures sequential feature linearization in deep networks. *arXiv preprint arXiv:2605.06258*, 2026.

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

Pala Tej Deep, Rishabh Bhardwaj, and Soujanya Poria. DELLA-Merging: Reducing interference in model merging through magnitude-based sampling. *arXiv preprint arXiv:2406.11617*, 2024.

DeepSeek-AI. DeepSeek-V4: Towards highly efficient million-token context intelligence. *arXiv preprint arXiv:2606.19348*, 2026.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021.

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 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Bingxiang He, Zekai Qu, Zeyuan Liu, Yinghao Chen, Yuxin Zuo, Cheng Qian, Kaiyan Zhang, Weize Chen, Chaojun Xiao, Ganqu Cui, Ning Ding, and Zhiyuan Liu. JustRL: Scaling a 1.5B LLM with a simple RL recipe. *arXiv preprint arXiv:2512.16649*, 2025.

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*, 2024.

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.

Kimi Team. Kimi K3: Open frontier intelligence. *arXiv preprint arXiv:2607.24653*, 2026.

Jiazheng Li, Hongzhou Lin, Hong Lu, Kaiyue Wen, Zaiwen Yang, Jiaxuan Gao, Yi Wu, and Jingzhao Zhang. QuestA: Expanding reasoning capacity in LLMs via question augmentation. In *International Conference on Learning Representations*, 2026.

Shudong Liu, Hongwei Liu, Junnan Liu, Linchen Xiao, Songyang Gao, Chengqi Lyu, Yuzhe Gu, Wenwei Zhang, Derek F. Wong, Songyang Zhang, and Kai Chen. CompassVerifier: A unified and robust verifier for LLMs evaluation and outcome reward. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025.

Ivan Moshkov, Darragh Hanley, Ivan Sorokin, Shubham Toshniwal, Christof Henkel, Benedikt Schifferer, Wei Du, and Igor Gitman. AIMO-2 winning solution: Building state-of-the-art mathematical reasoning models with OpenMathReasoning dataset. *arXiv preprint arXiv:2504.16891*, 2025.

Cheng Qian, Emre Can Acikgoz, Qi He, Hongru Wang, Xiusi Chen, Dilek Hakkani-Tur, Gokhan Tur, and Heng Ji. ToolRL: Reward is all tool learning needs. In *Advances in Neural Information Processing Systems*, 2025.

Adityanarayanan Radhakrishnan, Daniel Beaglehole, Parthe Pandit, and Mikhail Belkin. Mechanism for feature learning in neural networks and backpropagation-free machine learning models. *Science*, 383(6690):1461–1467, 2024.

Ronald Skorobogat, Karsten Roth, and Mariana-Iuliana Georgescu. Subspace-boosted model merging. *arXiv preprint arXiv:2506.16506*, 2025.

George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with SVD to tie the Knots. In *International Conference on Learning Representations*, 2025.

Jiakang Wang, Runze Liu, Qingpeng Cai, Lei Lin, Wenping Hu, Xiu Li, Fuzheng Zhang, Guorui Zhou, Kun Gai, and Ling Pan. When importance sampling misallocates credit: Asymmetric ratios for outcome-supervised RL. *arXiv preprint arXiv:2510.06062*, 2025a.

Yinjie Wang, Ling Yang, Ye Tian, Ke Shen, and Mengdi Wang. Co-evolving LLM coder and unit tester via reinforcement learning. In *Advances in Neural Information Processing Systems*, 2025b.

Mitchell Wortsman, Gabriel Ilharco, Samir Yitzhak 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 *International Conference on Machine Learning*, 2022.

Xiaomi LLM-Core Team. MiMo-V2-Flash technical report. *arXiv preprint arXiv:2601.02780*, 2026.

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*, 2023.

Hongli Yu, Tinghong Chen, Jiangtao Feng, Jiangjie Chen, Weinan Dai, Qiying Yu, Ya-Qin Zhang, Wei-Ying Ma, Jingjing Liu, Mingxuan Wang, and Hao Zhou. MemAgent: Reshaping long-context LLM with multi-conv RL-based memory agent. In *International Conference on Learning Representations*, 2026.

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 *International Conference on Machine Learning*, 2024.

Xiangchi Yuan, Dachuan Shi, Chunhui Zhang, Zheyuan Liu, Shenglong Yao, Soroush Vosoughi, and Wenke Lee. Behavior knowledge merge in reinforced agentic models. In *Annual Meeting of the Association for Computational Linguistics*, 2026.

Zhilong Zhang, Hongli Yu, Huan-ang Gao, Hanlin Wu, Yuxuan Song, Wei-Ying Ma, Ya-Qin Zhang, and Hao Zhou. Spectral rewiring for exploration, purification, and model merging. *arXiv preprint arXiv:2607.03065*, 2026.

Hanqing Zhu, Wenyan Cong, Zhizhou Sha, Sagnik Mukherjee, Xinyuan Song, David González-Martínez, Xiaoxia Wu, Yuandong Tian, Shiwei Liu, David Z. Pan, and Zhangyang Wang. ISO: An RLVR-native optimization stack. *arXiv preprint arXiv:2607.19331*, 2026a.

Kejian Zhu, Zhuoran Jin, Shangqing Tu, Hongbang Yuan, Yushi Bai, Kang Liu, Juanzi Li, and Jun Zhao. SFT conflicts, RL coexists: A theoretical and empirical analysis of multi-task learning for LLMs. *arXiv preprint arXiv:2608.03573*, 2026b.

A Evaluation Details

The Qwen2.5-7B-Instruct experiments follow the previously established evaluation protocol (Zhu et al., 2026a). Coding uses LiveBench and LiveCodeBench v2 with 16 sampled responses per prompt at temperature 1.0, reporting pass accuracy and unit-test accuracy. BFCL v4 uses the official single-turn leaderboard pipeline with single-sample near-greedy decoding at temperature 0.001, reporting the official live and non-live aggregates. RULER memory uses HotpotQA and SQuAD at 32K and 64K context lengths with 16 sampled responses per prompt at temperature 0.7. Coding and memory report mean and population standard deviation over three independent runs; the overall average is the unweighted mean over the four coding columns, two BFCL columns, and four memory columns.

For the DeepSeek-R1-Distill-Qwen-1.5B setting, the shared base model is DeepSeek-R1-Distill-Qwen-1.5B, and the two RL experts are Archer2.0-Code-1.5B-Preview for coding (Wang et al., 2025a) and JustRL-DeepSeek-1.5B for math (He et al., 2025). All evaluations in this setting use temperature 0.6 and top-$p = 0.95$. Coding uses LiveBench and LiveCodeBench v5 with 16 sampled responses per prompt and reports both pass accuracy and unit-test accuracy. Math uses AIME 2024 and AIME 2025 with 32 samples, AMC 2023 with 8 samples, and Minerva Math and OlympiadBench with 4 samples. The reported means and population standard deviations are computed over three independent runs.

For the OpenMath-Nemotron-1.5B setting, the shared base model is OpenMath-Nemotron-1.5B (Moshkov et al., 2025), and the two RL experts are QuestA-Nemotron-1.5B (Li et al., 2026) and JustRL-Nemotron-1.5B (He et al., 2025). We use the same five math benchmarks as the DeepSeek-R1-Distill-Qwen-1.5B math evaluation. Following the evaluation protocol used by the expert models, decoding uses temperature 0.7, top-$p = 0.95$, and maximum generation length 32000. AIME 2024 and AIME 2025 use 32 samples per problem, AMC 2023 uses 8 samples, Minerva Math uses 4 samples per problem, and OlympiadBench uses 4 samples per problem. Following JustRL (He et al., 2025), we augment the rule-based grader with CompassVerifier-3B, a lightweight model-based verifier, to address false negatives from rule-based verifiers (Liu et al., 2025). The reported means and population standard deviations are computed over two independent runs.

B Ablation Details for the Qwen2.5-7B Merge

Table 2 forms a 2×2 ablation over the unit used for coefficient estimation and the representation of each expert effect. All variants use the raw self-retention solve with $\lambda = 10^{-3}$ and the same transfer of the solved coefficients back to the corresponding projection edges. For a unit u with expert representations $r_{u,i}$, each variant constructs

$$\Gamma_u[i,j] = \langle r_{u,i}, r_{u,j} \rangle, \qquad (\Gamma_u + \lambda s_u I) c_u = \mathrm{diag}(\Gamma_u), \tag{25}$$

where s_u is the mean positive diagonal energy for that unit.

The three ablations and the full method instantiate $r_{u,i}$ as follows:

- **PM-Delta**: $u = e$ is one projection matrix and $r_{e,i} = \mathrm{vec}(\Delta_{i,e})$. Each projection matrix solves for its own coefficient vector independently.
- **PM-Gram**: $u = e$ is one projection matrix and $r_{e,i} = S_{e,i}$, using the appropriate read- or write-side Gram shift. No incident projections are aggregated.
- **Node-Delta**: $u = v$ is one residual node and $r_{v,i} = \bigoplus_{e \in E(v)} \mathrm{vec}(\Delta_{i,e})$. The raw deltas of all incident projections are concatenated into one node-level representation before solving for a shared coefficient vector.
- **RGM**: $u = v$ is one residual node and $r_{v,i} = m_{v,i} = \sum_{e \in E(v)} S_{v,e,i}$. The Gram shifts of all incident projections are summed before the node-level solve.

Here $E(v)$ denotes the projection edges incident to residual node v. Thus the ablation separates the effect of the solving unit from that of the expert-effect representation: PM variants solve independently per projection matrix, Node variants solve at residual states, Delta variants use raw task-vector deltas, and Gram variants use Gram-shift representations.

C Additional Experiment: ViT-B/32 Eight-Task Merging

Setup. We also evaluate RGM on the 8-task CLIP image-classification setting introduced by Task Arithmetic (Ilharco et al., 2023). The shared base is the zero-shot ViT-B/32 checkpoint, and the eight expert checkpoints are separately fine-tuned on SUN397, Cars, DTD, RESISC45, SVHN, GTSRB, MNIST, and EuroSAT. To compare the two merge rules under a common global scale, both methods use $\lambda = 0.3$, the best-performing coefficient for Task Arithmetic in this setting:

$$W_{\mathrm{TA}} = W_0 + 0.3 \sum_{i=1}^{8} \Delta_i. \tag{26}$$

$$W_{\mathrm{RGM}} = W_0 + 0.3\, \Delta_{\mathrm{RGM}}. \tag{27}$$

Pre-merge branch selection. The branch-selection rule selects NormBalanced-RGM for this ViT-B/32 setting based on the cross-pressure diagnostic. Across the 36 residual nodes, cross-pressure exceeds one at 7 nodes, with D_v p95 equal to 2.0160 and $\max_v D_v = 2.7639$. These statistics place the ViT merge closer to the high-overlap 1.5B language-model settings than to the low-overlap Qwen2.5-7B-Instruct setting, so raw Gram-shift magnitudes are not used directly for coefficient estimation.

Table 6 | **Additional ViT-B/32 eight-task image-classification merge.** Entries are accuracies. TA uses the best global scaling coefficient from the Task Arithmetic protocol, $W_0 + 0.3 \sum_i \Delta_i$. RGM uses the same global scale, $W_0 + 0.3\,\Delta_{\mathrm{RGM}}$, after NormBalanced coefficient estimation.

Benchmark	Finetuned acc	RGM acc	TA acc	Δ (RGM–TA)
SUN397	75.29	64.19	55.26	$+8.93$
Cars	77.68	63.06	54.94	$+8.12$
DTD	79.41	52.98	50.11	$+2.87$
RESISC45	96.11	72.35	66.68	$+5.67$
SVHN	97.46	75.28	80.21	-4.93
GTSRB	98.73	63.52	69.65	-6.13
MNIST	99.69	94.40	97.34	-2.94
EuroSAT	99.89	82.57	77.31	$+5.26$
Overall (avg)	**90.53**	**71.04**	**68.94**	$\mathbf{+2.10}$

Results. Table 6 shows that RGM improves the eight-task average over the tuned Task Arithmetic baseline by 2.10 points. The gains are concentrated on the more difficult benchmarks: RGM improves SUN397 by 8.93 points, Cars by 8.12 points, and DTD by 2.87 points. Together, these three gains sum to 19.92 points, whereas the remaining five benchmarks, all of which are near-saturated under the fine-tuned experts, show a combined change of -3.07 points. This pattern suggests that residual-state coefficient estimation can recover useful structure in a classical vision merging benchmark, especially where the merged model has substantial headroom relative to the corresponding expert. The results show that RGM also captures useful structure in ViT model merging, yielding meaningful gains.

D Consistency of the Self-Retention Surrogates

The coefficient solve is carried out on expert-induced residual-state sections, while the actual merged checkpoint is formed in weight space. Consequently, the Gram shift of the final checkpoint contains quadratic cross terms in the expert deltas and need not equal the linear surrogate used by the solver. Let W^* be the final merged weight matrix after applying the solved coefficients. For each residual node v, the actual merged section is:

$$m_v^{\mathrm{act}} = \sum_{e \text{ incident to } v} \left(\phi(W_e^*) - \phi(W_{0,e})\right), \tag{28}$$

where $\phi(W) = W^T W$ for read projections and $\phi(W) = W W^T$ for write projections. We first evaluate the raw branch using a true-retention check in the Qwen2.5-7B-Instruct setting. For each expert section $m_{v,i}$, we define raw true retention as:

$$\mathrm{ret}_i^{\mathrm{raw}}(v) = \frac{\langle m_v^{\mathrm{act}},\, m_{v,i} \rangle_F}{\|m_{v,i}\|_F^2 + \epsilon}. \tag{29}$$

Table 7: Raw-branch true retention for the Qwen2.5-7B-Instruct merge. Values are averaged over the 84 residual nodes.

Setting	CURE	ToolRL	MemAgent	Mean
Qwen2.5-7B-Instruct	0.9651	1.0065	1.0001	0.9906

Across the 84 residual nodes, raw true retention remains close to one for all three experts: 0.9651 for CURE, 1.0065 for ToolRL, and 1.0001 for MemAgent, with an overall mean of 0.9906. This indicates that the raw self-retention solve preserves the experts' state-level Gram effects in the low-overlap 7B setting, with little systematic attenuation or amplification.

For settings for which the pre-merge cross-pressure gate selects NormBalanced-RGM, we evaluate agreement with the NormBalanced surrogate. Using the normalized expert sections $\hat{m}_{v,i}$ from the NormBalanced branch, we define the NormBalanced surrogate at residual node v as:

$$m_{v,\mathrm{nb}}^{\mathrm{sur}} = \sum_i c_{v,i} \hat{m}_{v,i}. \tag{30}$$

We quantify this agreement with the Frobenius cosine:

$$\rho_{\mathrm{nb}}(v) = \frac{\langle m_v^{\mathrm{act}},\, m_{v,\mathrm{nb}}^{\mathrm{sur}} \rangle_F}{\|m_v^{\mathrm{act}}\|_F \, \|m_{v,\mathrm{nb}}^{\mathrm{sur}}\|_F + \epsilon}. \tag{31}$$

We report the mean and standard deviation of $\rho_{\mathrm{nb}}(v)$ over residual nodes.

Unlike the raw branch, NormBalanced-RGM intentionally removes the per-expert section magnitudes before solving for coefficients. Its target is therefore the NormBalanced surrogate rather than the per-expert raw true-retention values. We therefore evaluate directional agreement between the actual merged Gram shift and the aggregate NormBalanced surrogate using the Frobenius cosine $\rho_{\mathrm{nb}}(v)$.

Table 8: Frobenius cosine between the actual merged Gram shift and the NormBalanced surrogate. Values are computed over residual nodes; higher mean and lower standard deviation indicate closer directional agreement.

Setting	Mean ρ_{nb}	Std. ρ_{nb}
DeepSeek-R1-Distill-Qwen-1.5B	0.96840	0.00967
OpenMath-Nemotron-1.5B	0.97357	0.01570
ViT-B/32 (global scale 0.3)	0.9930	0.0045

The high cosine values across all three settings show that the actual weight-space merge remains closely aligned with the NormBalanced surrogate despite the omitted quadratic cross terms.

E Residual-State Diagnostics Across Merging Settings

This appendix visualizes the residual-state quantities used by RGM before coefficient aggregation. For the two-expert DeepSeek-R1-Distill-Qwen-1.5B and OpenMath-Nemotron-1.5B settings, we show normalized section Gram matrices, where each off-diagonal entry is the cosine overlap between the two expert-induced residual-state sections at a given layer and node. For the three-expert Qwen2.5-7B-Instruct setting, the raw branch retains both direction and magnitude, so we display the same raw Gram information through the exact factorization $\Gamma_v[i,j] = n_{v,i} n_{v,j} \rho_{v,i,j}$ into signed pairwise cosine overlaps and section norms. This decomposition includes every layer and expert pair while keeping small cross terms visible despite the large norm differences among experts. Together, the diagnostics expose two complementary regimes: strong directional alignment in the 1.5B settings, and near-orthogonality with substantial magnitude imbalance and localized overlap in the 7B setting.

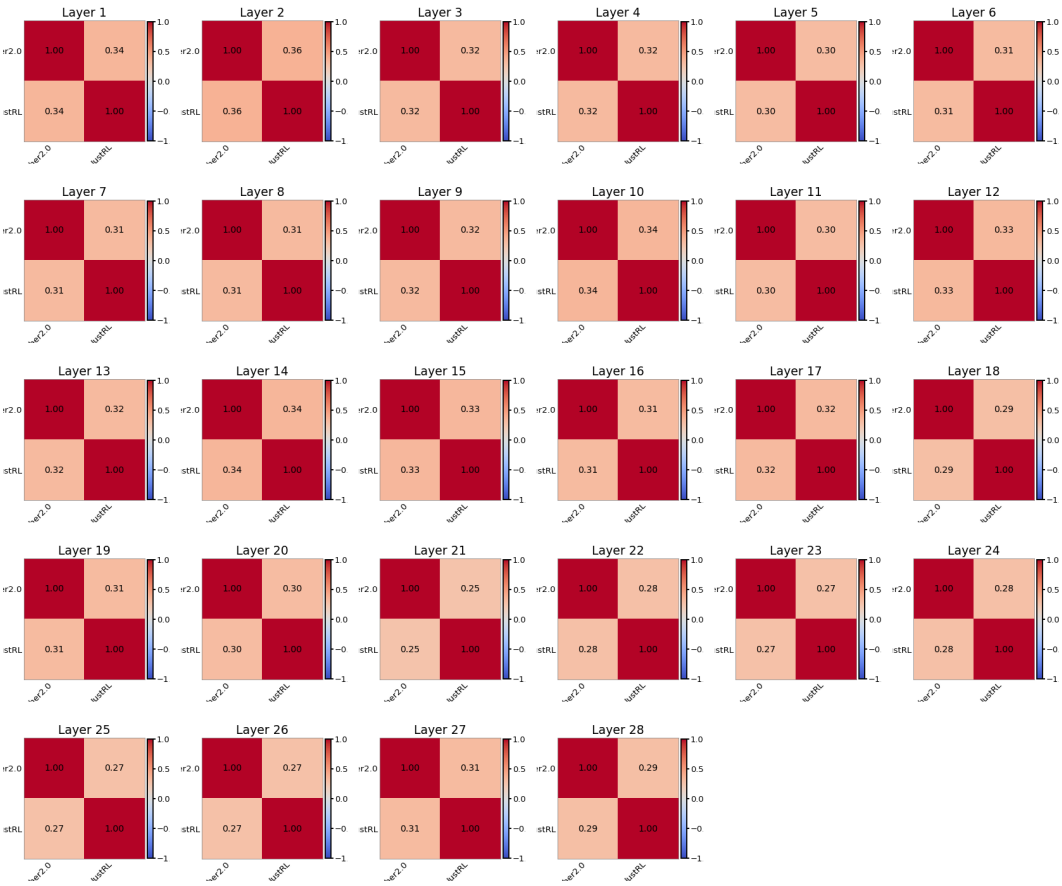

Figure 2: **DeepSeek-R1-Distill-Qwen-1.5B: normalized Gram overlap at the attention-input residual state.** Each panel shows the Archer2.0 and JustRL section overlap for one transformer layer.

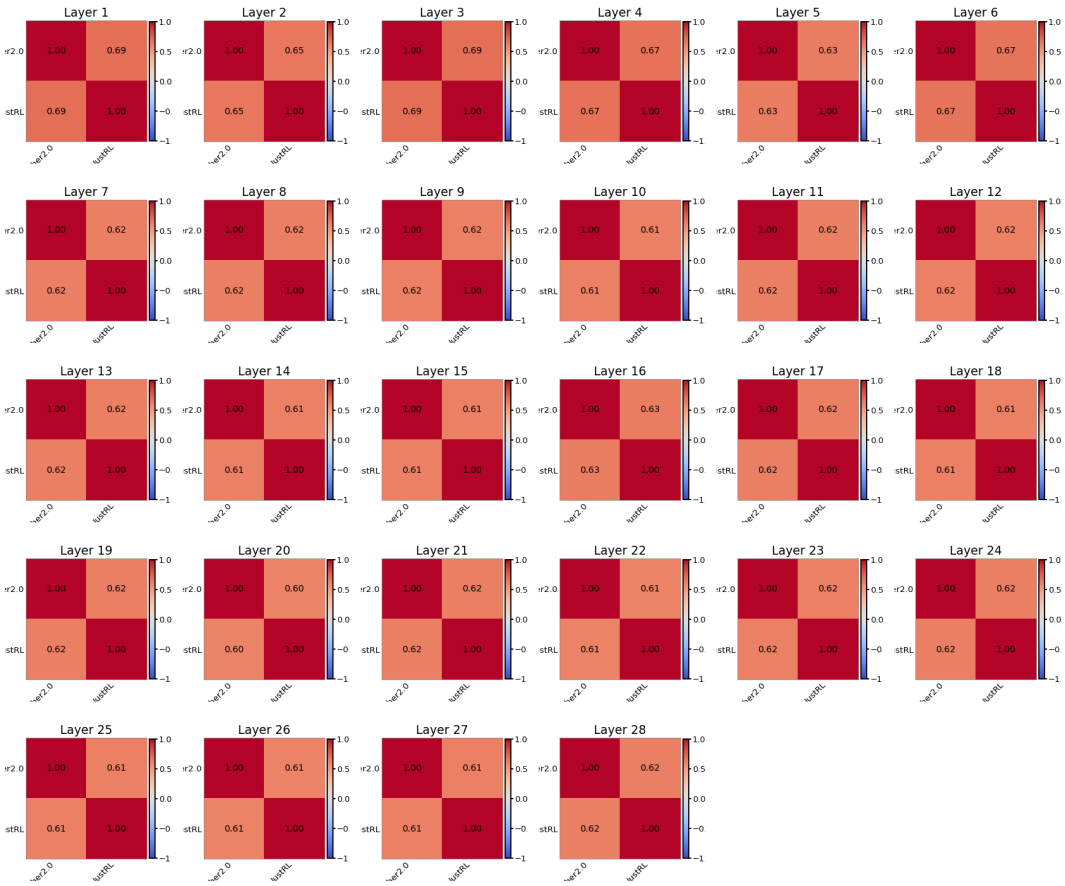

Figure 3: **DeepSeek-R1-Distill-Qwen-1.5B: normalized Gram overlap at the post-attention residual state.** The two experts exhibit consistently stronger positive overlap after attention than at the attention-input state.

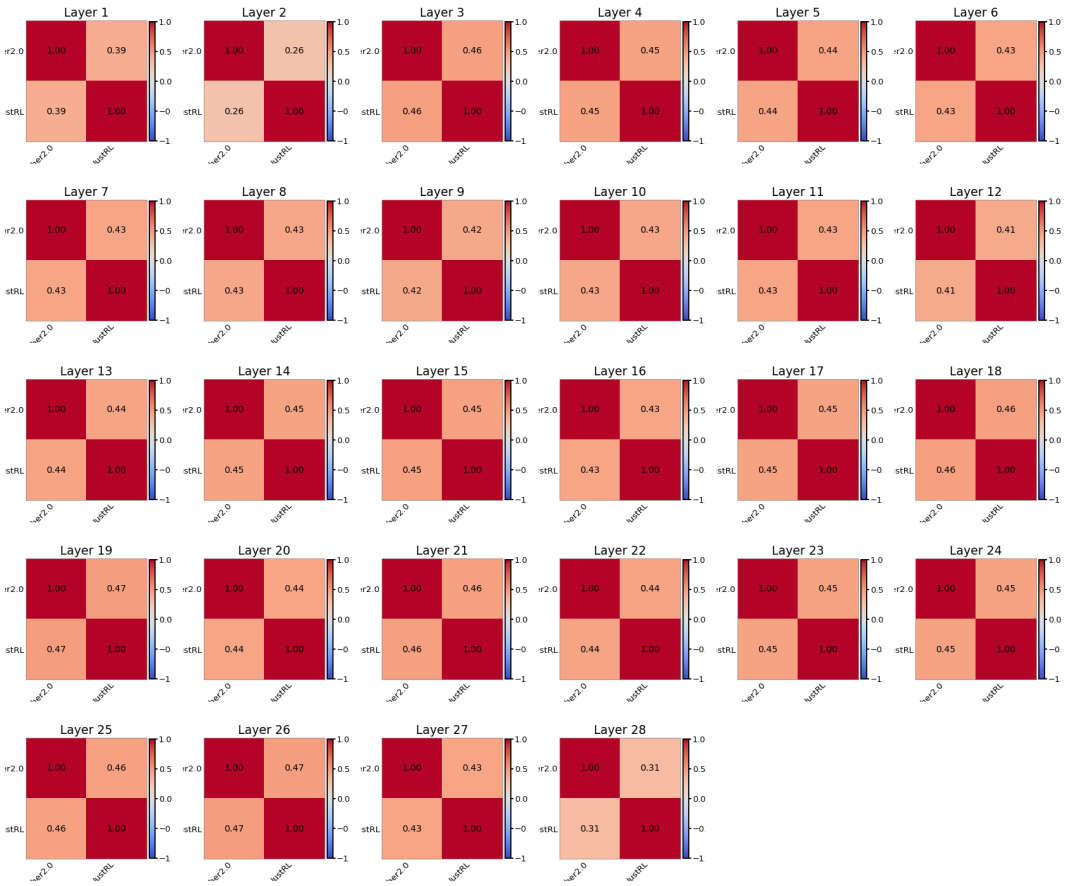

Figure 4: **DeepSeek-R1-Distill-Qwen-1.5B: normalized Gram overlap at the post-MLP residual state.** The post-MLP sections show moderate positive overlap across most layers.

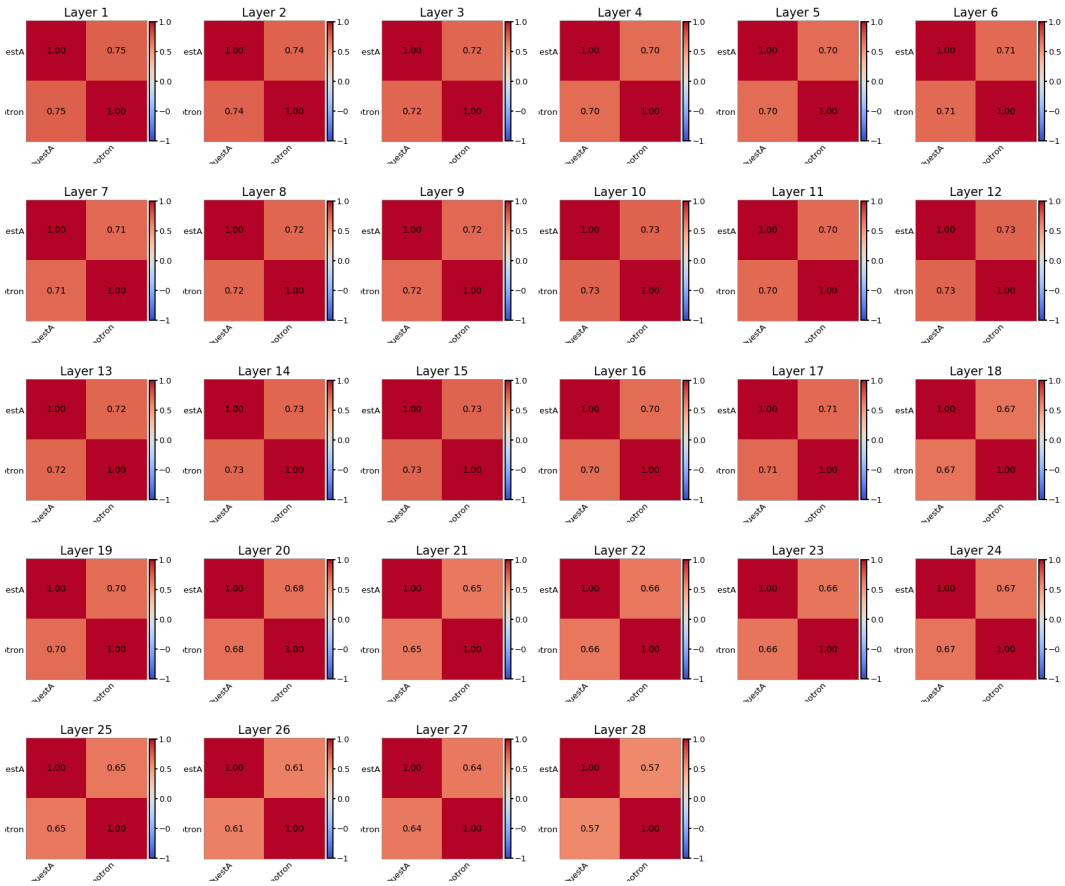

Figure 5: **OpenMath-Nemotron-1.5B: normalized Gram overlap at the attention-input residual state.** QuestA-Nemotron-1.5B and JustRL-Nemotron-1.5B are already highly aligned before attention in most layers.

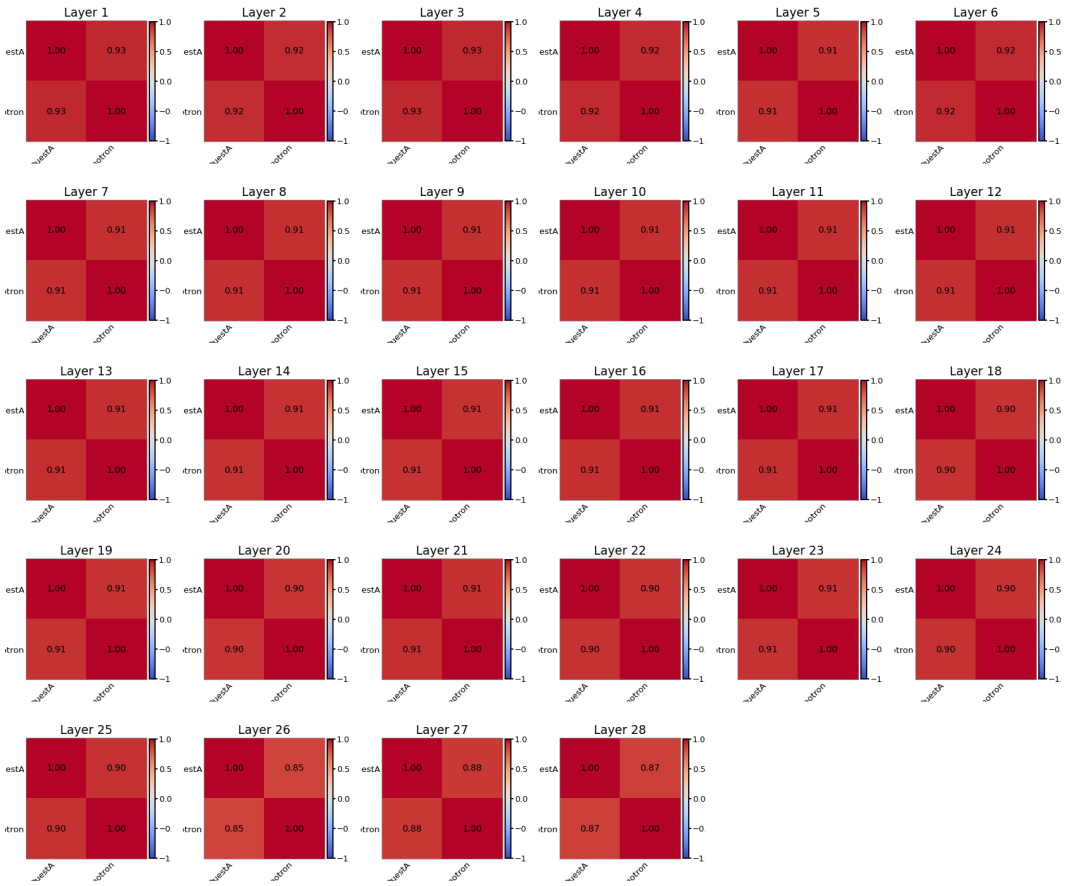

Figure 6: **OpenMath-Nemotron-1.5B: normalized Gram overlap at the post-attention residual state.** The post-attention sections show near-collinear positive overlap throughout the network.

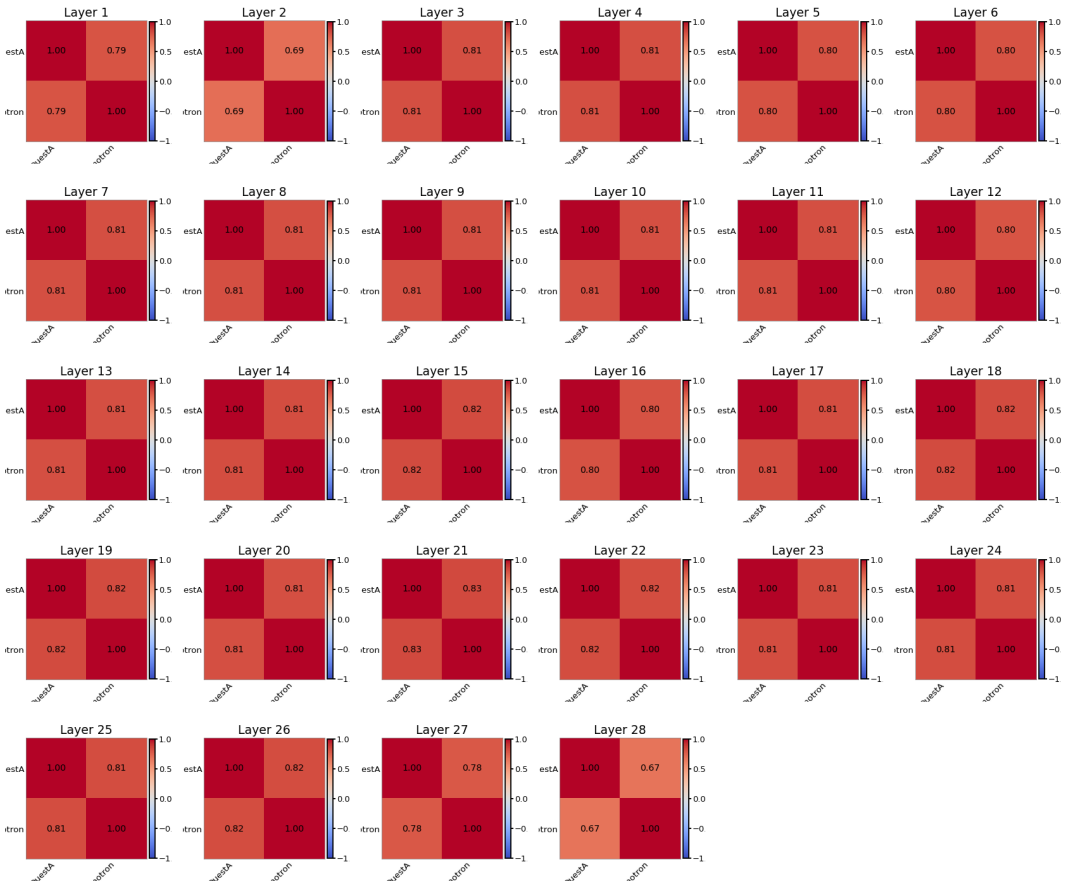

Figure 7: **OpenMath-Nemotron-1.5B: normalized Gram overlap at the post-MLP residual state.** The two mathematics-specialized experts remain strongly aligned after the MLP update.

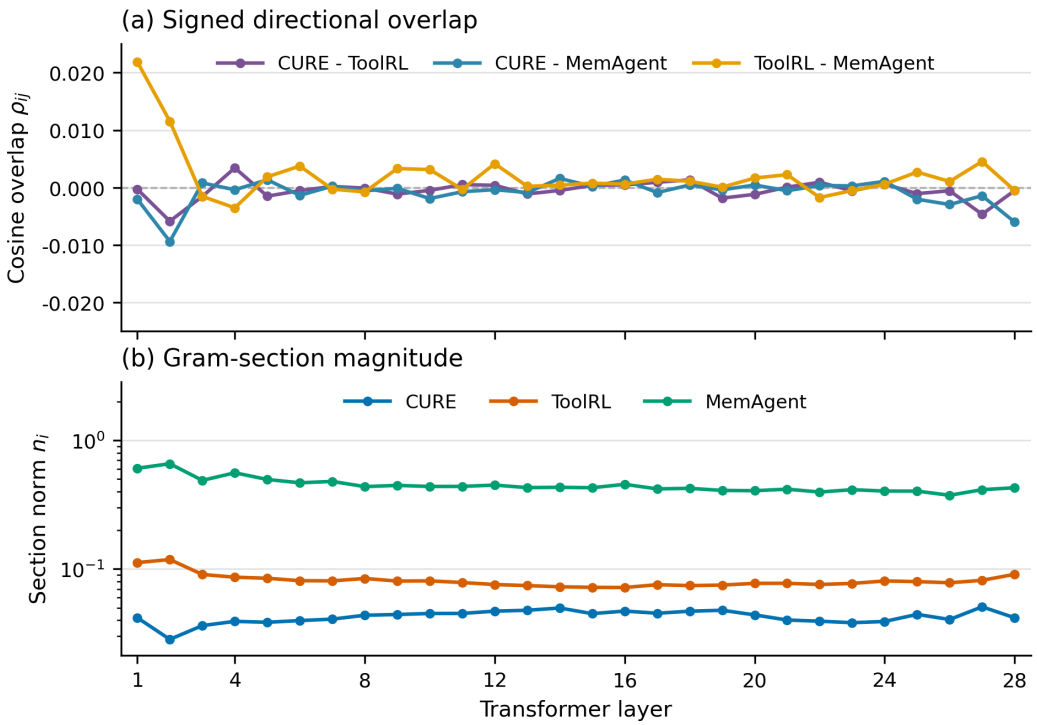

Figure 8: **Qwen2.5-7B-Instruct: raw residual-state geometry at the attention input.** The upper panel shows all signed pairwise cosine overlaps, and the lower panel shows all section norms on a log scale; together they exactly determine the raw Gram matrix at every layer.

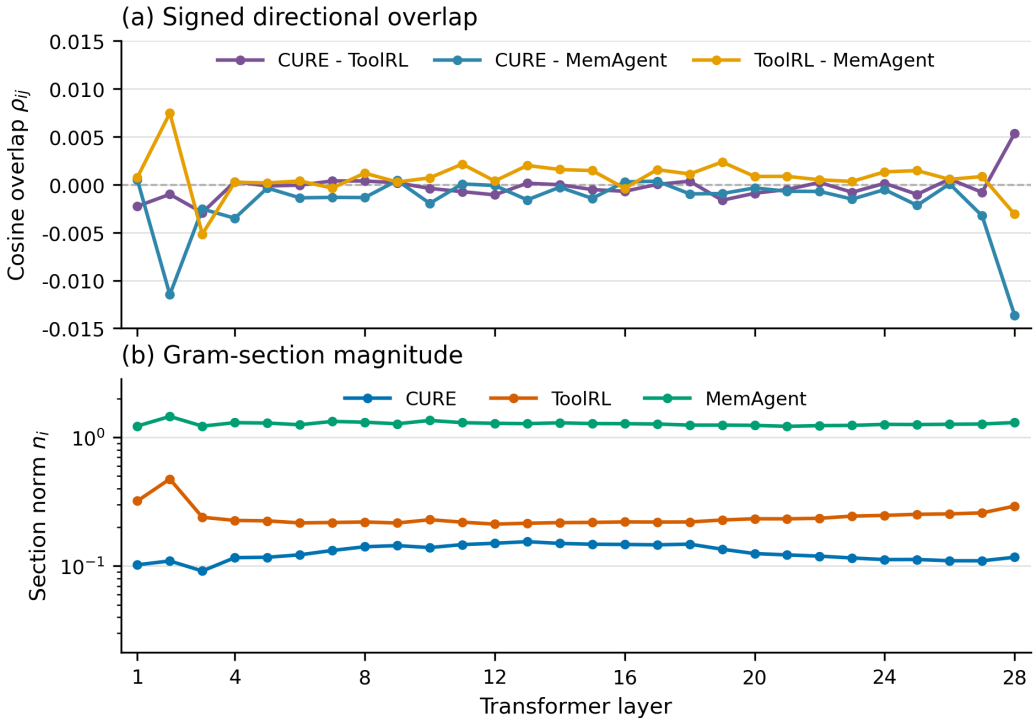

Figure 9: **Qwen2.5-7B-Instruct: raw residual-state geometry after attention.** Directional overlap remains close to zero across most layers, while the section norms maintain a stable expert-wise magnitude ordering.

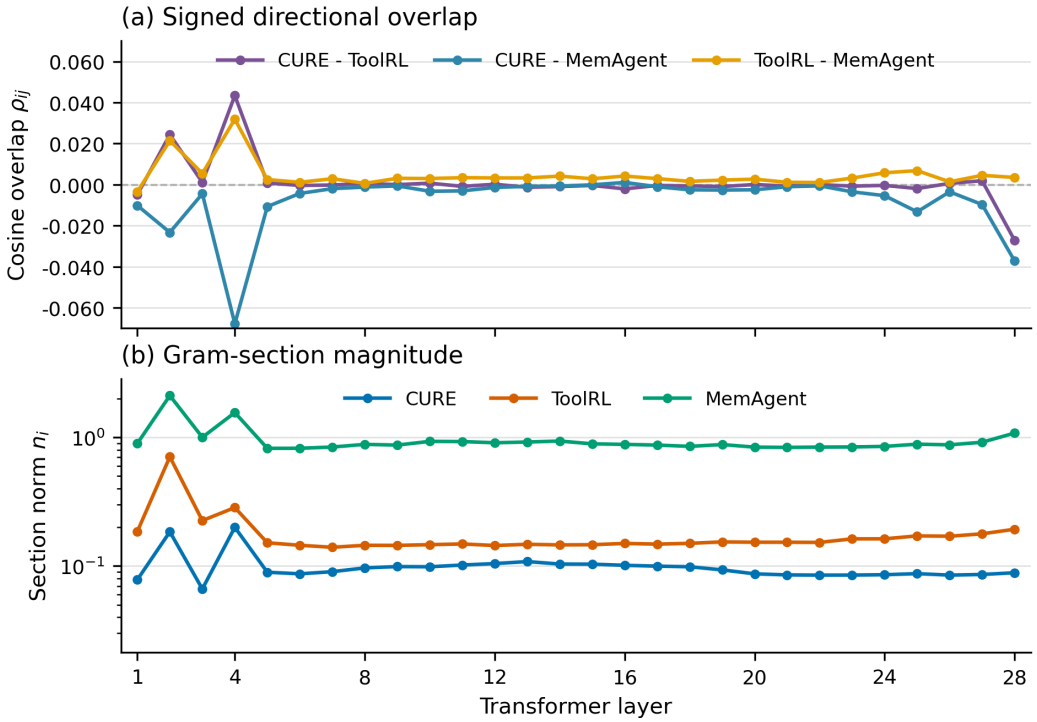

Figure 10: **Qwen2.5-7B-Instruct: raw residual-state geometry after the MLP.** The post-MLP state contains the most visible localized overlap, particularly in early layers, while the expert section norms remain strongly separated.

F The Use of Large Language Models

We used large language models only to polish portions of the language in this paper. All content was originally drafted by the authors. Their use was limited to refining pre-existing text, and all suggested modifications were reviewed by the authors to verify their accuracy and consistency with the original meaning.