RETHINKING HETEROGENEOUS LLM MERGING: A WEIGHTED MODEL AVERAGING PERSPECTIVE

Anonymous authors
Paper under double-blind review

ABSTRACT

Can large language models with substantially different parameter spaces be merged by direct weighted averaging, without training or semantic alignment? Existing heterogeneous fusion methods typically introduce distillation, adapters, learned latent spaces, routing, or feature alignment, leaving open whether a simpler recipe can work for genuinely different billion-parameter checkpoints. We revisit this counterintuitive question through training-free dimensional adaptation followed by ratio-controlled interpolation. In intersection-style merging, we truncate the larger model into the smaller parameter space; in union-style merging, we expand the smaller model into the larger parameter space (Figure 1). Across Qwen-family model pairs and benchmarks covering mathematical reasoning, code generation, language understanding, commonsense reasoning, knowledge, and instruction following, deterministic expansion largely preserves the source model function, and small-ratio interpolation can improve over strong source checkpoints by transferring complementary capabilities. However, near-balanced interpolation often collapses, and task-level results reveal a seesaw effect in which gains on some capabilities coexist with regressions on others. These results show that simple parameter averaging, when paired with lightweight dimensional adaptation and carefully controlled ratios, is a surprisingly strong baseline for heterogeneous LLM merging, suggesting that the limits of direct weighted fusion may also bound what more complex heterogeneous merging methods can achieve at scale.

1 INTRODUCTION

Model merging has become a compelling post-training paradigm for large language models, because it offers a direct way to combine capabilities from existing checkpoints without full retraining (Yang et al., 2026; Song and Zheng, 2026; Lu et al., 2024). In homogeneous settings, a broad family of methods has been proposed, including weight averaging and Model Soups (Wortsman et al., 2022), task arithmetic (Ilharco et al., 2023), TIES-Merging (Yadav et al., 2023), Fisher-weighted averaging (Matena and Raffel, 2022), RegMean (Jin et al., 2023), and sparsity-based variants (Davari and Belilovsky, 2024). Yet these successes largely rely on a strong compatibility condition: the source checkpoints share architecture, parameter shapes, and at least partially aligned internal representations. This condition is increasingly restrictive, since real model pools contain checkpoints of different scales, specializations, and design choices (Lu et al., 2024; Chen et al., 2025).

Heterogeneous model merging relaxes this compatibility condition, but existing solutions often pay for it with extra machinery. Training-based approaches use distillation, generated data, adapters, modular skill storage, or learned latent spaces to transfer knowledge across models (Hinton et al., 2015; Pfeiffer et al., 2021; Wan et al., 2024a;b; Du et al., 2025a; Soro et al., 2026). Training-free approaches reduce mismatch through permutation matching, activation repair, feature zipping, semantic alignment, or coefficient selection (Ainsworth et al., 2023; Jordan et al., 2023; Stoica et al., 2024; Gu et al., 2025; Du et al., 2025b; Hackmann, 2024). These methods are valuable, but they leave a sharper question unresolved: can we merge sufficiently different billion-parameter language models with low cost, no additional training, and no explicit semantic alignment? This question is especially important because direct weighted averaging across mismatched parameter spaces appears counterintuitive: a larger model and a smaller model do not even expose tensors of the same shape, and representational mismatch weakens the assumptions behind neuron-wise or feature-wise correspondence.

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.

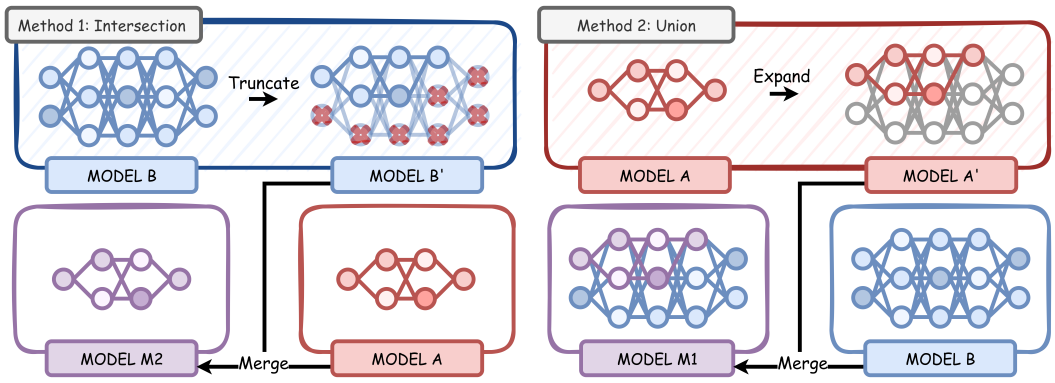


Figure 1: Two heterogeneous model merging strategies. (1) Intersection-style merging: truncate the larger model to the smaller model's dimensionality, then merge. (2) Union-style merging: expand the smaller model to match the larger model's dimensionality, then merge.

(a) Benchmarks

Family	Benchmarks
Math	GSM8K, MATH500
Code generation	HumanEval, MBPP
Instruction following	IFEval
QA and NLU	MNLI, RTE, QNLI, PIQA, Winogrande, COPA, BoolQ
Commonsense & knowledge	ARC, HellaSwag, MMLU
Comprehensive reasoning	BBH

(b) Model pairs

ID	Smaller model	Larger model
P1	Qwen2.5-14B	Qwen2.5-32B
P2	Qwen2.5-3B	Qwen2.5-32B
P3	Qwen3-4B-Thinking-2507	Qwen3-8B
P4	Qwen3-14B	Qwen3-32B
P5	Llama-3.2-3B	Llama-3.1-8B
P6	Llama-3.2-3B-Instruct	Llama-3.1-70B-Instruct
P7	Gemma-2-2B	Gemma-2-9B
P8	Gemma-2-2B-IT	Gemma-2-9B-IT

Table 1: Benchmark families and model-pair configurations. P3 uses Qwen3-4B-Thinking-2507 (Qwen Team, 2025).

The technical challenge has two parts. First, parameter-shape mismatch prevents direct interpolation, so the models must be placed in a common parameter space before any weighted averaging is possible. Second, even after shape compatibility is enforced, interpolation can destroy capability if the two solutions are not connected by a stable low-loss path; related homogeneous studies show that interpolation depends strongly on mode connectivity, permutation alignment, and representation similarity (Garipov et al., 2018; Frankle et al., 2020; Ainsworth et al., 2023; Raghu et al., 2017; Kornblith et al., 2019). Recent work on cross-scale fusion and scaling behavior further suggests that the effectiveness of merging rules may change at LLM scale (Chen et al., 2026; Wang et al., 2025). This motivates a controlled test of the simplest possible hypothesis: after deterministic dimensional adaptation, is direct weighted averaging actually enough, provided that the mixing ratio is small?

We answer this question through training-free dimensional adaptation followed by ratio-controlled interpolation. In the intersection-style setting, we truncate the larger checkpoint into the smaller parameter space before merging; in the union-style setting, we expand the smaller checkpoint into the larger parameter space before merging. The resulting procedure uses no fine-tuning, no distillation data, no adapters, no routing module, no learned latent encoder, and no semantic alignment step. This makes the intervention deliberately simple and interpretable: performance changes can be attributed primarily to dimensional adaptation, mixing ratio, and the compatibility of the two source checkpoints.

Across Qwen-family model pairs at billion-parameter scale and diverse benchmark families, this simple procedure works more often than the standard intuition would suggest. On Qwen2.5-3B, every intersection ratio $\mu \in \{0.01, 0.02, 0.03, 0.04\}$ already improves the 16-task mean (up to 55.54 from 54.59); the task-wise best envelope on the same grid reaches 57.16. In union-style merging between Qwen3-14B and Qwen3-32B, the two sources score 74.27 and 74.74, while the task-wise Potential envelope reaches 77.44 (Table 3). Task-level results reveal a seesaw effect in which gains on some capabilities coexist with regressions on others.

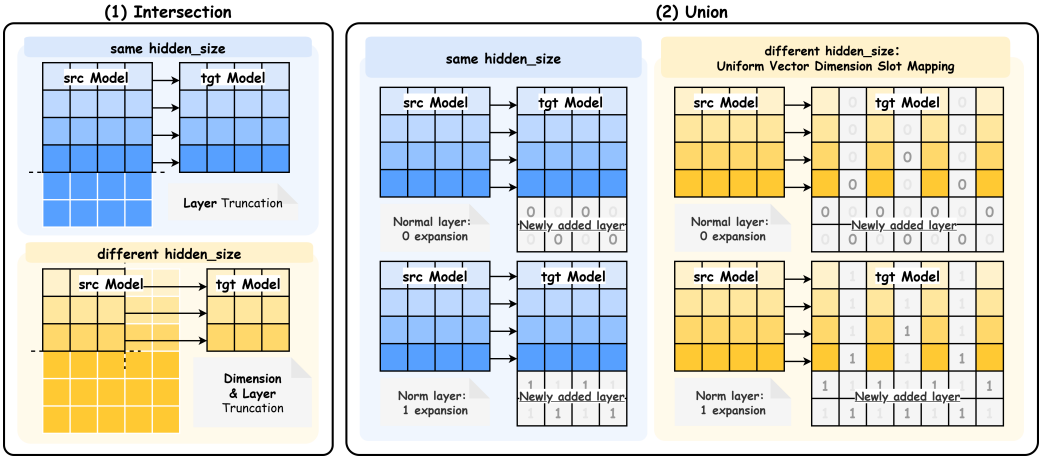


Figure 2: Dimensional adaptation procedures for heterogeneous model merging. (a) Truncation procedure for intersection-style merging. A larger checkpoint is projected into the smaller parameter space by copying compatible tensors, slicing oversized tensors, and removing layers beyond the target depth. (b) Initialization scheme for heterogeneous layer extension. Compatible source tensors are copied into the target model; newly introduced attention and MLP branches are initialized to zero, and normalization scales are initialized to one.

These results show that simple parameter averaging, paired with lightweight dimensional adaptation, is a strong baseline for heterogeneous LLM merging, and that the limits of direct weighted fusion may also bound what more complex heterogeneous methods can achieve at scale (Hitit et al., 2026; Yadav et al., 2025). Our contributions are threefold. (1) Direct weighted averaging of heterogeneous language models is counterintuitive enough that the field has not tried this simple linear operation; we show that it is possible once the checkpoints are placed in a common parameter space, with no training and no semantic alignment. (2) We evaluate Qwen, Llama, and Gemma pairs at billion-parameter scale on 16 benchmarks, a broader suite than existing heterogeneous merging studies. (3) Some benchmarks improve substantially, while others seesaw, and the strength of that seesaw depends on the model family: Qwen gains the most, whereas Llama and Gemma improve on individual datasets but show a milder overall effect.

2 Related Work

Recent surveys summarize model merging for LLMs, MLLMs, and related foundation-model settings (Yang et al., 2026; Song and Zheng, 2026).

Homogeneous model merging. Prior work has largely focused on homogeneous settings, where models share architecture and parameter layout. Weight-based methods include weight averaging and Model Soups (Wortsman et al., 2022), task arithmetic (Ilharco et al., 2023), TIES-Merging (Yadav et al., 2023), and DARE-style variants (Yu et al., 2024). Geometry- or statistics-based methods include SLERP (Shoemake, 1985), Fisher-weighted averaging (Matena and Raffel, 2022), and RegMean (Jin et al., 2023). Sparse-mask merging reduces interference by retaining selected parameter updates (Davari and Belilovsky, 2024).

Limitations in homogeneous settings. Homogeneous merging relies on aligned parameter shapes, similar internal representations, and often careful hyperparameter tuning. More sophisticated strategies add optimization or engineering overhead, and scaling-law evidence suggests that merging gains do not automatically scale with model size (Wang et al., 2025). In a recent in-the-wild evaluation with heterogeneous, potentially overlapping or conflicting fine-tuned checkpoints, Task Arithmetic was the only evaluated method that reliably yielded gains, while more complex subspace- or interference-aware methods often failed to provide clear improvements (Hitit et al., 2026). These constraints limit reuse of diverse real-world model pools.

Subspace-based merging and benchmarks. A closely related line reduces interference by localizing or matching task-relevant parameter subspaces. TALL-masks localize sparse task information and use the resulting supports for compression and consensus merging (Wang et al., 2024); Localize-and-Stitch identifies compact task-localized regions and stitches them into a pretrained model through sparse task arithmetic (He et al., 2024); and MaTS formulates merging as matching models in task parameter subspaces through a linear-system solution (Tam et al., 2024). These methods are connected to our intersection-style setting because both restrict the region of parameter space that participates in merging. The difference is that subspace-based methods usually estimate task supports or matching subspaces for shape-compatible task vectors, whereas our truncate-then-merge protocol first applies a deterministic architecture projection from a larger checkpoint to a smaller parameter space and then performs small-ratio interpolation. We therefore ask whether useful projected perturbations can be absorbed without learning a task subspace, solving a matching problem, or requiring additional optimization. Recent benchmarks such as MergeBench further emphasize the need for explicit evaluation of domain-specialized LLM merging methods and failure modes (He et al., 2025); our setting is complementary because it focuses on cross-scale dimensional adaptation rather than standard same-shape merging.

Heterogeneous model merging. Heterogeneous merging is more challenging because parameter-wise averaging is no longer obviously meaningful when source models differ in width, depth, module layout, or even modality. Existing approaches therefore often introduce additional mechanisms to make cross-model reuse feasible. One line of work uses training or learned intermediate modules: knowledge distillation transfers behavior through teacher–student optimization (Hinton et al., 2015), AdapterFusion composes trained adapters rather than raw checkpoints (Pfeiffer et al., 2021), FuseLLM and FuseChat fuse knowledge through model-generated data and subsequent training (Wan et al., 2024a;b), GraftLLM stores capabilities in modular SkillPacks (Du et al., 2025a), and LS-Merge learns a latent weight space before merging language models (Soro et al., 2026). These methods broaden the reuse of heterogeneous model assets, but their reliance on optimization, generated or curated data, trained modules, or latent encoders weakens the original appeal of model merging as a fast training-free operation.

A second line seeks training-free compatibility through alignment or repair. Git Re-Basin aligns models modulo permutation symmetries (Ainsworth et al., 2023), REPAIR corrects activation statistics after permutation-based interpolation (Jordan et al., 2023), and ZipIt merges models by matching and zipping related features across networks (Stoica et al., 2024). For language and multimodal models, semantic-alignment and coefficient-selection methods further reduce mismatch before interpolation, as in SeMe and AdaMMS (Gu et al., 2025; Du et al., 2025b), while HM3 studies heterogeneous multi-class classifier merging (Hackmann, 2024). However, many of these methods either focus on relatively small networks, require explicit semantic or feature alignment, or study heterogeneity that remains close to a shared base or task structure. As a result, it remains unclear whether sufficiently different billion-parameter language models can be merged by a simpler recipe: adapt dimensions without training and then directly interpolate weights.

Recent studies on cross-scale language-model fusion and model-merging scaling further motivate this question (Chen et al., 2026; Wang et al., 2025). Homogeneous LLM evaluations already find that simple linear rules remain competitive at scale: Task Arithmetic is often the only reliable in-the-wild method (Hitit et al., 2026), and averaging, TIES, and Task Arithmetic become hard to distinguish for large models (Yadav et al., 2025). Heterogeneous merging may follow a similar trajectory if a deterministic shape map is enough to restore a linear baseline. Our work tests this possibility directly by removing learned alignment, adapters, routing, and post-hoc training, and by asking whether dimensional adaptation plus small-ratio interpolation can transfer complementary capabilities across genuinely different LLM scales.

Position of this work. We target training-free heterogeneous merging under parameter-scale mismatch with lightweight dimensional adaptation and ratio-controlled interpolation. Homogeneous LLM studies already rehabilitate simple linear merging at scale (Hitit et al., 2026; Yadav et al., 2025); we ask whether that conclusion survives a deterministic shape map between unequal architectures.

3 METHOD

3.1 PROBLEM SETUP

We consider two pretrained models, denoted by A and B, where model A is no larger than model B in parameter dimensionality. Let $\theta_A \in \mathbb{R}^{d_A}$ and $\theta_B \in \mathbb{R}^{d_B}$ denote their parameter vectors, with $d_A \leq d_B$. We study two complementary merging settings, referred to as *intersection-style merging* and *union-style merging*.

3.2 INTERSECTION-STYLE MERGING (TRUNCATE THEN MERGE)

We first project the larger model B into the smaller parameter space of model A via a truncation operator:

$$\mathcal{T} : \mathbb{R}^{d_B} \to \mathbb{R}^{d_A}, \qquad \tilde{\theta}_B = \mathcal{T}(\theta_B). \tag{1}$$

As shown in Figure 2(a), we instantiate the small-model configuration and initialize it from the large checkpoint by copying shape-compatible tensors, slicing oversized tensors, and removing layers beyond the target depth. Truncation discards parameters and is inherently lossy, so the projected checkpoint enters the merge as a small perturbation and μ is kept small. Merging is performed in the low-dimensional space:

$$\theta^{\cap}(\mu) = (1-\mu)\theta_A + \mu\tilde{\theta}_B, \qquad \mu \in [0,1]. \tag{2}$$

3.3 UNION-STYLE MERGING (EXPAND THEN MERGE)

Complementary to truncation, we map model A to the larger space of model B through an expansion operator

$$\mathcal{E} : \mathbb{R}^{d_A} \to \mathbb{R}^{d_B}, \qquad \tilde{\theta}_A = \mathcal{E}(\theta_A). \tag{3}$$

Expansion deterministically copies compatible source tensors into the larger target tensors and initializes newly introduced dimensions to minimize functional perturbation, as illustrated in Figure 2(b): newly introduced linear parameters are zero-initialized, normalization scales are initialized to neutral values, and additional Transformer blocks (Vaswani et al., 2017) are initialized as residual identity mappings. Architecture-aware mappings preserve grouped-query attention structure (Ainslie et al., 2023) and expand MLP matrices along their semantic input or output dimensions. Further normalization details are given in Appendix D. We then interpolate near a source, with small λ or λ close to one, so the expanded checkpoint acts as a small perturbation of a working model:

$$\theta^{\cup}(\lambda) = (1-\lambda)\tilde{\theta}_A + \lambda\theta_B, \qquad \lambda \in [0,1]. \tag{4}$$

3.4 EVALUATION PROTOCOL

Intersection-style and union-style merging are two strategies on fixed ratio grids. Each grid point is a real model, and a single fixed ratio can already improve the source. Main-text tables also report Potential, the per-task best on that grid. Potential is not itself one fixed ratio:

$$\text{Potential}_t^{\cap} = \max_{\mu \in M} S_t\big(\theta^{\cap}(\mu)\big), \qquad \text{Potential}_t^{\cup} = \max_{\lambda \in \Lambda} S_t\big(\theta^{\cup}(\lambda)\big), \tag{5}$$

where S_t is the score on task t and the grids are specified in Section 4. Those grids are the operating ratios. Appendix C reports the 16-task mean of each fixed ratio on the same grids, so the fixed-ratio scores themselves are already given. Appendix A shows that merging at coarser ratios still collapses performance, illustrated on GSM8K. A single ratio cannot dominate all tasks; this paper uses a simple fixed grid search to show what the linear method can achieve, and leaves automatic selection of μ or λ without a per-task oracle to future work.

4 EXPERIMENT

We evaluate Qwen2.5 (Yang et al., 2024) and Qwen3 (Yang et al., 2025) model pairs ranging from 3B to 32B parameters. The benchmark suite spans mathematical reasoning, code generation, instruction following, NLU, commonsense, knowledge, and comprehensive reasoning, using

Dataset	3B	Potential	Dataset	3B	Potential
gsm8k	65.66	(+6) **72.10**	QNLI	52.86	(+2) **55.26**
MATH500	38.40	(+2) **40.60**	RTE	64.26	(+13) **77.26**
HumanEval	**43.90**	43.29	Winogrande	63.38	**64.01**
MBPP	58.99	(+2) **60.58**	ARC	**34.42**	34.30
BoolQ	70.80	(+2) **72.72**	HellaSwag	**50.94**	50.87
COPA	78.00	(+2) **80.00**	MMLU	44.96	**45.14**
MNLI	45.98	(+7) **53.34**	ifeval	40.41	**41.13**
PIQA	75.79	**75.84**	BBH	44.63	(+3) **48.06**
Avg.				**54.59**	(+3) **57.16**

Table 2: Intersection-style for Qwen2.5-3B and truncated Qwen2.5-32B (Table 17).

Dataset	14B	32B	Potential	Dataset	14B	32B	Potential
gsm8k	**92.27**	91.13	91.81	QNLI	83.80	81.04	(+2) **85.43**
MATH500	81.60	77.40	**81.80**	RTE	77.62	75.45	(+1) **79.06**
HumanEval	67.07	68.90	(+5) **73.78**	Winogrande	72.77	**73.01**	72.61
MBPP	68.78	64.02	(+3) **71.69**	ARC	50.93	53.41	**53.52**
BoolQ	**89.24**	86.33	88.96	HellaSwag	66.19	**70.65**	70.61
COPA	90.00	94.00	(+2) **96.00**	MMLU	71.35	**76.21**	76.04
MNLI	67.06	69.89	(+1) **71.04**	ifeval	44.84	46.52	(+10) **56.35**
PIQA	79.76	81.50	**81.88**	BBH	85.00	86.39	(+2) **88.43**
Avg.					**74.27**	**74.74**	(+3) **77.44**

Table 3: Union-style for Qwen3-14B and Qwen3-32B (Table 23).

GSM8K (Cobbe et al., 2021), MATH (Hendrycks et al., 2021; Lightman et al., 2024), HumanEval and MBPP (Chen et al., 2021; Austin et al., 2021), IFEval (Zhou et al., 2023), GLUE-style NLU tasks (Williams et al., 2018; Wang et al., 2019), PIQA (Bisk et al., 2020), Winogrande (Sakaguchi et al., 2020), COPA (Roemmele et al., 2011), BoolQ (Clark et al., 2019), ARC (Clark et al., 2018), HellaSwag (Zellers et al., 2019), MMLU (Hendrycks et al., 2020), and BBH (Suzgun et al., 2022). We use the EleutherAI LM Evaluation Harness (Gao et al., 2023) and BigCode Evaluation Harness (Ben Allal et al., 2022). The benchmark families and model-pair configurations are summarized in Tables 1a and 1b.

Following Section 3, we interpolate after Truncate or Expand. For intersection-style merging, $\mu \in \{0.01, 0.02, 0.03, 0.04, 0.05\}$ weights the truncated larger checkpoint. For union-style merging, $\lambda \in \{0.02, 0.04, 0.96, 0.98\}$ weights the larger checkpoint. These are the operating ratios; each grid point is a real checkpoint. Main-text tables report per-task Potential on these grids; Appendix C reports the 16-task mean of each fixed ratio. Merging at coarser ratios still collapses performance, as one would generally expect. Appendix A illustrates this on GSM8K.

5 Main Results

On Qwen2.5-3B intersection, $\mu \in \{0.01, 0.02, 0.03, 0.04\}$ raises the 16-task mean from 54.59 to 55.54, 55.51, 55.14, and 55.23, and Potential reaches 57.16 (Table 2, Appendix C). For Qwen3-14B and Qwen3-32B, $\lambda \in \{0.02, 0.04, 0.96, 0.98\}$; the two sources score 74.27 and 74.74, $\lambda = 0.98$ reaches 76.10, and Potential reaches 77.44 (Table 3, Appendix C). The other five pairs move in the same direction, with smaller average gains. Intersection-style Potential rises from 70.44 to 71.61 for Qwen2.5-14B, from 75.08 to 75.91 for Qwen3-14B, and from 62.77 to 63.69 for Qwen3-4B-Thinking-2507 (Tables 4–6). Union-style Potential reaches 75.25 for Qwen2.5-14B and Qwen2.5-32B, above 70.44 and 74.24, and 70.16 for Qwen3-4B-Thinking-2507 and Qwen3-8B, above 62.71 and 68.48 (Tables 7 and 8).

5.1 Scale Gap and Non-Nested Knowledge

We evaluate intersection-style merging by interpolating Qwen2.5-3B with a Qwen2.5-32B checkpoint truncated to the 3B parameter space, $(1-\mu)$Qwen2.5-3B $+\mu$Qwen2.5-32B-truncated.

Dataset	14B	Potential	Dataset	14B	Potential
gsm8k	87.79	**88.48**	QNLI	67.71	(+9) **76.57**
MATH500	54.60	(+2) **56.40**	RTE	79.06	**79.78**
HumanEval	56.04	**56.16**	Winogrande	75.69	(+2) **77.74**
MBPP	67.38	**67.75**	ARC	**52.41**	52.00
BoolQ	85.32	**86.12**	HellaSwag	**69.11**	69.03
COPA	90.00	(+1) **91.00**	MMLU	**70.94**	70.88
MNLI	66.89	(+2) **68.80**	ifeval	**44.24**	43.88
PIQA	82.48	**82.54**	BBH	77.31	(+1) **78.70**
Avg.				**70.44**	(+1) **71.61**

Table 4: Intersection-style for Qwen2.5-14B after truncating Qwen2.5-32B (Table 18).

Dataset	14B	Potential	Dataset	14B	Potential
gsm8k	92.27	**93.86**	QNLI	83.80	**84.46**
MATH500	81.60	(+1) **82.80**	RTE	77.62	(+2) **79.78**
HumanEval	75.00	(+2) **76.83**	Winogrande	72.77	**73.40**
MBPP	73.81	**74.34**	ARC	50.93	**51.27**
BoolQ	89.24	**89.33**	HellaSwag	66.19	**66.83**
COPA	**90.00**	**90.00**	MMLU	**71.35**	71.29
MNLI	**67.06**	66.99	ifeval	44.84	(+2) **47.12**
PIQA	79.76	**80.25**	BBH	85.00	(+1) **86.02**
Avg.				**75.08**	(+1) **75.91**

Table 5: Intersection-style for Qwen3-14B after truncating Qwen3-32B (Table 19).

Dataset	4B	Potential	Dataset	4B	Potential
gsm8k	**88.63**	88.40	QNLI	52.66	(+6) **58.63**
MATH500	51.00	(+3) **53.80**	RTE	**81.23**	80.87
HumanEval	**52.20**	50.55	Winogrande	65.90	**67.48**
MBPP	**52.75**	51.93	ARC	**38.87**	38.72
BoolQ	74.77	(+3) **77.98**	HellaSwag	**51.52**	51.46
COPA	**82.00**	81.00	MMLU	58.54	**59.40**
MNLI	58.49	**58.87**	ifeval	46.52	(+1) **47.72**
PIQA	75.46	**75.63**	BBH	73.80	(+3) **76.57**
Avg.				**62.77**	(+1) **63.69**

Table 6: Intersection-style for Qwen3-4B-Thinking-2507 after truncating Qwen3-8B (Table 20).

Dataset	14B	32B	Potential	Dataset	14B	32B	Potential
gsm8k	87.79	90.14	**90.45**	QNLI	67.71	**85.96**	85.94
MATH500	54.60	**61.80**	61.40	RTE	79.06	**81.59**	81.23
HumanEval	56.04	61.65	(+3) **64.94**	Winogrande	75.69	75.69	(+2) **77.74**
MBPP	67.38	74.05	**74.13**	ARC	52.41	53.20	**53.41**
BoolQ	85.32	**87.40**	87.19	HellaSwag	69.11	**71.75**	71.56
COPA	90.00	87.00	(+1) **91.00**	MMLU	70.94	76.20	**76.30**
MNLI	66.89	72.17	**73.00**	ifeval	44.24	48.32	(+2) **50.72**
PIQA	82.48	82.43	**82.54**	BBH	77.31	78.52	(+4) **82.41**
Avg.				**70.44**	**74.24**	(+1) **75.25**

Table 7: Union-style for Qwen2.5-14B and Qwen2.5-32B (Table 21).

Although truncation is lossy, a small-ratio injection of the truncated 32B branch still yields a useful gain. A single μ already improves the 16-task mean (up to 55.54), and the Potential envelope in Tables 2–6 reaches 57.16 for Qwen2.5-3B and 71.61 for Qwen2.5-14B. The envelopes in Tables 4–6 are milder because their dimensional gaps are smaller than the 3B–32B gap in Table 2. Individual tasks still move by more than these averages. RTE on Qwen2.5-3B rises from 64.26 to 77.26 at $\mu = 0.04$ (Table 2), and QNLI on Qwen2.5-14B rises from 67.71 to 76.57 at $\mu = 0.05$ (Table 4). On

Dataset	4B	8B	Potential	Dataset	4B	8B	Potential
gsm8k	88.63	87.64	**90.07**	QNLI	52.66	78.24	**78.99**
MATH500	51.00	53.00	(+2) **54.80**	RTE	81.23	77.62	81.23
HumanEval	58.54	60.91	(+7) **67.68**	Winogrande	65.90	67.65	**68.51**
MBPP	45.50	63.28	**64.02**	ARC	38.87	46.40	46.10
BoolQ	74.77	86.73	**87.09**	HellaSwag	51.52	61.25	60.94
COPA	82.00	85.00	(+2) **87.00**	MMLU	58.54	66.95	**67.22**
MNLI	58.49	62.20	(+2) **63.87**	ifeval	46.52	42.33	(+2) **48.80**
PIQA	75.46	77.53	**77.75**	BBH	73.80	78.89	78.52
Avg.					**62.71**	**68.48**	(+2) **70.16**

Table 8: Union-style for Qwen3-4B-Thinking-2507 and Qwen3-8B (Table 22).

these pairs, intersection-style merging is more useful when the source models have a large capability gap, provided that the truncated branch remains a small perturbation.

Union-style merging improves in the other direction. For Qwen3-14B and Qwen3-32B, the two sources score 74.27 and 74.74, while the task-wise Potential envelope reaches 77.44 (Table 3). A larger parameter count does not mean the 32B checkpoint already contains everything the 14B checkpoint knows. The two models carry different knowledge structures, so a small-ratio injection of the expanded smaller model can still raise the accuracy of the larger one.

5.2 Task-Level Complementarity

Heterogeneous merging can help when the two source models are strong on different tasks. A larger checkpoint need not already contain the smaller model's knowledge, or lead on every capability: the two models carry different knowledge structures, so injecting a small weight of the expanded smaller model can still raise the larger one by combining their strengths. On GSM8K, Qwen3-4B-Thinking-2507 scores 88.63 and Qwen3-8B scores 87.64; injecting the expanded 4B checkpoint at weight 0.02 ($\lambda = 0.98$) reaches 90.07 (Table 22). On BBH, Qwen2.5-14B scores 77.31 and Qwen2.5-32B scores 78.52; injecting the expanded 14B checkpoint at weight 0.04 ($\lambda = 0.96$) reaches 82.41, consistent with the two models covering different items inside BBH (Table 21). On IFEval, Qwen3-14B scores 44.84 and Qwen3-32B scores 46.52; the same 0.02 and 0.04 injections ($\lambda = 0.98$ and $\lambda = 0.96$) reach 53.60 and 56.35 (Table 23). These gains sit in different benchmark families. This is a task-level complementarity effect, not a claim about aligned internal representations (Raghu et al., 2017; Kornblith et al., 2019).

6 Analysis

6.1 Analysis of Gain Patterns

For intersection-style merging, useful structure can survive truncation and be absorbed as a small perturbation (Table 2). This effect is stronger when the scale gap between the smaller and larger models is large.

For union-style merging, the gain is larger when the two sources are complementary rather than nearly nested. Qwen3-14B and Qwen3-32B are one such pair: the 32B checkpoint does not outperform the 14B checkpoint on every task. Each model is stronger on different benchmarks and carries a different knowledge structure (Table 3). Injecting the smaller model can therefore still improve the larger one.

6.2 Seesaw Effect

Heterogeneous merging is not monotonic. Appendices B and C report the full per-ratio results on the Qwen, Llama, and Gemma families. At one fixed ratio, gains on some tasks can come with drops on others. This is the familiar seesaw: different tasks and different models rarely share the same optimal configuration. Some benchmarks improve consistently, while others seesaw, and the strength of that seesaw depends on the model family. Qwen shows the most stable transfer of capability. Gemma and Llama improve on individual datasets, but the overall effect is milder.

6.3 Cross-Family Performance

Average gains concentrate in the Qwen family. The strongest are the Qwen2.5-3B intersection and the two Qwen3 union pairs. Intersection-style Potential raises Qwen2.5-3B from 54.59 to 57.16, and $\mu = 0.01$ already reaches 55.54 (Table 2). For Qwen3-14B and Qwen3-32B, the sources score 74.27 and 74.74, $\lambda = 0.98$ reaches 76.10, and Potential reaches 77.44 (Table 3). Potential for Qwen3-4B-Thinking-2507 and Qwen3-8B reaches 70.16, above 62.71 and 68.48 (Table 8).

Some benchmarks improve consistently, while others seesaw, and the strength of that seesaw depends on the model family. Qwen shows the most stable transfer of capability. Gemma and Llama improve on individual datasets, with a milder overall effect. On Llama-3.2-3B-Instruct, $\mu = 0.02$ raises MBPP from 33.60 to 37.30 and QNLI from 60.06 to 61.63 (Table 13). On Gemma-2-2B, $\mu = 0.002$ raises HumanEval from 12.20 to 15.24 and MBPP from 30.95 to 32.54, and $\mu = 0.02$ raises COPA from 81.00 to 83.00 (Table 14). HumanEval on Gemma-2-2B-IT rises from 31.10 to 32.93 when only the first 36 layers are mixed (α=0.002), and MATH500 on Llama-3.2-3B rises from 8.20 to 10.00 under the uniform MLP-only setting (α=0.0001) (Tables 16 and 15). The 16-task means move less: Llama-3.2-3B-Instruct goes from 57.29 to 56.86, Gemma-2-2B from 43.07 to 43.40, and the G0/G3 settings on Llama-3.2-3B/8B from 46.34 to 46.36 and 46.40 (Table 15). Gemma-2-2B-IT stays near 56.10 (56.01–56.17).

6.4 Perturbation Controls and Baseline Comparisons

We further test whether the gains can be explained by generic perturbation rather than cross-checkpoint transfer. Appendix A reports noise controls on a selection of settings where the gains are more pronounced. Noise sometimes improves individual tasks, but it does not match the merge scores on the reported tasks. For example, on RTE, the baseline, noise, and merge scores are 64.26, 65.70, and 72.56, respectively. On GSM8K, noise reaches 68.08, while the merge reaches 70.28. This indicates that the improvement is not fully explained by adding a non-checkpoint perturbation of comparable scale.

We also compare weighted averaging after truncation with several standard merging rules in the same small-model parameter space. The comparison is not intended to show that weighted averaging dominates all possible merging algorithms. Instead, it tests whether the simplest zero-training, zero-alignment rule remains competitive after dimensional adaptation. On the reported average score, Weight Averaging obtains 55.51, close to TIES at 55.87 and above SLERP, the 3B baseline, and DARE (Appendix A). Thus, dimensional adaptation should be viewed as a compatibility operator that can support simple averaging and may also support stronger or complementary merging rules.

7 Conclusion

We show that deterministic dimensional adaptation enables direct weighted averaging across heterogeneous LLM scales. A single fixed ratio can already improve the source checkpoint. Across Qwen-family pairs, small-ratio interpolation transfers complementary capabilities without training, adapters, routing, or semantic alignment. These results make simple weighted fusion a competitive zero-training baseline that more complex heterogeneous methods should beat rather than skip (Hitit et al., 2026; Yadav et al., 2025). Gains remain local under perturbation, seesaw, and family checks; automatic μ/λ selection is left to future work.

Limitations

The experiments measure accuracy on standard capability benchmarks and do not include safety, toxicity, or bias evaluations. Weighted averaging can carry forward undesirable behavior already present in a source checkpoint, and the merge itself adds no alignment step.

Ethical Considerations

Merged models may inherit unsafe or biased behavior from source checkpoints, so deployment requires safety checks.

Reproducibility Statement

The Truncate and Expand operators, interpolation formulas, and protocol-specific ratio grids are specified in Section 3 and Section 4, with tensor-level rules and pseudocode in Appendix D. Model pairs, benchmarks, and evaluation harnesses are listed in Tables 1a and 1b. Main-text tables report per-task Potential on those grids; Appendix C reports the 16-task mean of each fixed ratio. Diagnostic and cross-family results appear in Appendices A–B. All source checkpoints and benchmarks are publicly available; this work does not introduce a new dataset.

AI Use Statement

In this work, we used generative AI tools to edit the manuscript for readability, format tables and LaTeX, assist with English phrasing, search and organize related literature, and help draft some analysis text that interprets experimental trends. We have not used generative AI tools to generate synthetic datasets, invent theoretical claims, write proofs, design the merging operators, run the reported evaluations, or produce numerical results. All AI-assisted wording was checked against the source experiments and revised by the authors. We take responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI.

References

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, Yury Zemlyanskiy, Federico Lebron, and Sumit Sanghai. 2023. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4895–4901.

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. 2023. Git Re-Basin: Merging models modulo permutation symmetries. In *Proceedings of the 11th International Conference on Learning Representations*.

Jacob Austin, Augustus Odena, Maxwell Nye, Maarten Bosma, Henryk Michalewski, David Dohan, Ellen Jiang, Carrie Cai, Michael Terry, Quoc Le, and Charles Sutton. 2021. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Loubna Ben Allal, Niklas Muennighoff, Logesh Kumar Umapathi, Ben Lipkin, and Leandro von Werra. 2022. A framework for the evaluation of code generation models. `https://github.com/bigcode-project/bigcode-evaluation-harness`.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.

Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, and 1 others. 2021. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*.

Shilian Chen, Jie Zhou, Qin Chen, Wen Wu, Xin Li, Qi Feng, and Liang He. 2026. Can heterogeneous language models be fused? *arXiv preprint arXiv:2604.01674*.

Zhijun Chen, Xiaodong Lu, Jingzheng Li, Pengpeng Chen, Zhuoran Li, Kai Sun, Yuankai Luo, Qianren Mao, Ming Li, Likang Xiao, Dingqi Yang, Xiao Huang, Yikun Ban, Hailong Sun, and Philip S. Yu. 2025. Harnessing multiple large language models: A survey on LLM ensemble. *arXiv preprint arXiv:2502.18036*.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. 2019. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

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

Guodong Du, Zhuo Li, Xuanning Zhou, Junlin Li, Zesheng Shi, Wanyu Lin, Ho-Kin Tang, Xiucheng Li, Fangming Liu, Wenya Wang, Min Zhang, and Jing Li. 2025a. Knowledge fusion of large language models via modular SkillPacks. *arXiv preprint arXiv:2505.18502*.

Yiyang Du, Xiaochen Wang, Chi Chen, Jiabo Ye, Yiru Wang, Peng Li, Ming Yan, Ji Zhang, Fei Huang, Zhifang Sui, Maosong Sun, and Yang Liu. 2025b. AdaMMS: Model merging for heterogeneous multimodal large language models with unsupervised coefficient optimization. *arXiv preprint arXiv:2503.23733*.

Jonathan Frankle, Gintare Karolina Dziugaite, Daniel M. Roy, and Michael Carbin. 2020. Linear mode connectivity and the lottery ticket hypothesis. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3259–3269.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, and 5 others. 2023. A framework for few-shot language model evaluation.

Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry P. Vetrov, and Andrew Gordon Wilson. 2018. Loss surfaces, mode connectivity, and fast ensembling of DNNs. In *Advances in Neural Information Processing Systems*, volume 31.

Jian Gu, Aldeida Aleti, Chunyang Chen, and Hongyu Zhang. 2025. SeMe: Training-free language model merging via semantic alignment. *arXiv preprint arXiv:2505.20144*.

Stefan Hackmann. 2024. HM3: Heterogeneous multi-class model merging. *arXiv preprint arXiv:2409.19173*.

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

Yifei He, Siqi Zeng, Yuzheng Hu, Rui Yang, Tong Zhang, and Han Zhao. 2025. MergeBench: A benchmark for merging domain-specialized LLMs. In *Advances in Neural Information Processing Systems*.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basu, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the MATH dataset. *arXiv preprint arXiv:2103.03874*.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2015. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*.

Oğuz Kağan Hitit, Leander Girrbach, and Zeynep Akata. 2026. A systematic study of in-the-wild model merging for large language models. *Transactions on Machine Learning Research*.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. 2023. Editing models with task arithmetic. In *Proceedings of the 11th International Conference on Learning Representations*.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. 2023. Dataless knowledge fusion by merging weights of language models. In *Proceedings of the 11th International Conference on Learning Representations*.

Keller Jordan, Hanie Sedghi, Olga Saukh, Rahim Entezari, and Behnam Neyshabur. 2023. REPAIR: Renormalizing permuted activations for interpolation repair. In *Proceedings of the 11th International Conference on Learning Representations*.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. 2019. Similarity of neural network representations revisited. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 3519–3529.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *Proceedings of the 12th International Conference on Learning Representations*.

Jinliang Lu, Ziliang Pang, Min Xiao, Yaochen Zhu, Rui Xia, and Jiajun Zhang. 2024. Merge, ensemble, and cooperate! A survey on collaborative strategies in the era of large language models. *arXiv preprint arXiv:2407.06089*.

Michael Matena and Colin Raffel. 2022. Merging models with fisher-weighted averaging. In *Advances in Neural Information Processing Systems*, volume 35.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 487–503.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. 2017. SVCCA: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. In *Advances in Neural Information Processing Systems*, volume 30.

Melissa Roemmele, Cosmin Adrian Bejan, and Andrew S. Gordon. 2011. Choice of plausible alternatives: An evaluation of commonsense causal reasoning. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. WinoGrande: An adversarial Winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8732–8740.

Ken Shoemake. 1985. Animating rotation with quaternion curves. In *Proceedings of the 12th Annual Conference on Computer Graphics and Interactive Techniques*, pages 245–254.

Mingyang Song and Mao Zheng. 2026. Model merging in the era of large language models: Methods, applications, and future directions. *arXiv preprint arXiv:2603.09938*.

Bedionita Soro, Aoxuan Silvia Zhang, Bruno Andreis, Jaehyeong Jo, Song Chong, and Sung Ju Hwang. 2026. LS-Merge: Merging language models in latent space. In *Proceedings of the 14th International Conference on Learning Representations*.

George Stoica, Daniel Bolya, Jakob Bjorner, Pratik Ramesh, Taylor Hearn, and Judy Hoffman. 2024. ZipIt! merging models from different tasks without training. In *Proceedings of the 12th International Conference on Learning Representations*.

Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V. Le, Ed H. Chi, Denny Zhou, and Jason Wei. 2022. Challenging BIG-Bench tasks and whether chain-of-thought can solve them. *arXiv preprint arXiv:2210.09261*.

Derek Tam, Mohit Bansal, and Colin Raffel. 2024. Merging by matching models in task parameter subspaces. *Transactions on Machine Learning Research*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30.

Fanqi Wan, Xinting Huang, Deng Cai, Xiaojun Quan, Wei Bi, and Shuming Shi. 2024a. Knowledge fusion of large language models. *arXiv preprint arXiv:2401.10491*.

Fanqi Wan, Longguang Zhong, Ziyi Yang, Ruijun Chen, and Xiaojun Quan. 2024b. FuseChat: Knowledge fusion of chat models. *arXiv preprint arXiv:2408.07990*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*.

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

Yuanyi Wang, Yanggan Gu, Yiming Zhang, Qi Zhou, Zhaoyi Yan, Congkai Xie, Xinyao Wang, Jianbo Yuan, and Hongxia Yang. 2025. Model merging scaling laws in large language models. *arXiv preprint arXiv:2509.24244*.

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

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. 2022. 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*, pages 23965–23998.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. 2023. TIES-Merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36.

Prateek Yadav, Tu Vu, Jonathan Lai, Alexandra Chronopoulou, Manaal Faruqui, Mohit Bansal, and Tsendsuren Munkhdalai. 2025. What matters for model merging at scale? *arXiv preprint arXiv:2410.03617*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Qwen Team. 2025. Qwen3-4B-Thinking-2507. Hugging Face model card.

An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, and 1 others. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

Enneng Yang, Li Shen, Guibing Guo, Xingwei Wang, Xiaochun Cao, Jie Zhang, and Dacheng Tao. 2026. Model merging in LLMs, MLLMs, and beyond: Methods, theories, applications, and opportunities. *ACM Computing Surveys*, 58(8).

Le Yu, Bowen Yu, Haiyang Yu, Fei Huang, and Yongbin Li. 2024. Language models are Super Mario: Absorbing abilities from homologous models as a free lunch. In *Proceedings of the 41st International Conference on Machine Learning*.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. 2019. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800.

Biao Zhang and Rico Sennrich. 2019. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, volume 32.

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

A Additional Diagnostics

A.1 Ablation Controls and Baseline Comparisons

We add two diagnostic comparisons to separate cross-checkpoint transfer from simpler alternative explanations. Table 9 reports noise controls on a selection of settings where the gains are more pronounced. Table 10 compares weighted averaging after truncation with representative merging rules evaluated in the same small-model parameter space.

Setting	Task	Baseline	Noise	Merge
Intersection	gsm8k	65.66	68.08	**70.28**
Intersection	MBPP	58.99	57.67	**60.58**
Intersection	MNLI	45.98	46.91	**49.19**
Intersection	PIQA	75.79	75.30	**75.84**
Intersection	RTE	64.26	65.70	**72.56**
Intersection	MMLU	44.96	44.99	**45.14**
Union	IFEval	46.52	48.20	**53.60**

Table 9: Perturbation controls on a selection of settings where the merge gains are more pronounced. Noise tests whether those gains can be explained by non-checkpoint perturbations of comparable scale.

TIES	Weight Averaging	SLERP	3B baseline	DARE
55.87	55.51	54.74	54.59	51.10

Table 10: Baseline comparison in the small-model parameter space. Weighted averaging is not claimed to dominate all merging rules, but remains competitive as a zero-training, zero-alignment baseline after truncation.

A.2 GSM8K Ratio Sensitivity

Tables 11 and 12 show that coarser-ratio merges collapse on GSM8K. Intersection: μ weights the truncated 32B checkpoint (Truncate alone scores 1.14), so large μ fails. Union: Eq. (4) stays near a source at the endpoints; a balanced mix fails.

Model/Ratio	Flexible		Strict	
	Mean	Var	Mean	Var
Qwen2.5-32B	90.14	0.82	80.21	1.10
Qwen2.5-14B	87.79	0.90	73.39	1.22
Truncate	1.14	0.29	0.00	0.00
$\mu = 0.1$	81.73	1.06	74.45	1.20
$\mu = 0.2$	21.76	1.14	21.46	1.13
$\mu = 0.3$	2.20	0.40	0.00	0.00
$\mu = 0.4$	1.59	0.34	0.00	0.00
$\mu = 0.5$	0.76	0.24	0.00	0.00
$\mu = 0.6$	0.00	0.00	0.00	0.00
$\mu = 0.7$	0.00	0.00	0.00	0.00
$\mu = 0.8$	0.53	0.20	0.00	0.00
$\mu = 0.9$	0.53	0.20	0.00	0.00

Table 11: GSM8K, intersection. Truncate is the truncated 32B checkpoint (Flexible 1.14).

Model/Ratio	Flexible		Strict	
	Mean	Var	Mean	Var
Qwen2.5-32B	90.14	0.82	80.21	1.10
Qwen2.5-14B	87.79	0.90	73.39	1.22
Expand	88.02	0.89	73.46	1.22
$\lambda = 0.1$	80.59	1.09	72.10	1.24
$\lambda = 0.2$	7.81	0.74	1.97	0.38
$\lambda = 0.3$	1.44	0.33	0.00	0.00
$\lambda = 0.4$	0.45	0.19	0.00	0.00
$\lambda = 0.5$	1.21	0.30	0.00	0.00
$\lambda = 0.6$	2.27	0.41	0.23	0.13
$\lambda = 0.7$	44.96	1.37	29.19	1.25
$\lambda = 0.8$	85.97	0.96	62.85	1.33
$\lambda = 0.9$	90.45	0.81	67.85	1.29

Table 12: GSM8K, union. Endpoints stay near a source; mid-range mixes fail.

B Additional Llama and Gemma Results

Task	Llama-3B	μ=0.002	μ=0.01	μ=0.02	μ=0.03	μ=0.04	μ=0.05
gsm8k	77.10	76.88	77.03	75.36	74.37	70.58	62.47
MATH500	40.20	38.60	39.60	35.00	33.00	33.00	25.80
HumanEval	32.93	33.54	31.71	28.05	28.66	28.66	19.51
MBPP	33.60	31.22	32.28	37.30	34.39	26.19	20.37
BoolQ	77.31	77.13	77.89	78.53	78.10	78.17	77.22
COPA	82.00	80.00	80.00	80.00	80.00	75.00	77.00
MNLI	44.73	44.64	45.33	44.77	43.27	41.17	38.16
PIQA	76.55	76.44	76.82	76.71	76.12	75.79	74.92
QNLI	60.06	60.06	61.43	61.63	60.83	61.67	60.53
RTE	74.01	73.65	73.65	72.92	71.84	72.20	71.84
Winogrande	66.22	65.59	65.27	62.83	62.67	62.83	62.04
ARC	35.21	35.45	35.56	34.87	33.63	31.37	28.33
HellaSwag	50.48	50.54	50.18	49.22	47.60	44.55	40.36
MMLU	33.93	34.25	34.01	33.95	34.36	32.60	29.87
ifeval	82.85	82.01	83.09	81.77	81.18	81.89	79.86
BBH	49.44	49.44	45.93	40.65	34.07	29.72	24.54
Avg.	**57.29**	**56.84**	**56.86**	**55.85**	**54.63**	**52.84**	**49.55**

Table 13: Task-level outcomes for the Llama-3.2-3B-Instruct and Llama-3.1-70B-Instruct model pair under intersection-style merging.

Task	Gemma-2B	μ=0.002	μ=0.005	μ=0.01	μ=0.02	μ=0.03	μ=0.04	μ=0.05
gsm8k	25.85	26.61	26.46	26.00	26.00	25.32	24.03	24.56
MATH500	16.80	17.80	18.00	17.60	18.00	16.60	16.60	14.00
HumanEval	12.20	15.24	14.63	14.63	12.80	12.20	14.02	12.80
MBPP	30.95	32.54	29.37	28.04	28.04	25.40	28.84	25.93
BoolQ	66.42	66.12	65.63	65.87	65.20	65.38	64.37	64.19
COPA	81.00	81.00	81.00	82.00	83.00	81.00	79.00	79.00
MNLI	35.45	35.45	35.44	35.44	35.45	35.45	35.45	35.45
PIQA	75.73	75.68	75.57	75.46	75.14	74.76	74.43	74.76
QNLI	49.86	50.34	50.14	49.97	49.99	49.77	49.75	49.44
RTE	53.43	53.79	53.43	53.43	53.07	52.71	52.71	52.71
Winogrande	66.54	65.59	64.96	65.98	65.67	63.77	63.22	63.22
ARC	33.01	32.73	32.84	32.65	32.31	32.14	31.67	31.26
HellaSwag	52.00	51.90	51.90	51.79	51.55	51.27	50.89	50.57
MMLU	22.74	22.73	22.73	22.73	22.73	22.73	22.73	22.72
ifeval	26.38	25.78	26.26	26.26	25.78	24.58	24.58	24.58
BBH	40.74	41.02	40.74	39.72	39.54	38.33	36.94	33.61
Avg.	**43.07**	**43.40**	**43.07**	**42.97**	**42.77**	**41.96**	**41.83**	**41.17**

Table 14: Task-level outcomes for the Gemma-2-2B and Gemma-2-9B model pair under intersection-style merging.

Task	3B	8B	G0	G3
gsm8k	30.10	56.25	29.95	29.80
MATH500	8.20	21.40	9.00	10.00
HumanEval	21.34	28.05	21.34	21.34
MBPP	40.48	52.12	39.15	39.68
BoolQ	73.55	82.94	73.55	73.61
COPA	82.00	91.00	83.00	82.00
MNLI	34.76	52.64	34.83	34.72
PIQA	78.13	80.74	78.18	77.58
QNLI	49.86	49.75	49.72	49.84
RTE	54.87	71.12	54.51	54.51
Winogrande	69.30	74.27	69.85	69.61
ARC	37.35	45.09	37.20	37.16
HellaSwag	55.21	62.69	55.26	55.28
MMLU	46.50	54.05	46.53	46.58
ifeval	12.23	14.87	12.59	12.59
BBH	47.50	65.37	47.04	48.15
Avg.	**46.34**	**56.40**	**46.36**	**46.40**

Table 15: Llama-3.2-3B and Llama-3.1-8B under intersection-style merging. G0: first-36 layers, α=0.0002. G3: uniform, MLP-only, α=0.0001.

Task	2B-IT	9B-IT	G0	G1
gsm8k	53.22	77.48	53.68	54.21
MATH500	21.80	35.80	20.60	21.20
HumanEval	31.10	10.37	32.93	32.32
MBPP	42.86	48.41	42.06	43.39
BoolQ	82.63	88.59	82.75	82.69
COPA	85.00	88.00	84.00	85.00
MNLI	60.54	69.48	60.70	60.61
PIQA	77.37	80.03	77.69	77.58
QNLI	71.41	85.81	71.52	71.76
RTE	78.70	79.42	78.34	78.70
Winogrande	64.80	71.74	64.25	64.72
ARC	36.16	47.30	36.13	36.13
HellaSwag	47.60	60.04	47.65	47.67
MMLU	26.40	31.58	26.35	26.36
ifeval	69.90	82.37	69.42	68.71
BBH	48.06	66.48	48.06	47.59
Avg.	**56.10**	**63.93**	**56.01**	**56.17**

Table 16: Gemma-2-2B-IT and Gemma-2-9B-IT under intersection-style merging. G0: first-36 layers, α=0.002. G1: uniform, α=0.002.

C Additional Task-Level Results

The BEST column is the same per-task envelope reported as Potential in the main text. Each μ or λ column is one checkpoint; the Avg. row is that checkpoint's 16-task mean. Tables 17–20 report the per-task scores at each fixed intersection ratio for Qwen2.5-3B, Qwen2.5-14B, Qwen3-14B, and Qwen3-4B-Thinking-2507.

On HumanEval and MBPP, intersection-style tables use the smaller base model's decoding setup and union-style tables use the larger base model's, because these code tasks typically need model-specific prompts and generation lengths. Qwen3 pairs use base-specific setups and should not be compared across intersection versus union. Qwen2.5-14B and Qwen2.5-32B share one setup, so those source scores match across tables. Within each table, the source column and the merged columns use the same protocol.

Dataset	**3B**	$\mu = 0.01$	$\mu = 0.02$	$\mu = 0.03$	$\mu = 0.04$	$\mu = 0.05$	**BEST**	**WORST**
gsm8k	65.66	**68.39**	**70.28**	**71.42**	**72.10**	**70.43**	(+6pp) **72.10**	**68.39**
MATH500	38.40	**40.60**	38.40	35.60	36.60	35.60	(+2pp) **40.60**	35.60
HumanEval	43.90	43.29	36.71	32.32	33.54	29.27	43.29	29.27
MBPP	58.99	58.73	**60.58**	58.47	57.94	**59.26**	(+2pp) **60.58**	57.94
BoolQ	70.80	**70.89**	70.40	**71.16**	**72.54**	**72.72**	(+2pp) **72.72**	70.40
COPA	78.00	**79.00**	**79.00**	**79.00**	**80.00**	**80.00**	(+2pp) **80.00**	**79.00**
MNLI	45.98	**48.17**	**49.19**	**50.37**	**52.03**	**53.34**	(+7pp) **53.34**	**48.17**
PIQA	75.79	75.63	**75.84**	74.81	74.59	74.05	**75.84**	74.05
QNLI	52.86	**53.45**	52.79	**53.78**	**54.51**	**55.26**	(+2pp) **55.26**	52.79
RTE	64.26	**69.31**	**72.56**	**75.81**	**77.26**	**74.37**	(+13pp) **77.26**	**69.31**
Winogrande	63.38	**64.01**	63.22	**63.77**	62.12	62.35	(+1pp) **64.01**	62.12
ARC	34.42	34.30	34.21	33.72	33.61	33.57	34.30	33.57
HellaSwag	50.94	50.87	50.68	50.27	50.03	49.49	50.87	49.49
MMLU	44.96	**45.04**	**45.14**	44.07	43.23	42.41	**45.14**	42.41
ifeval	40.41	39.57	**41.13**	**40.65**	38.61	38.25	(+1pp) **41.13**	38.25
BBH	44.63	**47.31**	**48.06**	**46.94**	**44.91**	43.06	(+3pp) **48.06**	43.06
Avg.	54.59	**55.54**	**55.51**	**55.14**	**55.23**	54.59	(+3pp) **57.16**	53.36

Table 17: Task-level best and worst merging outcomes for Qwen2.5-3B under small-ratio intersection-style merging.

Dataset	**14B**	$\mu = 0.01$	$\mu = 0.02$	$\mu = 0.03$	$\mu = 0.04$	$\mu = 0.05$	**BEST**	**WORST**
gsm8k	87.79	87.72	**88.48**	**88.17**	**88.10**	87.19	**88.48**	87.19
MATH500	54.60	**56.40**	**55.20**	**55.20**	54.60	54.00	(+2pp) **56.40**	54.00
HumanEval	56.04	**56.16**	55.79	55.61	54.94	53.78	**56.16**	53.78
MBPP	67.38	**67.75**	**67.67**	66.40	66.24	65.42	**67.75**	65.42
BoolQ	85.32	**85.84**	**86.12**	**85.81**	**85.60**	85.17	**86.12**	85.17
COPA	90.00	90.00	**91.00**	**91.00**	**91.00**	90.00	(+1pp) **91.00**	90.00
MNLI	66.89	**67.67**	**68.01**	**68.36**	**68.64**	**68.80**	(+2pp) **68.80**	**67.67**
PIQA	82.48	82.32	**82.54**	82.32	82.37	81.77	**82.54**	81.77
QNLI	67.71	**69.43**	**70.31**	**72.62**	**73.86**	**76.57**	(+9pp) **76.57**	**69.43**
RTE	79.06	**79.78**	**79.78**	79.06	79.06	79.06	**79.78**	79.06
Winogrande	75.69	**76.64**	**76.01**	**77.11**	**77.74**	**77.27**	(+2pp) **77.74**	**76.01**
ARC	52.41	52.00	51.44	51.23	51.06	50.50	52.00	50.50
HellaSwag	69.11	69.03	68.95	68.59	68.45	68.05	69.03	68.05
MMLU	70.94	70.87	70.88	70.79	70.73	70.47	70.88	70.47
ifeval	44.24	43.88	43.53	42.69	42.81	41.85	43.88	41.85
BBH	77.31	76.94	76.94	**77.59**	**78.52**	**78.70**	(+1pp) **78.70**	76.94
Avg.	70.44	**70.78**	**70.79**	**70.78**	**70.86**	**70.54**	(+1pp) **71.61**	69.83

Table 18: Task-level best and worst merging outcomes for Qwen2.5-14B under small-ratio intersection-style merging.

Dataset	14B	$\mu = 0.01$	$\mu = 0.02$	$\mu = 0.03$	$\mu = 0.04$	$\mu = 0.05$	BEST	WORST
gsm8k	92.27	**93.63**	**93.78**	**93.86**	**93.48**	92.42	**93.86**	92.42
MATH500	81.60	81.60	**82.20**	**82.80**	80.60	81.00	(+1pp) **82.80**	80.60
HumanEval	75.00	**76.22**	75.00	**75.61**	**75.61**	**76.83**	(+2pp) **76.83**	75.00
MBPP	73.81	73.54	**74.34**	73.81	**74.07**	73.54	**74.34**	73.54
BoolQ	89.24	**89.33**	89.20	89.02	89.02	88.81	**89.33**	88.81
COPA	90.00	90.00	90.00	90.00	89.00	88.00	90.00	88.00
MNLI	67.06	66.99	66.53	66.33	65.69	65.33	66.99	65.33
PIQA	79.76	**80.09**	**79.82**	**80.25**	**79.98**	**79.82**	**80.25**	**79.82**
QNLI	83.80	83.76	**84.02**	**84.20**	**84.31**	**84.46**	**84.46**	83.76
RTE	77.62	**78.34**	**78.34**	**79.42**	**79.42**	**79.78**	(+2pp) **79.78**	**78.34**
Winogrande	72.77	**73.16**	**73.32**	**73.40**	72.69	**73.40**	**73.40**	72.69
ARC	50.93	**50.95**	**51.21**	**51.27**	**51.12**	**51.19**	**51.27**	**50.95**
HellaSwag	66.19	**66.42**	**66.57**	**66.61**	**66.66**	**66.83**	**66.83**	**66.42**
MMLU	71.35	71.29	71.17	71.05	70.95	70.61	71.29	70.61
ifeval	44.84	**46.64**	**47.00**	**46.16**	44.60	**47.12**	(+2pp) **47.12**	44.60
BBH	85.00	83.70	84.26	**85.46**	**86.02**	80.74	(+1pp) **86.02**	80.74
Avg.	75.08	**75.35**	**75.42**	**75.58**	**75.20**	74.99	(+1pp) **75.91**	74.48

Table 19: Task-level best and worst merging outcomes for Qwen3-14B under small-ratio intersection-style merging.

Dataset	4B	$\mu = 0.01$	$\mu = 0.02$	$\mu = 0.03$	$\mu = 0.04$	$\mu = 0.05$	BEST	WORST
gsm8k	88.63	87.72	88.40	88.17	88.40	87.49	88.40	87.49
MATH500	51.00	**52.40**	**51.20**	**53.80**	**51.40**	**52.00**	(+3pp) **53.80**	**51.20**
HumanEval	52.20	50.55	47.44	44.76	41.22	39.33	50.55	39.33
MBPP	52.75	51.93	50.79	49.42	48.78	44.52	51.93	44.52
BoolQ	74.77	**76.36**	**76.64**	**77.58**	**77.49**	**77.98**	(+3pp) **77.98**	**76.36**
COPA	82.00	81.00	80.00	79.00	78.00	78.00	81.00	78.00
MNLI	58.49	**58.79**	**58.87**	**58.81**	**58.55**	58.28	**58.87**	58.28
PIQA	75.46	75.30	75.24	**75.63**	**75.57**	75.41	**75.63**	75.24
QNLI	52.66	**53.47**	**54.16**	**55.99**	**57.18**	**58.63**	(+6pp) **58.63**	**53.47**
RTE	81.23	80.87	80.14	80.14	80.87	80.87	80.87	80.14
Winogrande	65.90	**66.22**	**66.46**	**67.40**	**67.48**	**66.93**	(+2pp) **67.48**	**66.22**
ARC	38.87	38.72	38.42	38.40	38.19	37.91	38.72	37.91
HellaSwag	51.52	51.46	51.36	50.94	50.81	50.36	51.46	50.36
MMLU	58.54	**59.23**	**59.40**	**59.19**	**59.12**	**59.08**	**59.40**	**59.08**
ifeval	46.52	**47.00**	45.92	**47.72**	**47.00**	**47.48**	(+1pp) **47.72**	45.92
BBH	73.80	**75.74**	**75.74**	**75.74**	**76.57**	**75.19**	(+3pp) **76.57**	**75.19**
Avg.	62.77	**62.92**	62.51	62.67	62.29	61.84	(+1pp) **63.69**	61.17

Table 20: Task-level best and worst merging outcomes for Qwen3-4B-Thinking-2507 under small-ratio intersection-style merging.

Dataset	14B	32B	$\lambda = 0.98$	$\lambda = 0.96$	$\lambda = 0.04$	$\lambda = 0.02$	BEST	WORST
gsm8k	87.79	90.14	**90.45**	89.92	87.87	88.32	**90.45**	87.87
MATH500	54.60	61.80	61.40	58.60	53.80	55.60	61.40	53.80
HumanEval	56.04	61.65	**64.82**	**64.94**	56.28	55.91	(+3pp) **64.94**	55.91
MBPP	67.38	74.05	**74.13**	73.92	67.91	68.33	**74.13**	67.91
BoolQ	85.32	87.40	87.19	86.82	85.54	85.90	87.19	85.54
COPA	90.00	87.00	89.00	89.00	90.00	**91.00**	(+1pp) **91.00**	89.00
MNLI	66.89	72.17	**72.57**	**73.00**	68.69	68.23	**73.00**	68.23
PIQA	82.48	82.43	82.32	82.43	**82.54**	82.26	**82.54**	82.26
QNLI	67.71	85.96	85.87	85.94	74.28	70.40	85.94	70.40
RTE	79.06	81.59	81.23	81.23	78.70	79.06	81.23	78.70
Winogrande	75.69	75.69	75.69	**76.87**	**77.74**	**76.09**	(+2pp) **77.74**	75.69
ARC	52.41	53.20	**53.28**	**53.41**	50.99	51.44	**53.41**	50.99
HellaSwag	69.11	71.75	71.56	71.16	68.44	68.93	71.56	68.44
MMLU	70.94	76.20	**76.30**	**76.23**	70.73	70.91	**76.30**	70.73
ifeval	44.24	48.32	**48.80**	**50.72**	43.17	44.12	(+2pp) **50.72**	43.17
BBH	77.31	78.52	**82.31**	**82.41**	78.15	77.04	(+4pp) **82.41**	77.04
Avg.	70.44	74.24	**74.81**	**74.79**	70.93	70.85	**75.25**	70.36

Table 21: Task-level best/worst outcomes for Qwen2.5-14B and Qwen2.5-32B under union-style merging.

Dataset	4B	8B	$\lambda = 0.98$	$\lambda = 0.96$	$\lambda = 0.04$	$\lambda = 0.02$	BEST	WORST
gsm8k	88.63	87.64	**90.07**	87.49	86.05	89.23	**90.07**	86.05
MATH500	51.00	53.00	**54.80**	**54.60**	51.80	46.80	(+2pp) **54.80**	46.80
HumanEval	58.54	60.91	**67.68**	**66.46**	50.61	59.15	(+7pp) **67.68**	50.61
MBPP	45.50	63.28	62.96	**64.02**	44.97	44.44	**64.02**	44.44
BoolQ	74.77	86.73	**86.82**	**87.09**	75.69	75.38	**87.09**	75.38
COPA	82.00	85.00	**87.00**	**86.00**	79.00	80.00	(+2pp) **87.00**	79.00
MNLI	58.49	62.20	**62.98**	**63.87**	58.49	59.30	(+2pp) **63.87**	58.49
PIQA	75.46	77.53	**77.75**	77.53	74.81	75.14	**77.75**	74.81
QNLI	52.66	78.24	**78.99**	**78.42**	56.01	54.35	**78.99**	54.35
RTE	81.23	77.62	77.62	78.34	81.23	79.78	81.23	77.62
Winogrande	65.90	67.65	**68.27**	**68.51**	65.51	65.67	**68.51**	65.51
ARC	38.87	46.40	46.10	46.01	38.49	38.68	46.10	38.49
HellaSwag	51.52	61.25	60.94	60.50	50.39	50.86	60.94	50.39
MMLU	58.54	66.95	**67.22**	**67.16**	59.21	59.18	**67.22**	59.18
ifeval	46.52	42.33	40.77	37.77	**48.80**	**48.68**	(+2pp) **48.80**	37.77
BBH	73.80	78.89	78.33	78.52	75.37	75.09	78.52	75.09
Avg.	62.71	68.48	**69.27**	**68.89**	62.28	62.61	(+2pp) **70.16**	60.87

Table 22: Task-level best/worst outcomes for Qwen3-4B-Thinking-2507 and Qwen3-8B under union-style merging.

Dataset	14B	32B	$\lambda = 0.98$	$\lambda = 0.96$	$\lambda = 0.04$	$\lambda = 0.02$	BEST	WORST
gsm8k	92.27	91.13	91.81	88.48	91.21	91.81	91.81	88.48
MATH500	81.60	77.40	77.40	75.80	79.20	**81.80**	**81.80**	75.80
HumanEval	67.07	68.90	**73.78**	68.90	66.46	65.85	(+5pp) **73.78**	65.85
MBPP	68.78	64.02	65.87	**69.84**	**71.69**	**71.16**	(+3pp) **71.69**	65.87
BoolQ	89.24	86.33	87.52	87.89	88.87	88.96	88.96	87.52
COPA	90.00	94.00	94.00	**96.00**	88.00	89.00	(+2pp) **96.00**	88.00
MNLI	67.06	69.89	**71.04**	69.51	69.18	68.47	(+1pp) **71.04**	**68.47**
PIQA	79.76	81.50	**81.88**	80.25	80.47	80.09	**81.88**	**80.09**
QNLI	83.80	81.04	83.65	**85.15**	**85.43**	**84.68**	(+2pp) **85.43**	83.65
RTE	77.62	75.45	75.81	76.53	**79.06**	**78.70**	(+1pp) **79.06**	**75.81**
Winogrande	72.77	73.01	72.61	71.90	72.22	72.38	72.61	71.90
ARC	50.93	53.41	**53.52**	52.45	51.78	51.60	**53.52**	51.60
HellaSwag	66.19	70.65	70.61	67.82	66.78	66.55	70.61	66.55
MMLU	71.35	76.21	76.04	72.48	70.99	71.29	76.04	70.99
ifeval	44.84	46.52	**53.60**	**56.35**	45.80	**46.88**	(+10pp) **56.35**	**45.80**
BBH	85.00	86.39	**88.43**	85.93	85.83	86.20	(+2pp) **88.43**	85.83
Avg.	74.27	74.74	**76.10**	**75.33**	74.56	74.71	(+3pp) **77.44**	73.26

Table 23: Task-level best/worst outcomes for Qwen3-14B and Qwen3-32B under union-style merging.

D DIMENSIONAL ADAPTATION DETAILS

This appendix gives the implementation-level form of the two dimensional adaptation operators used in Section 3. We separate the operator that projects a larger checkpoint into a smaller architecture, denoted by TRUNCATE, from the operator that maps a smaller checkpoint into a larger architecture, denoted by EXPAND. These operators correspond to intersection-style merging and union-style merging, respectively. They are deterministic tensor maps applied before interpolation, not learned alignment modules.

D.1 INTERSECTION-STYLE TRUNCATE

For intersection-style merging, we instantiate the small-model architecture and project the large checkpoint into that state dictionary. Let $W_B \in \mathbb{R}^{s_1 \times \cdots \times s_r}$ be a tensor from the larger checkpoint and let $W_A \in \mathbb{R}^{t_1 \times \cdots \times t_r}$ be the corresponding tensor shape in the smaller architecture. The truncation operator copies the overlapping prefix region:

$$\big[\mathcal{T}(W_B)\big]_{i_1,\ldots,i_r} = \big[W_B\big]_{i_1,\ldots,i_r}, \qquad 0 \le i_j < \min(s_j, t_j). \tag{6}$$

Dimensions outside the target shape are discarded. Identical-shape tensors are copied directly. Layers whose indices exceed the small model depth are removed. For MLP matrices, this corresponds to retaining the target intermediate width: gate and up projections are sliced on their output dimension, while down projections are sliced on their input-intermediate dimension. For tied embeddings, the output head follows the target small-model tying convention.

Pseudocode 1: TRUNCATE for intersection-style merging

Input: large checkpoint θ_B, small architecture $\mathcal{A}_A$, small tokenizer/config metadata.
Output: projected checkpoint $\tilde{\theta}_B = \mathcal{T}(\theta_B)$ in the small parameter space.

1 Initialize a model with the small architecture $\mathcal{A}_A$ and read its target state-dictionary keys and shapes.
2 For each target tensor key, find the tensor with the same key in the large checkpoint.
3 If the source and target shapes are identical, copy the source tensor with the target dtype.
4 If the ranks match but some dimensions are larger in the source tensor, copy the overlapping prefix slice into the target tensor.
5 If a large-model layer index is beyond the target depth, discard that layer.
6 If a target key is tied to the embedding matrix, reuse the target embedding tensor according to the small-model convention.
7 Save the projected checkpoint and tokenizer metadata from the small architecture.
8 Use the projection only as a perturbation in $\theta^{\cap}(\mu) = (1-\mu)\theta_A + \mu\mathcal{T}(\theta_B)$ with small μ.

Table 24: TRUNCATE projects the large checkpoint into the small-model parameter space by same-key copying, prefix slicing of oversized tensors, and layer removal.

Truncation is intentionally lossy, and the merge uses the projected checkpoint as a perturbation of θ_A. The interpolated checkpoint can be written as

$$\theta^{\cap}(\mu) = \theta_A + \mu\big(\mathcal{T}(\theta_B) - \theta_A\big), \tag{7}$$

so the residual stream, normalization scales, and attention computations remain anchored by θ_A when μ is small. If the local input-output map around θ_A is approximated by a locally Lipschitz function F, then

$$\|F(\theta^{\cap}(\mu)) - F(\theta_A)\| \lesssim \mu L \,\|\mathcal{T}(\theta_B) - \theta_A\|\,, \tag{8}$$

up to higher-order nonlinear terms. This local bound is why μ stays small. Coarser ratios collapse performance.

D.2 UNION-STYLE EXPAND

Let the source checkpoint have hidden size h_s, intermediate size m_s, number of layers L_s, number of query heads H_s, number of key-value heads K_s, and head dimension d_h. Let the target architecture have the corresponding quantities h_t, m_t, L_t, H_t, K_t, with target dimensions no smaller than the source dimensions for the expanded axes. We use a slot map

$$\pi_{n\to m}(i) = \left\lfloor \frac{im}{n} \right\rfloor, \qquad i = 0, \ldots, n-1, \tag{9}$$

to place source coordinates or heads into deterministic target slots. For a tensor $W \in \mathbb{R}^{a_s \times b_s}$ expanded along rows and columns, the generic zero-extension rule is

$$\big[\mathcal{E}(W)\big]_{\pi_{a_s \to a_t}(i),\, \pi_{b_s \to b_t}(j)} = c\, W_{ij}, \qquad \big[\mathcal{E}(W)\big]_{uv} = 0 \text{ otherwise}, \tag{10}$$

where c is one for ordinary linear weights and may be set to $\sqrt{h_s/h_t}$ for copied hidden-width normalization coordinates in width-changing expansions. In practice, hidden coordinates are placed at a head-dimension block granularity when the architecture exposes a fixed head dimension. This preserves contiguous per-head subspaces rather than scattering individual coordinates.

Attention projections are expanded with grouped-query structure preserved. Query rows are mapped at whole-head granularity, while key and value rows are mapped at key-value-head granularity. The output projection applies the inverse placement on its input-head dimension and the hidden-coordinate placement on its output dimension. Thus the target architecture keeps its own attention scale $1/\sqrt{d_h}$ and head dimension. New attention rows or columns are zero at initialization, so they do not introduce untuned high-variance logits before interpolation.

The MLP expansion follows the semantic axes of the SwiGLU block. For gate and up projections, source rows are copied into the first or mapped m_s intermediate slots and source hidden columns are copied into mapped hidden slots. For the down projection, source hidden rows are copied into mapped hidden slots and source intermediate columns are copied into the retained intermediate slots. All newly introduced MLP coordinates are zero-filled. Because a SwiGLU branch has the form

$$\mathrm{MLP}(x) = W_{\mathrm{down}}\left(\mathrm{SwiGLU}(W_{\mathrm{gate}}x, W_{\mathrm{up}}x)\right), \tag{11}$$

zero-filling new gate, up, and down coordinates prevents newly inserted nonlinear branches from contributing before interpolation.

For additional layers, we use residual identity initialization. Consider a pre-normalization Transformer block with normalization operator $\mathcal{N}_\ell$ and attention/MLP branch $\mathcal{G}_\ell$:

$$h_{\ell+1} = h_\ell + \mathcal{G}_\ell\big(\mathcal{N}_\ell(h_\ell); W_\ell\big). \tag{12}$$

For a newly inserted layer, all attention and MLP branch weights are set to zero, while normalization scale parameters in the inserted layer are set to the neutral value one. This gives

$$\mathcal{G}_\ell\big(\mathcal{N}_\ell(h_\ell); W_\ell^{\mathrm{new}}\big) = 0, \qquad h_{\ell+1} = h_\ell, \tag{13}$$

so $\Delta h_\ell = 0$ at initialization. Normalization scales are not set to zero, because zero scales would collapse normalized feature variation. For LayerNorm (Ba et al., 2016),

$$\mathcal{N}_\ell(h) = \gamma_\ell \odot \frac{h - \bar{h}}{\sqrt{\mathrm{Var}(h) + \epsilon}} + \beta_\ell, \tag{14}$$

and the local Jacobian scales with γ_ℓ. Setting $\gamma_\ell = \mathbf{1}$ preserves a well-conditioned normalized signal. The same intuition applies to RMSNorm (Zhang and Sennrich, 2019).

Pseudocode 2: EXPAND for union-style merging

Input: source checkpoint θ_s, target architecture $\mathcal{A}_t$, source and target configs.
Output: expanded checkpoint $\tilde{\theta}_s = \mathcal{E}(\theta_s)$ in the target parameter space.

1 Initialize a model with architecture $\mathcal{A}_t$ and obtain its target state dictionary.
2 Build slot maps for hidden coordinates, query heads, key-value heads, and intermediate MLP dimensions.
3 Copy embeddings and the language-model head into mapped hidden coordinates; set unmapped coordinates to zero.
4 For each source layer $\ell < L_s$, copy normalization parameters into matched coordinates and initialize newly introduced normalization scales to neutral values when applicable.
5 Expand q, k, v, and o attention matrices by whole-head or key-value-head slots, preserving grouped-query attention structure.
6 Expand MLP gate/up/down matrices along their semantic input and output dimensions; zero-fill newly introduced SwiGLU coordinates.
7 For layers $L_s, \ldots, L_t - 1$, set attention and MLP branch weights to zero and set normalization scales to one, producing residual identity blocks.
8 Save the expanded checkpoint and use it in $\theta^{\cup}(\lambda) = (1-\lambda)\mathcal{E}(\theta_s) + \lambda\theta_t$.

Table 25: EXPAND constructs a larger-architecture checkpoint by deterministic slot insertion, zero-filled new branches, and residual identity initialization for added layers.

After expansion, interpolation is applied near a source:

$$\theta^{\cup}(\lambda) = \mathcal{E}(\theta_s) + \lambda\big(\theta_t - \mathcal{E}(\theta_s)\big). \tag{15}$$

Small λ keeps the expanded source as the anchor; λ close to one keeps the target checkpoint as the anchor. Coarser ratios collapse performance.

The absence of permutation or semantic alignment is deliberate. TRUNCATE and EXPAND assume architecture-compatible cross-scale models whose tensor names, module roles, and head structure are sufficiently comparable. Coarser ratios collapse because that assumption is then too weak for direct weighted averaging.