From "Weak" Signals to Strong Models: Preference Delta Aggregation with LoRA Merging

Qi Sun[1*] **Siyue Zhang**[2*†] **Yulin Chen**[3] **Yuxiang Xue**[1] **Ru Peng**[4] **Chen Zhao**[1†]

[1]NYU Shanghai [2]NTU [3]NYU [4]ZJU

{qs2196, yc7320, yx3044, cz1285}@nyu.edu
siyue001@e.ntu.edu.sg rupeng@zju.edu.cn

AlbertQiSun/Preference-Delta-Aggregation

Abstract

Training strong large language models (LLMs) requires high-quality supervision, which is often scarce. Recent work shows that paired preference data from weak-weaker model pairs (*e.g.,* Qwen3 4B over 1.7B), despite the limited quality of individual responses, can provide an effective supervision signal through relative quality deltas, which we term a "weak" signal. This motivates a key research question: can multiple "weak" signals be constructively aggregated for improving strong models (*e.g.,* Qwen3 8B)? To this end, we propose Preference Delta Aggregation (PDA), the first framework that derives a preference delta from each weak-weaker model pair, instantiates it as a LoRA adapter learned through preference optimization, and aggregates the resulting deltas via LoRA merging. To improve the composition of independently trained adapters, we introduce Geometric Alignment Merging (GAM), which aligns their left and right singular bases through orthogonal Procrustes rotations and averages the aligned factors. Evaluations on knowledge reasoning and agentic search show that PDA-GAM exceeds the strongest individual delta, with further gains along the evaluated sequences of additional deltas. Using three reasoning deltas and four agentic deltas, PDA-GAM improves Qwen3-8B by 6.8 and 7.3 points on average, respectively, exceeding the strongest single-delta baseline by 2.1 and 4.3 points. These configurations achieve the highest average scores among the evaluated merging methods, with similar gains observed on a second student model. Further analysis suggests that these gains come from composing complementary capabilities encoded across distinct preference deltas.[3]

1 Introduction

High-quality data is widely recognized as a key ingredient in building strong large language models (LLMs). Accordingly, substantial prior work has focused on curating training data throughout the pipeline, spanning pretraining [Li et al., 2024b, Penedo et al., 2024, Walsh et al., 2025] and post-training [Nvidia et al., 2024, Yang et al., 2025b, Chen et al., 2026]. However, many desirable tasks remain difficult to supervise effectively, either due to prohibitive annotation costs or because they exceed human expertise. This challenge has motivated growing interest in leveraging low-quality data to expand the frontier of LLM capabilities [Zhu et al., 2025, Yao et al., 2025]. Building on this direction, recent work introduces the Delta Learning Hypothesis [Geng et al., 2025], which posits that

*Equal contribution.

†Corresponding authors: Chen Zhao (cz1285@nyu.edu) and Siyue Zhang (siyue001@e.ntu.edu.sg).

[3]Code and data are available at https://github.com/AlbertQiSun/Preference-Delta-Aggregation.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

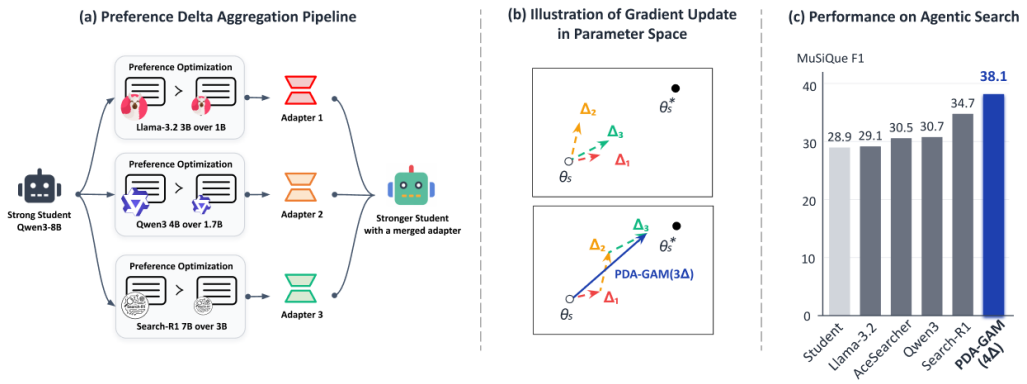


Figure 1: **(a)** Preference Delta Aggregation (PDA) independently preference-tunes a strong student model on preference datasets from different weak-weaker model pairs using LoRA fine-tuning, then merges the resulting adapters. **(b)** In parameter space, each adapter induces a distinct update direction, and PDA aggregates these deltas to compose complementary improvements, moving the student toward a better solution. **(c)** By aggregating multiple preference deltas, PDA achieves larger gains than baselines trained on preference data from any single weak-weaker model pair.

relative quality differences between weak responses (*e.g.,* Qwen3 4B over 1.7B) can serve as effective "weak" supervision signals for improving strong models (*e.g.,* Qwen3-8B) through preference tuning.

The success of the Delta Learning Hypothesis raises a question: can multiple such "weak" signals be aggregated to yield further gains? We first examine sequential preference optimization over multiple datasets and joint optimization on their mixture. In our experiments, sequential training falls below the untuned student, while joint training falls below the strongest single delta. The agentic joint baseline uses the same preference datasets and a similar total number of optimization steps as our aggregation framework. These results motivate an alternative to combining the datasets during training.

Therefore, we propose the Preference Delta Aggregation framework (PDA), which independently preference-tunes the strong student on datasets from different weak-weaker model pairs using LoRA [Hu et al., 2021], as illustrated in Figure 1. Each adapter encodes the parameter update learned from one pair, which we term a *preference delta*. We aggregate these updates through LoRA merging [Yadav et al., 2023, Stoica et al., 2024, Huang et al., 2024]. Independent training retains a separate update for each source pair, allowing us to combine the learned signals after optimization. All adapters start from the same student and modify the same target modules, making them compatible for parameter-space merging. This separates the question of whether a source pair provides useful supervision from the question of how multiple learned updates should be composed. The merging operator determines how the adapters contribute to the resulting model. We introduce Geometric Alignment Merging (GAM), which decomposes each update by singular value decomposition (SVD), aligns its left and right singular bases to a reference, and averages the aligned factors.

Following prior work [Geng et al., 2025], we evaluate the proposed PDA and GAM on knowledge reasoning tasks, and further extend the evaluation to more challenging agentic search settings requiring both reasoning and retrieval [Zhang et al., 2025, 2026]. We construct preference datasets from GSM8K [Cobbe et al., 2021] and MuSiQue [Trivedi et al., 2022] using multiple weak-weaker model pairs drawn from diverse model families, including Llama-3.2 [Grattafiori et al., 2024], Qwen3 [Yang et al., 2025a], DeepSeek-R1 [Guo et al., 2025], AceSearcher [Xu et al., 2025], and Search-R1 [Jin et al., 2025a]. We evaluate strong student models, including Qwen3-8B and Tülu3-8B [Lambert et al., 2025], across diverse reasoning benchmarks as well as both single-hop and multi-hop search QA benchmarks, and compare against four categories of baselines: the original student models without fine-tuning, single-delta preference tuning, training-based multi-delta aggregation methods (*e.g.,* sequential and joint training), and existing LoRA merging approaches such as naive averaging [Ilharco et al., 2023], TIES-Merging [Yadav et al., 2023], and KnOTS [Stoica et al., 2024].

Using three reasoning deltas and four agentic deltas, PDA-GAM improves Qwen3-8B by 6.8 and 7.3 points on average, respectively. It exceeds the strongest single delta by 2.1 and 4.3 points, and

similar gains are observed on Tülu3-8B. Performance increases along the evaluated sequences of additional deltas. GAM achieves the highest average scores among the compared operators at these configurations, while TIES-Merging performs better with two deltas on agentic search. Behavioral analysis suggests that aggregation combines complementary capabilities learned from different source pairs. Our contributions are as follows:

(1) A framework for improving strong models by aggregating "weak" preference signals. We propose PDA, the first framework that improves strong models without relying on high-quality data by leveraging paired preference data from multiple weak-weaker model pairs and aggregating the resulting signals through LoRA merging.

(2) A LoRA merging method that mitigates directional interference among preference deltas. We introduce GAM, which aligns singular bases through orthogonal Procrustes rotations and averages the aligned factors to produce an update of rank at most r.

(3) Strong empirical results on reasoning and agentic search tasks. Experiments on two domains and two student models show that PDA-GAM exceeds the strongest single-delta and training-based aggregation baselines at the largest evaluated delta sets. Additional analyses examine training variance, merge rank, training budget, and preference-label reliability.

2 Related Work

Weak-to-strong Generalization. Weak-to-strong generalization studies whether supervision from a weaker model can effectively improve a stronger one without collapsing to the weak supervisor's performance ceiling [Burns et al., 2023, Ye et al., 2026]. Prior studies largely center on two directions: uncovering capabilities already present in pretrained base models [Hase et al., 2024], and using models to recursively refine the data used for subsequent training [Wu et al., 2025]. In both settings, supervision is commonly delivered as pointwise judgments from a single weak teacher. However, recent work on AI feedback suggests that such absolute supervision can propagate the teacher's systematic errors and hallucination biases [Zheng et al., 2023, Li et al., 2024a]. This motivates relative supervision signals based on quality differences rather than absolute judgments [Zhu et al., 2025, Yao et al., 2025, Geng et al., 2025].

Delta Learning. Existing alignment approaches rely heavily on human annotations or supervision from frontier models through frameworks such as RLHF or DPO [Ouyang et al., 2022, Bai et al., 2022, Rafailov et al., 2024]. Delta learning relaxes this costly supervision requirement by constructing preference pairs from a weak model and its weaker variant, using the quality gap between their responses as an optimization signal for a stronger student [Geng et al., 2025]. While delta learning is related to synthetic preference generation [Yuan et al., 2025], its application to multi-turn agentic search remains underexplored. In this setting, models iteratively interact with external tools [Yao et al., 2023]. We evaluate delta learning on agentic search and study whether aggregating multiple deltas can improve on the strongest individual delta.

LoRA Tuning and Merging. Low-Rank Adaptation (LoRA) [Hu et al., 2021] enables parameter-efficient adaptation of large language models by introducing lightweight low-rank modules, avoiding full-model fine-tuning for each downstream task. Its modular structure naturally supports post-hoc composition, leading to increasing interest in combining specialized capabilities learned from different tasks or domains [Jin et al., 2025b]. Task Arithmetic [Ilharco et al., 2023] composes parameter updates through weighted addition. TIES-Merging [Yadav et al., 2023] and DARE [Yu et al., 2024] reduce interference through sparsification and, in TIES, sign resolution. KnOTS [Stoica et al., 2024] concatenates updates and computes a joint SVD to obtain a shared basis, then applies an existing merging operator to adapter-specific factors. Geometric Alignment Merging instead decomposes each adapter separately, aligns both singular bases through orthogonal Procrustes, and averages the aligned bases and core matrices. AdapterSoup [Chronopoulou et al., 2023] and LoraHub [Huang et al., 2024] select or weight adapters for transfer across domains or tasks. Our experiments aggregate adapters derived from different source model pairs within each training domain.

Noisy Preference Labels. Preference labels produced by models are often noisy or mutually inconsistent. Prior work measures the consistency and transitivity of LLM judges [Feng et al., 2025], aggregates the cyclic preferences of multiple evaluators into acyclic rankings [Hu et al., 2026],

resolves contradictions within the preferences of a single model [Zhang et al., 2024], and designs preference objectives that remain stable under label noise [Shankar and Fiterau, 2025]. Delta learning assigns each label by model size, which is a noisy proxy for response quality at the level of individual pairs. We report how often this label agrees with gold answers for every source pair in Appendix G.

3 Preference Delta Aggregation

To aggregate multiple "weak" supervision signals, we propose **Preference Delta Aggregation (PDA)**, a two-stage framework that first derives a preference delta from each weak-weaker model pair and then aggregates the resulting LoRA-instantiated updates through parameter-space merging.

Deriving Preference Deltas. Training on paired responses allows a student to learn from relative response quality [Geng et al., 2025]. We construct n preference datasets from n model families. For each family i, a weak model π_{w_i} and a smaller model $\pi_{w_i'}$ generate responses to the same queries. We label the larger model's response as chosen and the smaller model's as rejected, yielding $\mathcal{D}_i$. Model size serves as a noisy proxy for pairwise quality. We reserve π for the fixed source policies and θ for the student parameters.

We train the student θ_s on each dataset $\mathcal{D}_i$ through preference optimization, using objectives such as DPO [Rafailov et al., 2024] or ORPO [Hong et al., 2024]. We call the resulting parameter update a *preference delta*, denoted by Δ_i. Its usefulness for the student is evaluated empirically. This supervision does not require either source response to match the quality of the student. Instead, the preference objective compares the two responses and changes the student according to their assigned relative ordering. Different source families can supply different response patterns on the same queries, giving rise to distinct updates even when the training questions are shared.

Formulating Delta Aggregation. Given preference deltas $\{\Delta_i\}_{i=1}^n$, our goal is to effectively aggregate these diverse signals to improve the strong student model θ_s:

$$\mathcal{A}(\{\Delta_i\}_{i=1}^n; \theta_s) = \theta_s',$$

where $\mathcal{A}$ is an aggregation operator and θ_s' is the resulting model. Training-based approaches either jointly optimize on $\bigcup_{i=1}^n \mathcal{D}_i$ or train sequentially on the datasets. Gradient conflicts and forgetting are potential difficulties in such settings [Yu et al., 2020, Chen and Liu, 2018], motivating independent training followed by parameter-space aggregation.

Aggregating Deltas via LoRA Merging. PDA instantiates each preference delta as an independently trained LoRA adapter. For each preference dataset $\mathcal{D}_i$, we fine-tune the same base student model θ_s with LoRA, obtaining an adapter ΔW_i that realizes the abstract preference delta Δ_i in parameter space. Loading this adapter into the base model gives

$$\theta_s^{(i)} = \theta_s \oplus \Delta W_i,$$

where $\oplus$ denotes applying the LoRA adapter to the target modules of θ_s. The learned adapters $\{\Delta W_i\}_{i=1}^n$ are then merged to produce the final model θ_s'.

For a given adapted weight matrix, each LoRA adapter represents a low-rank update

$$\Delta W_i = B_i A_i \in \mathbb{R}^{d\times k},$$

where k and d denote the input and output dimensions, $B_i \in \mathbb{R}^{d\times r}$, $A_i \in \mathbb{R}^{r\times k}$, and $r \ll \min(d,k)$. A standard post-hoc baseline averages the expanded updates. Our PDA-Averaging baseline truncates this average to rank at most r:

$$\Delta W_{\text{AVG}} = \frac{1}{n}\sum_{i=1}^{n} B_i A_i, \qquad \theta_s' = \theta_s \oplus \mathcal{P}_r(\Delta W_{\text{AVG}}).$$

Here $\mathcal{P}_r$ retains the leading r singular components. The untruncated average can have rank up to nr, whereas both PDA-Averaging and GAM have rank at most r. Appendix E compares against the untruncated average as well.

Weight-space averaging is invariant to the choice of factors for each update. GAM instead combines the factors separately, so their relative basis orientations matter. We align these bases before aggregation while preserving the update represented by each adapter.

4 Geometric Alignment Merging

Geometric Alignment Merging (GAM) decomposes each adapter through SVD, aligns its left and right singular bases to a reference through orthogonal Procrustes, and averages the aligned factors.

Step 1: Decomposition. For each trained LoRA adapter, we factor its update $\Delta W_i = B_i A_i \in \mathbb{R}^{d\times k}$ into r singular components:

$$\Delta W_i = U_i S_i V_i^\top \tag{1}$$

where $U_i \in \mathbb{R}^{d\times r}$ and $V_i \in \mathbb{R}^{k\times r}$ have orthonormal columns, and $S_i \in \mathbb{R}^{r\times r}$ is diagonal. This factorization is exact because ΔW_i has rank at most r. It can be computed without forming the dense update. Given thin QR factorizations $B_i = Q_B R_B$ and $A_i^\top = Q_A R_A$, we take the small SVD $R_B R_A^\top = \hat{U} S_i \hat{V}^\top$ and set $U_i = Q_B \hat{U}$ and $V_i = Q_A \hat{V}$.

Step 2: Basis Alignment by Orthogonal Procrustes. A subspace admits multiple orthonormal bases related by rotation [Edelman et al., 1998]. Independently trained adapters can therefore have different basis orientations, which can cause cancellation when the bases are averaged directly.

We align each adapter to a reference ΔW_1, selected by validation performance, using orthogonal Procrustes [Hurley and Cattell, 1962]:

$$R_i^U = \arg\min_{R\in O(r)} \|U_i R - U_1\|_F, \qquad R_i^V = \arg\min_{R\in O(r)} \|V_i R - V_1\|_F. \tag{2}$$

These rotations minimize basis mismatch to the reference while preserving each adapter's subspaces. The problem has a closed-form solution: if $U_i^\top U_1 = P_i \Sigma_i Q_i^\top$, then $R_i^U = P_i Q_i^\top$, with the right rotation obtained analogously. Alignment changes the coordinates used to represent an update, rather than moving its column or row subspace. Thus, distinct subspaces need not coincide after alignment. We use an unweighted objective so that alignment treats all retained basis vectors equally. To preserve each complete update, we transform its core matrix along with the bases:

$$\tilde{U}_i = U_i R_i^U, \qquad \tilde{S}_i = (R_i^U)^\top S_i R_i^V, \qquad \tilde{V}_i = V_i R_i^V. \tag{3}$$

The aligned core $\tilde{S}_i$ is generally not diagonal. This transformation satisfies $\tilde{U}_i \tilde{S}_i \tilde{V}_i^\top = \Delta W_i$. Reference-choice sensitivity is reported in Appendix O.

Step 3: Averaging and Reconstruction. We average the aligned bases and core matrices separately:

$$\bar{U} = \frac{1}{n}\sum_{i=1}^{n} \tilde{U}_i, \quad \bar{S} = \frac{1}{n}\sum_{i=1}^{n} \tilde{S}_i, \quad \text{and} \quad \bar{V} = \frac{1}{n}\sum_{i=1}^{n} \tilde{V}_i. \tag{4}$$

Finally, the averaged components are multiplied to reconstruct the final merged weight update:

$$\Delta W_{\text{GAM}} = \bar{U}\,\bar{S}\,\bar{V}^\top \tag{5}$$

The product has rank at most r and can be stored as one LoRA adapter with $B' = \bar{U}\bar{S}$ and $A' = \bar{V}^\top$. Its adapter parameter count is independent of n. Unlike a sum of complete updates, the product of averaged factors also combines factors originating from different adapters. The rotations therefore affect the merged result even though they leave every individual update unchanged. Appendix C proves update preservation under rotation and reduction to ordinary weight averaging when all adapters share both left and right subspaces. These properties clarify where GAM can differ from averaging. Whether that difference improves task performance is evaluated through the merging comparisons and controls below.

5 Experiments

We evaluate Preference Delta Aggregation on knowledge reasoning and agentic search with two student models. Our experiments examine whether individual weak preference signals transfer to a stronger student, whether aggregation improves on the strongest individual signal, and how the merging operator affects the result. We then use training-budget comparisons, repeated training seeds, and fixed-adapter controls to examine the observed gains.

5.1 Knowledge Reasoning Task Setup

Tasks and Data. We evaluate Preference Delta Aggregation on the knowledge reasoning task using four widely used benchmarks: MATH [Lightman et al., 2023], GPQA [Rein et al., 2023], MMLU [Hendrycks et al., 2021], and GSM8K [Cobbe et al., 2021]. To construct the preference data, we use the full GSM8K training set, consisting of 7,473 queries. For each query, we generate one pair of response trajectories for every weak-weaker model pair. Preference labels are automatically assigned based on model size, with responses from larger (weak) models treated as preferred over those from smaller (weaker) models ($y_{\text{weak}} \succ y_{\text{weaker}}$). This label is a noisy proxy for response quality at the level of individual pairs, and Appendix G reports its agreement with gold answers for all seven source pairs.

Model Pairs and Preference Deltas. We use Qwen3-8B [Yang et al., 2025a] as the primary strong student model, and Tülu3-8B [Lambert et al., 2025] to evaluate cross-model generalization. The preference deltas are derived from three pairs of weak and weaker models: Llama-3.2 3B over 1B [Grattafiori et al., 2024], Qwen3 4B over 1.7B, and DeepSeek-R1 7B over 1.5B [Guo et al., 2025]. As detailed in Appendix A, all selected weak and weaker LLMs consistently underperform the student model, thereby ensuring a setting with limited high-quality supervision.

Training and Merging Details. We train one LoRA adapter per source pair using Direct Preference Optimization (DPO) [Rafailov et al., 2024, Hu et al., 2021]. GAM combines these adapters as described in Section 4. Training hyperparameters are provided in Appendix K.

5.2 Agentic Search Task Setup

Tasks and Data. We evaluate Preference Delta Aggregation on agentic search using three single-hop QA benchmarks (NQ [Kwiatkowski et al., 2019], TriviaQA [Joshi et al., 2017], PopQA [Mallen et al., 2023]) and four multi-hop QA benchmarks (HotpotQA [Yang et al., 2018], 2WikiMultiHopQA [Ho et al., 2020], Bamboogle [Press et al., 2023], MuSiQue [Trivedi et al., 2022]). Following the same process described in Section 5.1, the preference data is constructed using weak-weaker model pairs and 8,000 queries randomly sampled from the MuSiQue training set. Unlike the reasoning setup, agentic search trajectories require strict structural formatting. We apply a formatting filter only to the weak model's (chosen) responses, while retaining formatting errors in the weaker model's (rejected) responses to preserve the negative signal.

Model Pairs and Preference Deltas. We use the same student models as in Section 5.1, namely Qwen3-8B and Tülu3-8B. We construct preference deltas from four LLM families: Llama-3.2 3B over 1B, AceSearcher 7B over 3B [Xu et al., 2025], Qwen3 4B over 1.7B, and Search-R1 7B over 3B [Jin et al., 2025a]. Llama 3.2 and Qwen3 are prompted using ReAct [Yao et al., 2023] and one example to operate as search agents, whereas AceSearcher and Search-R1 are fine-tuned for this role. Similar to the model selection in Section 5.1, all selected weak and weaker LLMs underperform the student model as shown in Appendix A.

Training and Merging Details. Agentic search requires interleaved reasoning and retrieval steps. In preliminary experiments, DPO reduces the student's format correctness rate from 0.97 to 0.83. We therefore use ORPO [Hong et al., 2024], whose supervised component preserves formatting while optimizing preferences, restoring the rate to 0.97. We also train a chosen-only SFT baseline on the Qwen3-4B responses. It underperforms the untuned student (Appendix P), indicating that direct imitation of these responses is insufficient to explain the preference-tuning gains. Other training settings follow the reasoning setup (Appendix K).

5.3 Baselines

We compare Preference Delta Aggregation and Geometric Alignment Merging against three categories of baselines: **(1) Student Baseline**: We evaluate the strong student LLMs, *i.e.,* Qwen3-8B and Tülu3-8B, without any fine-tuning. **(2) Single Preference Delta Baselines**: We fine-tune the student separately on the preference data from each source pair. **(3) Multiple Preference Delta Baselines**: We compare methods that leverage signals from multiple quality and preference deltas, covering both training-based and LoRA-based approaches. For training-based aggregation methods, we consider

Table 1: Exact Match scores on knowledge reasoning benchmarks. For multi-delta experiments, $k\Delta$ uses the first k preference deltas in the order DeepSeek-R1, Qwen3, Llama-3.2. For example, 2Δ denotes aggregating DeepSeek-R1 and Qwen3. The best score is highlighted in **bold** for single-delta and multi-delta baselines.

Method	Qwen3-8B					Tülu3-8B				
	MATH	GPQA	MMLU	GSM8K	Avg.	MATH	GPQA	MMLU	GSM8K	Avg.
Student Baseline	81.5	42.1	78.5	88.0	72.5	78.2	38.5	75.2	85.1	69.3
Single Preference Delta										
Llama-3.2 (3B over 1B)	82.8	42.6	79.1	88.5	73.3	79.5	39.2	76.0	86.0	70.2
Qwen3 (4B over 1.7B)	84.5	44.0	80.2	90.2	74.7	81.2	40.5	77.5	88.2	71.9
DeepSeek-R1 (7B over 1.5B)	**87.2**	**46.5**	**81.8**	**93.4**	**77.2**	**84.6**	**43.1**	**79.2**	**91.0**	**74.5**
Multiple Preference Deltas										
Sequential DPO (2Δ)	79.2	39.8	76.9	88.2	71.0	76.3	34.6	72.3	82.2	66.4
Joint DPO (3Δ)	85.1	44.3	80.6	89.6	74.9	82.7	40.2	78.4	89.5	72.7
PDA-Averaging (3Δ)	86.8	46.2	80.5	92.5	76.5	83.5	42.8	78.4	90.1	73.7
PDA-KnOTS (3Δ)	87.8	47.1	81.4	93.7	77.5	85.5	44.2	79.7	91.4	75.2
PDA-TIES (2Δ)	87.6	46.8	80.4	93.7	77.1	85.2	43.9	79.5	91.2	75.0
PDA-TIES (3Δ)	88.1	47.3	81.5	93.8	77.7	85.8	44.5	79.8	91.5	75.4
PDA-GAM (2Δ)	88.5	47.9	82.4	94.2	78.3	86.2	44.8	80.5	92.2	75.9
PDA-GAM (3Δ)	**89.6**	**48.8**	**83.1**	**95.5**	**79.3**	**87.4**	**46.0**	**81.2**	**93.5**	**77.0**

Sequential Training, where the student is fine-tuned consecutively across different deltas, and *Joint Training*, which optimizes the student model at once with a mixture of data from multiple deltas. For LoRA-based aggregation methods within our Preference Delta Aggregation framework, we compare our proposed GAM with existing LoRA merging methods, including *Naive Averaging* [Ilharco et al., 2023] truncated to rank r (Section 3), *TIES-Merging* [Yadav et al., 2023], and *KnOTS* [Stoica et al., 2024].

5.4 Main Results

Single preference deltas improve both student models. Fine-tuning Qwen3-8B on the DeepSeek-R1 pair improves its average reasoning score from 72.5 to 77.2 (Table 1), consistent with the Delta Learning Hypothesis [Geng et al., 2025]. On agentic search, the Search-R1 pair raises average F1 from 50.7 to 53.7 (Table 2). Each evaluated source pair improves the corresponding student average, with different pairs yielding different gains. These improvements also hold for Tülu3-8B, supporting the use of relative preferences from weaker models across both domains.

Parameter-space aggregation improves on the strongest single delta. At the largest evaluated delta sets, PDA-GAM reaches 79.3 on reasoning and 58.0 on agentic search for Qwen3-8B, exceeding the strongest single delta by 2.1 and 4.3 points. Joint training falls below that single-delta baseline, and sequential training falls below the untuned student. TIES also improves Qwen3-8B from the strongest single delta's 53.7 to 57.4 on agentic search. Thus, the benefit of aggregating independently trained deltas extends beyond the proposed merging operator.

The gains persist under the reported training-budget controls. On agentic search, joint training and PDA use the same 30,608 available preference pairs, with 227 and 225 total optimization steps, respectively. Both exclude a 5% validation split from optimization. Joint ORPO reaches 52.8 average F1, below the strongest single delta at 53.7. Extending that single delta to four epochs reaches 54.1 in 236 steps, compared with 58.0 for PDA-GAM (Appendix F). The extra epochs reuse the same source pair's data, whereas PDA trains on responses from four source pairs. These comparisons show that the improvement is not reproduced by simply extending the strongest single-delta run. PDA does train four separate adapters, while retaining only one merged adapter at inference.

Along the strongest-first sequences in the main tables, PDA-GAM improves as deltas are added. On Qwen3-8B agentic search, its average F1 rises from 54.8 with 2Δ to 57.4 with 3Δ and 58.0 with 4Δ. Even the later sources, which are weaker individually, contribute gains in these combinations. Averaging over all subsets of each size gives 52.35, 53.72, 55.73, and 58.00 on agentic search, and 75.07, 77.40, and 79.30 on reasoning (Appendix H). Thus, the trend extends beyond the strongest-first sequence, although source quality still matters.

Table 2: Token-level F1 scores on agentic search benchmarks. For multi-delta experiments, $k\Delta$ uses the first k preference deltas in the order Search-R1, Qwen3, AceSearcher, Llama-3.2. For example, 2Δ denotes aggregating Search-R1 and Qwen3. The best score is highlighted in **bold** for single-delta and multi-delta baselines.

Method	Qwen3-8B								Tülu3-8B							
	Single-hop QA			Multi-hop QA				Avg.	Single-hop QA			Multi-hop QA				Avg.
	NQ	TQA	Pop.	Hot.	2Wiki	Bam.	MuSi.		NQ	TQA	Pop.	Hot.	2Wiki	Bam.	MuSi.	
Student Baseline	46.2	61.4	44.6	53.8	58.1	61.6	28.9	50.7	45.8	60.5	43.0	52.4	55.1	58.8	25.6	48.7
Single Preference Delta																
Llama-3.2 (3B over 1B)	46.5	61.6	44.9	53.9	58.5	62.0	29.1	50.9	46.0	60.8	43.5	52.6	55.8	59.5	26.2	49.2
AceSearcher (7B over 3B)	47.8	62.1	46.2	54.1	60.1	62.8	30.5	51.9	46.5	61.5	45.2	53.0	57.4	61.0	27.8	50.3
Qwen3 (4B over 1.7B)	48.5	62.8	47.0	54.2	61.5	63.5	32.8	52.9	46.8	62.0	46.8	53.5	59.2	62.1	28.5	51.3
Search-R1 (7B over 3B)	**49.1**	**63.2**	**47.9**	**54.4**	**62.6**	**64.0**	**34.7**	**53.7**	**47.2**	**62.8**	**48.7**	**53.8**	**61.3**	**62.9**	**29.5**	**52.3**
Multiple Preference Deltas																
Sequential ORPO (2Δ)	44.6	58.2	41.7	51.4	52.2	59.5	27.6	47.9	42.8	56.9	39.8	51.6	50.1	53.7	21.4	45.2
Joint ORPO (4Δ)	48.7	62.8	46.8	53.6	62.0	63.6	32.0	52.8	47.0	62.3	47.1	53.4	60.6	62.5	28.9	51.7
PDA-Averaging (4Δ)	53.2	65.4	50.3	57.9	64.9	69.5	36.9	56.9	51.2	65.8	52.5	56.4	63.8	68.3	32.0	55.7
PDA-KnOTS (4Δ)	53.0	65.2	50.1	57.8	64.7	69.8	37.0	56.8	50.9	65.5	52.3	56.2	63.6	68.1	31.9	55.5
PDA-TIES (2Δ)	51.6	64.7	49.2	56.8	63.4	66.9	35.6	55.5	49.0	64.1	50.7	55.5	62.8	66.4	31.6	54.3
PDA-TIES (3Δ)	52.4	65.0	51.2	58.2	65.2	69.7	37.8	57.1	50.9	65.4	52.6	56.8	63.8	68.6	32.0	55.7
PDA-TIES (4Δ)	52.8	65.3	51.5	58.8	65.4	70.3	37.8	57.4	51.4	65.7	52.8	57.2	64.0	68.8	32.1	56.0
PDA-GAM (2Δ)	50.4	64.1	48.7	55.3	63.5	65.7	36.1	54.8	49.6	64.0	50.4	54.4	62.4	64.7	30.3	53.7
PDA-GAM (3Δ)	52.8	65.7	51.6	58.6	65.6	69.8	37.9	57.4	51.1	65.6	52.7	57.1	63.8	68.8	32.4	55.9
PDA-GAM (4Δ)	**53.2**	**65.9**	**52.4**	**59.3**	**65.9**	**71.2**	**38.1**	**58.0**	**51.6**	**65.9**	**52.9**	**57.2**	**64.2**	**69.1**	**32.8**	**56.2**

The benefit of Geometric Alignment Merging varies across domains and delta counts. On Qwen3-8B reasoning with 3Δ, PDA-GAM reaches 79.3, compared with 77.7 for TIES, 77.5 for KnOTS, and 76.5 for Averaging. Both TIES and KnOTS exceed the strongest single delta at 77.2, while GAM provides the largest additional gain. On agentic search with 4Δ, GAM leads TIES by 0.6 points for Qwen3-8B and 0.2 points for Tülu3-8B. TIES performs better with 2Δ for both students, scoring 55.5 versus 54.8 and 54.3 versus 53.7, respectively. Appendix J reports all four operators at each delta count for Qwen3-8B agentic search.

Paired training seeds support the operator comparison. We retrain each preference delta with three seeds while holding the preference data, hyperparameters, and evaluation sets fixed. Within each seed, GAM and TIES merge the same adapters, so their paired difference compares operators on a common set of trained updates. Across three training seeds for Qwen3-8B, the paired GAM-minus-TIES difference is $+1.43 \pm 0.43$ points on the four-benchmark reasoning average and $+0.67 \pm 0.40$ on MuSiQue, where $\pm$ denotes the sample standard deviation (Appendix D). The difference is positive for all three seeds in both settings and for each of the four reasoning benchmarks within each seed. On MuSiQue, the individual methods' scores vary by 3.0 to 3.3 points across seeds, while the paired difference has a standard deviation of 0.40. Reporting paired differences separates the operator comparison from variation shared by the two methods. The agentic seed study covers MuSiQue only, so the seven-benchmark averages in Table 2 are not verified across training seeds. For the fixed-model seven-benchmark comparison, 10,000 paired bootstrap resamples give a 95% confidence interval of $[+0.08, +1.18]$ for the GAM-minus-TIES difference (Appendix D).

Fixed-adapter controls examine alignment, magnitude, and rank. On a fixed set of four agentic adapters, untruncated weight averaging scores 37.0 on MuSiQue and rank-truncated averaging scores 36.9. Averaging the SVD factors separately without rotation also scores 36.9, although it is a different operator. We then rescale this componentwise average separately for each adapted matrix to match the corresponding GAM update's Frobenius norm. This control reaches 37.18, compared with 38.15 for GAM, a difference of 0.97 with a paired bootstrap 95% confidence interval of $[+0.26, +1.92]$. This supports a contribution from basis alignment beyond update magnitude (Appendix E). With the training rank fixed at 64, GAM exceeds rank-matched weight averaging by 1.2 to 1.7 points across merge ranks from 8 to 64. Both methods degrade below rank 32. These results distinguish the choice of merging operator from the output-rank constraint. Whole-model merging on an A800 takes 0.76 s for TIES, 4.41 s for factor-based Averaging, and 6.42 s for factor-based GAM (Appendix I). Appendix M and N provide qualitative examples.

Table 3: Model behavior analysis on 500 MuSiQue dev-set samples, annotated by Qwen3-32B and verified by the authors. Single-delta baselines exhibit distinct behavioral specializations (*e.g.,* Search-R1 excels at adaptive search but struggles with verification), whereas PDA-GAM balances these complementary behavioral capabilities.

Method	Steps	Verification (%)	Adaptive Search (%)	Authority (%)	Recovery (%)
Student Qwen3-8B	4.28	57.2	23.6	0.6	9.3
Single Preference Delta					
Qwen3 (4B over 1.7B)	4.64	75.9	14.7	2.6	10.7
Search-R1 (7B over 3B)	4.09	5.3	88.9	0.8	24.3
PDA-GAM (2Δ)	4.43	61.2	56.7	1.2	17.4

6 Analysis

6.1 Are preference deltas always effective signals for learning stronger models?

The Delta Learning Hypothesis [Geng et al., 2025] motivates learning from relative quality differences between weak and weaker models. Following this setting, we construct preference pairs from models in the same family that differ in scale. This choice provides a simple label proxy, whose reliability we assess against gold answers.

Across all seven source pairs, chosen responses have higher exact match than rejected responses, by 13.76 to 16.28 points on agentic search and 6.27 to 7.68 on reasoning (Appendix G). Exact match distinguishes only 17.6% to 23.6% of pairs. Within this subset, size-based labels agree with gold answers on 84.5% to 89.1% of agentic pairs and 66.3% to 66.8% of reasoning pairs. The positive aggregate difference therefore coexists with noisy individual labels, particularly on reasoning tasks. For most pairs, exact match cannot establish an ordering because both responses are incorrect. Such responses may still differ in intermediate reasoning or search behavior, but the audit does not measure those differences. The reported agreement is therefore conditional on separable pairs and should not be interpreted as the accuracy of all training labels. This distinction matters when using model size to assign preferences without an external judge.

Beyond the setting studied in the Delta Learning Hypothesis, we identify an additional failure case: the preference delta derived from the ZeroSearch family (*i.e.,* 7B over 3B) [Sun et al., 2025] fails to improve a stronger student model. In our experiments, training on this preference data caused a 1.3 F1 score drop on MuSiQue relative to the student baseline (*i.e.,* Qwen3-8B). ZeroSearch trains LLM search agents without real search engines by relying on a simulation LLM that generates progressively harder retrieval environments. We hypothesize that the degradation stems from mismatched task difficulties induced by different simulation LLMs during training, causing the two models to adapt along different informative axes despite sharing the same family. A similar failure was observed when experimenting with another self-play framework, SSRL [Fan et al., 2026]. These results suggest that preference deltas are not universally beneficial and should be selected carefully before aggregation.

6.2 What capabilities are composed when aggregating preference deltas?

We examine how aggregation affects search behavior in a two-delta comparison between Search-R1, Qwen3, and their PDA-GAM combination. Prior work identifies Information Verification, Authority Evaluation, Adaptive Search, and Error Recovery as relevant behaviors [Jin et al., 2026]. Table 3 shows distinct specializations: the Search-R1 delta yields 88.9% Adaptive Search but 5.3% Information Verification, whereas the Qwen3 delta yields 75.9% Information Verification but 14.7% Adaptive Search.

PDA-GAM (2Δ) reaches 56.7% Adaptive Search and 61.2% Information Verification, exceeding the averages of the two specialists at 51.8% and 40.6%, respectively. The merged model does not reach either specialist's peak, but it avoids the low rate of that specialist's other behavior. This is useful in multi-hop search, where an agent must both retrieve relevant evidence and check whether that evidence supports its answer. Improving only one of these behaviors can leave the other as a practical bottleneck. The examples in Appendix N illustrate their combination within the same model, offering

a possible behavioral explanation for the improved search performance. These behavioral frequencies describe generated trajectories rather than a decomposition of the model's parameters.

7 Conclusion

We introduce PDA to aggregate preference signals from multiple weak-weaker model pairs through independent LoRA training and parameter-space merging. Our GAM operator aligns singular bases and averages the aligned factors to combine these updates. With three reasoning deltas and four agentic deltas, PDA-GAM improves Qwen3-8B by 6.8 and 7.3 points on average, exceeding the strongest single delta by 2.1 and 4.3 points. The results support aggregation of complementary weak supervision beyond individual source pairs in the evaluated settings.

8 Limitations

Our experiments use one preference-data source per domain, GSM8K and MuSiQue, and a limited set of source pairs. All-subset results cover these source pairs on Qwen3-8B, but do not establish gains for arbitrary new sources. Size-based labels remain noisy, and the exact-match audit cannot resolve preferences when both responses are incorrect. The chosen-only SFT control covers the Qwen3 pair. Training-seed comparisons cover all four reasoning benchmarks but only MuSiQue for agentic search. The two algebraic properties of GAM provide no performance guarantee. We lack a theoretical account of the partially aligned regime and a criterion for predicting which deltas will compose. Other training-based aggregation methods, including gradient surgery [Yu et al., 2020], may also be effective.

Acknowledgments and Disclosure of Funding

We thank NYU High Performance Computing (HPC) and the NYU Shanghai High Performance Computing Center for providing computational resources. We also thank the NYU Shanghai Department of Computer Science, Data Science, and Engineering for supporting this research. This work received no external funding. The authors declare no competing interests.

References

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, Nicholas Joseph, Saurav Kadavath, Jackson Kernion, Tom Conerly, Sheer El-Showk, Nelson Elhage, Zac Hatfield-Dodds, Danny Hernandez, Tristan Hume, Scott Johnston, Shauna Kravec, Liane Lovitt, Neel Nanda, Catherine Olsson, Dario Amodei, Tom Brown, Jack Clark, Sam McCandlish, Chris Olah, Ben Mann, and Jared Kaplan. Training a helpful and harmless assistant with reinforcement learning from human feedback, 2022. URL https://arxiv.org/abs/2204.05862.

Collin Burns, Pavel Izmailov, Jan Hendrik Kirchner, Bowen Baker, Leo Gao, Leopold Aschenbrenner, Yining Chen, Adrien Ecoffet, Manas Joglekar, Jan Leike, Ilya Sutskever, and Jeff Wu. Weak-to-strong generalization: Eliciting strong capabilities with weak supervision, 2023. URL https://arxiv.org/abs/2312.09390.

Zhiyuan Chen and Bing Liu. *Continual Learning and Catastrophic Forgetting*. Springer International Publishing, 2018. URL https://doi.org/10.1007/978-3-031-01581-6_4.

Ziyu Chen, Yilun Zhao, Chengye Wang, Rilyn Han, Manasi Patwardhan, and Arman Cohan. Scimdr: Benchmarking and advancing scientific multimodal document reasoning, 2026. URL https://arxiv.org/abs/2603.12249.

Alexandra Chronopoulou, Matthew E. Peters, Alexander Fraser, and Jesse Dodge. AdapterSoup: Weight averaging to improve generalization of pretrained language models. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2054–2063, 2023.

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. Training verifiers to solve math word problems, 2021. URL https://arxiv.org/abs/2110.14168.

Alan Edelman, T. A. Arias, and Steven T. Smith. The geometry of algorithms with orthogonality constraints, 1998. URL https://arxiv.org/abs/physics/9806030.

Yuchen Fan, Kaiyan Zhang, Heng Zhou, Yuxin Zuo, Yu Fu, Yanxu Chen, Xinwei Long, Xuekai Zhu, Che Jiang, Yuchen Zhang, Li Kang, Cheng Huang, Gang Chen, Zhizhou He, Bingning Wang, LEI BAI, Ning Ding, and Bowen Zhou. SSRL: Self-search reinforcement learning, 2026. URL https://openreview.net/forum?id=WgRyl6fzLx.

Yuanning Feng, Sinan Wang, Zhengxiang Cheng, Yao Wan, and Dongping Chen. Are we on the right way to assessing LLM-as-a-judge?, 2025.

Scott Geng, Hamish Ivison, Chun-Liang Li, Maarten Sap, Jerry Li, Ranjay Krishna, and Pang Wei Koh. The delta learning hypothesis: Preference tuning on weak data can yield strong gains, 2025. URL https://arxiv.org/abs/2507.06187.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, Danny Wyatt, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Francisco Guzmán, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Govind Thattai, Graeme Nail, Gregoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel Kloumann, Ishan Misra, Ivan Evtimov, Jack Zhang, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Karthik Prasad, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, Khalid El-Arini, Krithika Iyer, Kshitiz Malik, Kuenley Chiu, Kunal Bhalla, Kushal Lakhotia, Lauren Rantala-Yeary, Laurens van der Maaten, Lawrence Chen, Liang Tan, Liz Jenkins, Louis Martin, Lovish Madaan, Lubo Malo, Lukas Blecher, Lukas Landzaat, Luke de Oliveira, Madeline Muzzi, Mahesh Pasupuleti, Mannat Singh, Manohar Paluri, Marcin Kardas, Maria Tsimpoukelli, Mathew Oldham, Mathieu Rita, Maya Pavlova, Melanie Kambadur, Mike Lewis, Min Si, Mitesh Kumar Singh, Mona Hassan, Naman Goyal, Narjes Torabi, Nikolay Bashlykov, Nikolay Bogoychev, Niladri Chatterji, Ning Zhang, Olivier Duchenne, Onur Çelebi, Patrick Alrassy, Pengchuan Zhang, Pengwei Li, Petar Vasic, Peter Weng, Prajjwal Bhargava, Pratik Dubal, Praveen Krishnan, Punit Singh Koura, Puxin Xu, Qing He, Qingxiao Dong, Ragavan Srinivasan, Raj Ganapathy, Ramon Calderer, Ricardo Silveira Cabral, Robert Stojnic, Roberta Raileanu, Rohan Maheswari, Rohit Girdhar, Rohit Patel, Romain Sauvestre, Ronnie Polidoro, Roshan Sumbaly, Ross Taylor, Ruan Silva, Rui Hou, Rui Wang, Saghar Hosseini, Sahana Chennabasappa, Sanjay Singh, Sean Bell, Seohyun Sonia Kim, Sergey Edunov, Shaoliang Nie, Sharan Narang, Sharath Raparthy, Sheng Shen, Shengye Wan, Shruti Bhosale, Shun Zhang, Simon Vandenhende, Soumya Batra, Spencer Whitman, Sten Sootla, Stephane Collot, Suchin Gururangan, Sydney Borodinsky, Tamar Herman, Tara Fowler, Tarek Sheasha, Thomas Georgiou, Thomas Scialom, Tobias Speckbacher, Todor Mihaylov, Tong Xiao, Ujjwal Karn, Vedanuj Goswami, Vibhor Gupta, Vignesh Ramanathan, Viktor Kerkez, Vincent Gonguet, Virginie Do, Vish Vogeti, Vítor Albiero, Vladan Petrovic, Weiwei Chu, Wenhan Xiong, Wenyin Fu, Whitney Meers, Xavier Martinet, Xiaodong Wang, Xiaofang Wang, Xiaoqing Ellen Tan, Xide Xia, Xinfeng Xie, Xuchao Jia, Xuewei Wang, Yaelle Goldschlag, Yashesh Gaur, Yasmine Babaei, Yi Wen, Yiwen Song, Yuchen Zhang, Yue Li, Yuning Mao, Zacharie Delpierre Coudert, Zheng Yan, Zhengxing Chen, Zoe Papakipos, Aaditya Singh, Aayushi Srivastava, Abha Jain, Adam Kelsey, Adam Shajnfeld, Adithya Gangidi, Adolfo Victoria, Ahuva Goldstand, Ajay Menon, Ajay Sharma, Alex Boesenberg, Alexei

Baevski, Allie Feinstein, Amanda Kallet, Amit Sangani, Amos Teo, Anam Yunus, Andrei Lupu, Andres Alvarado, Andrew Caples, Andrew Gu, Andrew Ho, Andrew Poulton, Andrew Ryan, Ankit Ramchandani, Annie Dong, Annie Franco, Anuj Goyal, Aparajita Saraf, Arkabandhu Chowdhury, Ashley Gabriel, Ashwin Bharambe, Assaf Eisenman, Azadeh Yazdan, Beau James, Ben Maurer, Benjamin Leonhardi, Bernie Huang, Beth Loyd, Beto De Paola, Bhargavi Paranjape, Bing Liu, Bo Wu, Boyu Ni, Braden Hancock, Bram Wasti, Brandon Spence, Brani Stojkovic, Brian Gamido, Britt Montalvo, Carl Parker, Carly Burton, Catalina Mejia, Ce Liu, Changhan Wang, Changkyu Kim, Chao Zhou, Chester Hu, Ching-Hsiang Chu, Chris Cai, Chris Tindal, Christoph Feichtenhofer, Cynthia Gao, Damon Civin, Dana Beaty, Daniel Kreymer, Daniel Li, David Adkins, David Xu, Davide Testuggine, Delia David, Devi Parikh, Diana Liskovich, Didem Foss, Dingkang Wang, Duc Le, Dustin Holland, Edward Dowling, Eissa Jamil, Elaine Montgomery, Eleonora Presani, Emily Hahn, Emily Wood, Eric-Tuan Le, Erik Brinkman, Esteban Arcaute, Evan Dunbar, Evan Smothers, Fei Sun, Felix Kreuk, Feng Tian, Filippos Kokkinos, Firat Ozgenel, Francesco Caggioni, Frank Kanayet, Frank Seide, Gabriela Medina Florez, Gabriella Schwarz, Gada Badeer, Georgia Swee, Gil Halpern, Grant Herman, Grigory Sizov, Guangyi, Zhang, Guna Lakshminarayanan, Hakan Inan, Hamid Shojanazeri, Han Zou, Hannah Wang, Hanwen Zha, Haroun Habeeb, Harrison Rudolph, Helen Suk, Henry Aspegren, Hunter Goldman, Hongyuan Zhan, Ibrahim Damlaj, Igor Molybog, Igor Tufanov, Ilias Leontiadis, Irina-Elena Veliche, Itai Gat, Jake Weissman, James Geboski, James Kohli, Janice Lam, Japhet Asher, Jean-Baptiste Gaya, Jeff Marcus, Jeff Tang, Jennifer Chan, Jenny Zhen, Jeremy Reizenstein, Jeremy Teboul, Jessica Zhong, Jian Jin, Jingyi Yang, Joe Cummings, Jon Carvill, Jon Shepard, Jonathan McPhie, Jonathan Torres, Josh Ginsburg, Junjie Wang, Kai Wu, Kam Hou U, Karan Saxena, Kartikay Khandelwal, Katayoun Zand, Kathy Matosich, Kaushik Veeraraghavan, Kelly Michelena, Keqian Li, Kiran Jagadeesh, Kun Huang, Kunal Chawla, Kyle Huang, Lailin Chen, Lakshya Garg, Lavender A, Leandro Silva, Lee Bell, Lei Zhang, Liangpeng Guo, Licheng Yu, Liron Moshkovich, Luca Wehrstedt, Madian Khabsa, Manav Avalani, Manish Bhatt, Martynas Mankus, Matan Hasson, Matthew Lennie, Matthias Reso, Maxim Groshev, Maxim Naumov, Maya Lathi, Meghan Keneally, Miao Liu, Michael L. Seltzer, Michal Valko, Michelle Restrepo, Mihir Patel, Mik Vyatskov, Mikayel Samvelyan, Mike Clark, Mike Macey, Mike Wang, Miquel Jubert Hermoso, Mo Metanat, Mohammad Rastegari, Munish Bansal, Nandhini Santhanam, Natascha Parks, Natasha White, Navyata Bawa, Nayan Singhal, Nick Egebo, Nicolas Usunier, Nikhil Mehta, Nikolay Pavlovich Laptev, Ning Dong, Norman Cheng, Oleg Chernoguz, Olivia Hart, Omkar Salpekar, Ozlem Kalinli, Parkin Kent, Parth Parekh, Paul Saab, Pavan Balaji, Pedro Rittner, Philip Bontrager, Pierre Roux, Piotr Dollar, Polina Zvyagina, Prashant Ratanchandani, Pritish Yuvraj, Qian Liang, Rachad Alao, Rachel Rodriguez, Rafi Ayub, Raghotham Murthy, Raghu Nayani, Rahul Mitra, Rangaprabhu Parthasarathy, Raymond Li, Rebekkah Hogan, Robin Battey, Rocky Wang, Russ Howes, Ruty Rinott, Sachin Mehta, Sachin Siby, Sai Jayesh Bondu, Samyak Datta, Sara Chugh, Sara Hunt, Sargun Dhillon, Sasha Sidorov, Satadru Pan, Saurabh Mahajan, Saurabh Verma, Seiji Yamamoto, Sharadh Ramaswamy, Shaun Lindsay, Shaun Lindsay, Sheng Feng, Shenghao Lin, Shengxin Cindy Zha, Shishir Patil, Shiva Shankar, Shuqiang Zhang, Shuqiang Zhang, Sinong Wang, Sneha Agarwal, Soji Sajuyigbe, Soumith Chintala, Stephanie Max, Stephen Chen, Steve Kehoe, Steve Satterfield, Sudarshan Govindaprasad, Sumit Gupta, Summer Deng, Sungmin Cho, Sunny Virk, Suraj Subramanian, Sy Choudhury, Sydney Goldman, Tal Remez, Tamar Glaser, Tamara Best, Thilo Koehler, Thomas Robinson, Tianhe Li, Tianjun Zhang, Tim Matthews, Timothy Chou, Tzook Shaked, Varun Vontimitta, Victoria Ajayi, Victoria Montanez, Vijai Mohan, Vinay Satish Kumar, Vishal Mangla, Vlad Ionescu, Vlad Poenaru, Vlad Tiberiu Mihailescu, Vladimir Ivanov, Wei Li, Wenchen Wang, Wenwen Jiang, Wes Bouaziz, Will Constable, Xiaocheng Tang, Xiaojian Wu, Xiaolan Wang, Xilun Wu, Xinbo Gao, Yaniv Kleinman, Yanjun Chen, Ye Hu, Ye Jia, Ye Qi, Yenda Li, Yilin Zhang, Ying Zhang, Yossi Adi, Youngjin Nam, Yu, Wang, Yu Zhao, Yuchen Hao, Yundi Qian, Yunlu Li, Yuzi He, Zach Rait, Zachary DeVito, Zef Rosnbrick, Zhaoduo Wen, Zhenyu Yang, Zhiwei Zhao, and Zhiyu Ma. The llama 3 herd of models, 2024. URL `https://arxiv.org/abs/2407.21783`.

Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Peiyi Wang, Qihao Zhu, Runxin Xu, Ruoyu Zhang, Shirong Ma, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, Aixin Liu, Bing Xue, Bingxuan Wang, Bochao Wu, Bei Feng, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chong Ruan, Damai Dai, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, Fuli Luo, Guangbo Hao, Guanting Chen, Guowei Li, H. Zhang, Hanwei Xu, Honghui Ding, Huazuo Gao, Hui Qu, Hui Li, Jianzhong Guo, Jiashi Li, Jingchang Chen, Jingyang Yuan, Jinhao Tu, Junjie Qiu, Junlong Li, J. L. Cai, Jiaqi Ni, Jian Liang,

Jin Chen, Kai Dong, Kai Hu, Kaichao You, Kaige Gao, Kang Guan, Kexin Huang, Kuai Yu, Lean Wang, Lecong Zhang, Liang Zhao, Litong Wang, Liyue Zhang, Lei Xu, Leyi Xia, Mingchuan Zhang, Minghua Zhang, Minghui Tang, Mingxu Zhou, Meng Li, Miaojun Wang, Mingming Li, Ning Tian, Panpan Huang, Peng Zhang, Qiancheng Wang, Qinyu Chen, Qiushi Du, Ruiqi Ge, Ruisong Zhang, Ruizhe Pan, Runji Wang, R. J. Chen, R. L. Jin, Ruyi Chen, Shanghao Lu, Shangyan Zhou, Shanhuang Chen, Shengfeng Ye, Shiyu Wang, Shuiping Yu, Shunfeng Zhou, Shuting Pan, S. S. Li, Shuang Zhou, Shaoqing Wu, Tao Yun, Tian Pei, Tianyu Sun, T. Wang, Wangding Zeng, Wen Liu, Wenfeng Liang, Wenjun Gao, Wenqin Yu, Wentao Zhang, W. L. Xiao, Wei An, Xiaodong Liu, Xiaohan Wang, Xiaokang Chen, Xiaotao Nie, Xin Cheng, Xin Liu, Xin Xie, Xingchao Liu, Xinyu Yang, Xinyuan Li, Xuecheng Su, Xuheng Lin, X. Q. Li, Xiangyue Jin, Xiaojin Shen, Xiaosha Chen, Xiaowen Sun, Xiaoxiang Wang, Xinnan Song, Xinyi Zhou, Xianzu Wang, Xinxia Shan, Y. K. Li, Y. Q. Wang, Y. X. Wei, Yang Zhang, Yanhong Xu, Yao Li, Yao Zhao, Yaofeng Sun, Yaohui Wang, Yi Yu, Yichao Zhang, Yifan Shi, Yiliang Xiong, Ying He, Yishi Piao, Yisong Wang, Yixuan Tan, Yiyang Ma, Yiyuan Liu, Yongqiang Guo, Yuan Ou, Yuduan Wang, Yue Gong, Yuheng Zou, Yujia He, Yunfan Xiong, Yuxiang Luo, Yuxiang You, Yuxuan Liu, Yuyang Zhou, Y. X. Zhu, Yanping Huang, Yaohui Li, Yi Zheng, Yuchen Zhu, Yunxian Ma, Ying Tang, Yukun Zha, Yuting Yan, Z. Z. Ren, Zehui Ren, Zhangli Sha, Zhe Fu, Zhean Xu, Zhenda Xie, Zhengyan Zhang, Zhewen Hao, Zhicheng Ma, Zhigang Yan, Zhiyu Wu, Zihui Gu, Zijia Zhu, Zijun Liu, Zilin Li, Ziwei Xie, Ziyang Song, Zizheng Pan, Zhen Huang, Zhipeng Xu, Zhongyu Zhang, and Zhen Zhang. Deepseek-r1 incentivizes reasoning in llms through reinforcement learning. *Nature*, 2025. URL http://dx.doi.org/10.1038/s41586-025-09422-z.

Peter Hase, Mohit Bansal, Peter Clark, and Sarah Wiegreffe. The unreasonable effectiveness of easy training data for hard tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2024. URL https://aclanthology.org/2024.acl-long.378/.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps, 2020. URL https://arxiv.org/abs/2011.01060.

Jiwoo Hong, Noah Lee, and James Thorne. Orpo: Monolithic preference optimization without reference model, 2024. URL https://arxiv.org/abs/2403.07691.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. Lora: Low-rank adaptation of large language models, 2021. URL https://arxiv.org/abs/2106.09685.

Zhengyu Hu, Jieyu Zhang, Zhihan Xiong, Alexander Ratner, Kaize Ding, and Ranjay Krishna. Towards acyclic preference evaluation of language models via multiple evaluators. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 21903–21911, 2026.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. Lorahub: Efficient cross-task generalization via dynamic loRA composition. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=TrloAXEJ2B.

John R. Hurley and Raymond B. Cattell. The procrustes program: Producing direct rotation to test a hypothesized factor structure. *Behavioral Science*, 7(2):258–262, 1962. doi: https://doi.org/10.1002/bs.3830070216. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/bs.3830070216.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic, 2023. URL https://arxiv.org/abs/2212.04089.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning, 2025a. URL https://arxiv.org/abs/2503.09516.

Jiahe Jin, Abhijay Paladugu, and Chenyan Xiong. Beneficial reasoning behaviors in agentic search and effective post-training to obtain them, 2026. URL https://arxiv.org/abs/2510.06534.

Xisen Jin, Xiang Ren, Daniel Preotiuc-Pietro, and Pengxiang Cheng. Dataless knowledge fusion by merging weights of language models, 2025b. URL https://arxiv.org/abs/2212.09849.

Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension, 2017. URL https://arxiv.org/abs/1705.03551.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276. URL https://aclanthology.org/Q19-1026/.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V. Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, Yuling Gu, Saumya Malik, Victoria Graf, Jena D. Hwang, Jiangjiang Yang, Ronan Le Bras, Oyvind Tafjord, Chris Wilhelm, Luca Soldaini, Noah A. Smith, Yizhong Wang, Pradeep Dasigi, and Hannaneh Hajishirzi. Tulu 3: Pushing frontiers in open language model post-training, 2025. URL https://arxiv.org/abs/2411.15124.

Ang Li, Qiugen Xiao, Peng Cao, Jian Tang, Yi Yuan, Zijie Zhao, Xiaoyuan Chen, Liang Zhang, Xiangyang Li, Kaitong Yang, Weidong Guo, Yukang Gan, Xu Yu, Daniell Wang, and Ying Shan. Hrlaif: Improvements in helpfulness and harmlessness in open-domain reinforcement learning from ai feedback, 2024a. URL https://arxiv.org/abs/2403.08309.

Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Kumar Guha, Sedrick Keh, Kushal Arora, Saurabh Garg, Rui Xin, Niklas Muennighoff, Reinhard Heckel, Jean Mercat, Mayee F Chen, Suchin Gururangan, Mitchell Wortsman, Alon Albalak, Yonatan Bitton, Marianna Nezhurina, Amro Kamal Mohamed Abbas, Cheng-Yu Hsieh, Dhruba Ghosh, Joshua P Gardner, Maciej Kilian, Hanlin Zhang, Rulin Shao, Sarah M Pratt, Sunny Sanyal, Gabriel Ilharco, Giannis Daras, Kalyani Marathe, Aaron Gokaslan, Jieyu Zhang, Khyathi Chandu, Thao Nguyen, Igor Vasiljevic, Sham M. Kakade, Shuran Song, Sujay Sanghavi, Fartash Faghri, Sewoong Oh, Luke Zettlemoyer, Kyle Lo, Alaaeldin El-Nouby, Hadi Pouransari, Alexander T Toshev, Stephanie Wang, Dirk Groeneveld, Luca Soldaini, Pang Wei Koh, Jenia Jitsev, Thomas Kollar, Alex Dimakis, Yair Carmon, Achal Dave, Ludwig Schmidt, and Vaishaal Shankar. Datacomp-LM: In search of the next generation of training sets for language models. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*, 2024b. URL https://openreview.net/forum?id=CNWdWn47IE.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step, 2023. URL https://arxiv.org/abs/2305.20050.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories, 2023. URL https://arxiv.org/abs/2212.10511.

Nvidia, :, Bo Adler, Niket Agarwal, Ashwath Aithal, Dong H. Anh, Pallab Bhattacharya, Annika Brundyn, Jared Casper, Bryan Catanzaro, Sharon Clay, Jonathan Cohen, Sirshak Das, Ayush Dattagupta, Olivier Delalleau, Leon Derczynski, Yi Dong, Daniel Egert, Ellie Evans, Aleksander Ficek, Denys Fridman, Shaona Ghosh, Boris Ginsburg, Igor Gitman, Tomasz Grzegorzek, Robert Hero, Jining Huang, Vibhu Jawa, Joseph Jennings, Aastha Jhunjhunwala, John Kamalu, Sadaf Khan, Oleksii Kuchaiev, Patrick LeGresley, Hui Li, Jiwei Liu, Zihan Liu, Eileen Long, Ameya Sunil Mahabaleshwarkar, Somshubra Majumdar, James Maki, Miguel Martinez, Maer Rodrigues de Melo, Ivan Moshkov, Deepak Narayanan, Sean Narenthiran, Jesus Navarro, Phong Nguyen, Osvald Nitski, Vahid Noroozi, Guruprasad Nutheti, Christopher Parisien, Jupinder Parmar, Mostofa Patwary, Krzysztof Pawelec, Wei Ping, Shrimai Prabhumoye, Rajarshi Roy, Trisha Saar, Vasanth Rao Naik Sabavat, Sanjeev Satheesh, Jane Polak Scowcroft, Jason Sewall, Pavel Shamis, Gerald Shen,

Mohammad Shoeybi, Dave Sizer, Misha Smelyanskiy, Felipe Soares, Makesh Narsimhan Sreedhar, Dan Su, Sandeep Subramanian, Shengyang Sun, Shubham Toshniwal, Hao Wang, Zhilin Wang, Jiaxuan You, Jiaqi Zeng, Jimmy Zhang, Jing Zhang, Vivienne Zhang, Yian Zhang, and Chen Zhu. Nemotron-4 340b technical report, 2024. URL https://arxiv.org/abs/2406.11704.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. Training language models to follow instructions with human feedback, 2022. URL https://arxiv.org/abs/2203.02155.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The fineweb datasets: decanting the web for the finest text data at scale. In *Proceedings of the 38th International Conference on Neural Information Processing Systems*, 2024. URL https://dl.acm.org/doi/10.5555/3737916.3738886.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A. Smith, and Mike Lewis. Measuring and narrowing the compositionality gap in language models, 2023. URL https://arxiv.org/abs/2210.03350.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn. Direct preference optimization: Your language model is secretly a reward model, 2024. URL https://arxiv.org/abs/2305.18290.

David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023. URL https://arxiv.org/abs/2311.12022.

Shiv Shankar and Madalina Fiterau. Self-concordant preference learning from noisy labels. In *ICML 2025 Workshop on Models of Human Feedback for AI Alignment*, 2025.

George Stoica, Pratik Ramesh, Boglarka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with svd to tie the knots, 2024. URL https://arxiv.org/abs/2410.19735.

Hao Sun, Zile Qiao, Jiayan Guo, Xuanbo Fan, Yingyan Hou, Yong Jiang, Pengjun Xie, Yan Zhang, Fei Huang, and Jingren Zhou. Zerosearch: Incentivize the search capability of llms without searching, 2025. URL https://arxiv.org/abs/2505.04588.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. Musique: Multihop questions via single-hop question composition, 2022. URL https://arxiv.org/abs/2108.00573.

Evan Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, Nathan Lambert, Dustin Schwenk, Oyvind Tafjord, Taira Anderson, David Atkinson, Faeze Brahman, Christopher Clark, Pradeep Dasigi, Nouha Dziri, Allyson Ettinger, Michal Guerquin, David Heineman, Hamish Ivison, Pang Wei Koh, Jiacheng Liu, Saumya Malik, William Merrill, Lester James Validad Miranda, Jacob Morrison, Tyler Murray, Crystal Nam, Jake Poznanski, Valentina Pyatkin, Aman Rangapur, Michael Schmitz, Sam Skjonsberg, David Wadden, Christopher Wilhelm, Michael Wilson, Luke Zettlemoyer, Ali Farhadi, Noah A. Smith, and Hannaneh Hajishirzi. 2 OLMo 2 furious (COLM's version). In *Second Conference on Language Modeling*, 2025. URL https://openreview.net/forum?id=2ezugTT9kU.

Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu, Yuandong Tian, Jiantao Jiao, Jason E Weston, and Sainbayar Sukhbaatar. Meta-rewarding language models: Self-improving alignment with LLM-as-a-meta-judge. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, November 2025. URL https://aclanthology.org/2025.emnlp-main.583/.

Ran Xu, Yuchen Zhuang, Zihan Dong, Jonathan Wang, Yue Yu, Joyce C. Ho, Linjun Zhang, Haoyu Wang, Wenqi Shi, and Carl Yang. Acesearcher: Bootstrapping reasoning and search for llms via reinforced self-play, 2025. URL https://arxiv.org/abs/2509.24193.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. Ties-merging: Resolving interference when merging models, 2023. URL https://arxiv.org/abs/2306.01708.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. Qwen3 technical report, 2025a. URL https://arxiv.org/abs/2505.09388.

Chen Yang, Ran Le, Yun Xing, Zhenwei An, Zongchao Chen, Wayne Xin Zhao, Yang Song, and Tao Zhang. Toolmind technical report: A large-scale, reasoning-enhanced tool-use dataset, 2025b. URL https://arxiv.org/abs/2511.15718.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering, 2018. URL https://arxiv.org/abs/1809.09600.

Jihan Yao, Wenxuan Ding, Shangbin Feng, Lucy Lu Wang, and Yulia Tsvetkov. Varying shades of wrong: Aligning LLMs with wrong answers only. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=p74CpDzw1Y.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. React: Synergizing reasoning and acting in language models, 2023. URL https://arxiv.org/abs/2210.03629.

Ruimeng Ye, Zihan Wang, Yang Xiao, Zinan Ling, Manling Li, and Bo Hui. Weak-to-strong generalization with failure trajectories. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=TXZ54qxdAF.

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, 2024. URL https://arxiv.org/abs/2311.03099.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, 2020.

Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho, Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason Weston. Self-rewarding language models, 2025. URL https://arxiv.org/abs/2401.10020.

Siyue Zhang, Yilun Zhao, Liyuan Geng, Arman Cohan, Anh Tuan Luu, and Chen Zhao. Diffusion vs. autoregressive language models: A text embedding perspective. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, 2025. URL https://aclanthology.org/2025.emnlp-main.213/.

Siyue Zhang, Yuan Gao, Xiao Zhou, Yilun Zhao, Tingyu Song, Arman Cohan, Anh Tuan Luu, and Chen Zhao. MRMR: A realistic and expert-level multidisciplinary benchmark for reasoning-intensive multimodal retrieval. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=XZNXSM4rHG.

Xu Zhang, Xunjian Yin, and Xiaojun Wan. ContraSolver: Self-alignment of language models by resolving internal preference contradictions, 2024.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging llm-as-a-judge with mt-bench and chatbot arena, 2023. URL https://arxiv.org/abs/2306.05685.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, Zheyan Luo, Zhangchi Feng, and Yongqiang Ma. Llamafactory: Unified efficient fine-tuning of 100+ language models, 2024. URL https://arxiv.org/abs/2403.13372.

Wenhong Zhu, Zhiwei He, Xiaofeng Wang, Pengfei Liu, and Rui Wang. Weak-to-strong preference optimization: Stealing reward from weak aligned model. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=f7KxfUrRSb.

A Performance of Student and Supervisor Models

To provide a comprehensive view of our experimental setup, we report the absolute zero-shot and few-shot performance of all models (both supervisors and students) used in this study. This serves two purposes: first, to confirm that our supervisor models exhibit a measurable performance gap (*i.e.,* preference delta) between the weak and weaker variants; second, to verify that the student models (Qwen3-8B and Tülu3-8B) consistently outperform their respective supervisors, ensuring a valid weak-to-strong supervision setting.

As shown in Table 4, the intended performance hierarchy is consistently observed. For general-purpose model families like Qwen3 and Llama-3.2, we report performance across all eight benchmarks. For domain-specialized families such as DeepSeek-R1 (reasoning-focused) and the agentic-search models Search-R1 and AceSearcher, we report performance only on their respective target domains. This unified absolute performance table provides the necessary context for the relative improvements discussed in the main results (Table 2).

Table 4: Performance of Student and Supervisor Models. We report zero-shot/few-shot metrics across Knowledge Reasoning and Agentic Search benchmarks. The capability hierarchy (Weaker $<$ Weak $<$ Student) is maintained across all model families, validating the data setup for preference extraction and aggregation.

Model	**Knowledge Reasoning**					**Agentic Search**				
	MAT.	**GPQ.**	**MML.**	**GSM.**	**Avg.**	**Hot.**	**2Wiki**	**Bam.**	**Mus.**	**Avg.**
Qwen3-8B	81.5	42.1	78.5	88.0	72.5	53.8	58.1	61.6	28.9	50.6
Tülu3-8B	78.2	38.5	75.2	85.1	69.3	52.4	55.1	58.8	25.6	48.0
DS-R1-Distill-1.5B (Weaker)	58.4	32.1	60.5	70.2	55.3	–	–	–	–	–
DS-R1-Distill-7B (Weak)	72.5	39.4	73.1	83.6	67.2	–	–	–	–	–
Qwen3-1.7B (Weaker)	56.2	30.5	58.3	68.4	53.4	41.2	43.5	46.8	18.2	37.4
Qwen3-4B (Weak)	68.1	36.8	67.2	78.5	62.7	48.5	51.2	53.4	23.6	44.2
Llama-3.2-1B (Weaker)	48.5	26.4	52.1	62.3	47.3	36.4	39.8	42.1	15.6	33.5
Llama-3.2-3B (Weak)	62.3	33.5	64.8	74.1	58.7	46.2	49.5	51.8	22.3	42.5
Search-R1-3B (Weaker)	–	–	–	–	–	42.5	46.1	49.3	19.5	39.4
Search-R1-7B (Weak)	–	–	–	–	–	50.1	54.3	57.2	26.4	47.0
AceSearcher-3B (Weaker)	–	–	–	–	–	41.8	44.2	47.5	18.8	38.1
AceSearcher-7B (Weak)	–	–	–	–	–	49.5	52.8	55.6	25.1	45.8

B Geometry of the Trained Adapters

We report two geometric quantities measured on the trained adapters of Qwen3-8B, averaged over all pairs of adapters and all 252 adapted matrices (Table 5). The cosine between flattened weight updates $\Delta W_i = B_i A_i$ measures whether two adapters change the weights in similar directions. It is invariant to the rotations used by GAM, because these rotations leave each ΔW_i unchanged (Appendix C). The distance of each Procrustes rotation from the identity measures how differently two adapters orient their singular bases, which is the quantity that GAM changes.

Table 5: Geometry of the trained adapters. $\|R - I\|_F/\sqrt{2r}$ is normalized so that it equals 0 for the identity and has expected squared value 1 for a Haar-random orthogonal matrix.

Quantity	**Knowledge Reasoning**	**Agentic Search**
Mean pairwise cosine between flattened ΔW_i	0.463	0.245
$\|R - I\|_F/\sqrt{2r}$ of the Procrustes rotations	0.514	≈ 1.0

The deltas are partially aligned. Their weight updates are far from co-linear but well above the value expected for unrelated directions in this dimension, and they are more similar on knowledge reasoning than on agentic search. The rotations are far from the identity in both domains. On agentic

search their distance from the identity is close to that expected for a random rotation, so averaging the singular bases without alignment combines vectors with largely unrelated orientations. On knowledge reasoning the rotations lie about halfway between the identity and a random rotation.

C Exact Properties and Geometric Checks of GAM

This appendix states two properties that hold exactly for GAM as defined in Section 4. They locate where GAM and naive averaging can differ. They do not show that the difference favors GAM, and that evidence rests on the ablation in Appendix E. We do not model the adapters as noisy copies of one shared update, since the cosines in Table 5 are far from the co-linear value such a model implies.

Property 1 (rotation leaves each adapter unchanged). For every i, $\tilde{U}_i\tilde{S}_i\tilde{V}_i^\top = \Delta W_i$. Using Equation (3) and $RR^\top = I$ for any orthogonal R,

$$\tilde{U}_i\tilde{S}_i\tilde{V}_i^\top = U_i R_i^U (R_i^U)^\top S_i R_i^V (R_i^V)^\top V_i^\top = U_i S_i V_i^\top = \Delta W_i.$$

GAM and naive averaging therefore combine the same n matrices. Any difference between them comes from how the aligned factors are combined, not from a change in what each adapter represents.

Property 2 (reduction to naive averaging). Suppose the left and right subspaces of all adapters coincide, so that $U_i = U_1 Q_i$ and $V_i = V_1 Q_i'$ for orthogonal $Q_i, Q_i' \in O(r)$. Then $\Delta W_{\text{GAM}} = \frac{1}{n}\sum_{i=1}^n \Delta W_i$. The Procrustes problem $\min_{R\in O(r)} \|U_1 Q_i R - U_1\|_F$ attains zero error at $R_i^U = Q_i^\top$, and this solution is unique because $U_1^\top U_i = Q_i$ is nonsingular. Likewise $R_i^V = Q_i'^\top$. Hence $\tilde{U}_i = U_1$, $\tilde{V}_i = V_1$, and $\tilde{S}_i = Q_i S_i Q_i'^\top$ for every i. Averaging gives $\bar{U} = U_1$ and $\bar{V} = V_1$, so

$$\Delta W_{\text{GAM}} = U_1\Big(\frac{1}{n}\sum_{i=1}^n Q_i S_i Q_i'^\top\Big)V_1^\top = \frac{1}{n}\sum_{i=1}^n U_i S_i V_i^\top = \frac{1}{n}\sum_{i=1}^n \Delta W_i.$$

In this case the average already has rank at most r, so rank-r truncation leaves it unchanged. By the contrapositive, GAM and naive averaging can differ only when the subspaces of the adapters do not coincide. Neither property gives a performance guarantee or an error bound.

Synthetic numerical check. We construct four rank-8 updates of size 64×48 with shared left and right subspaces, independent orthogonal rotations, and singular values sampled uniformly from $[0.5, 2.0)$. Using NumPy float64 with random seed 0, we decompose each update by SVD and apply GAM. The relative Frobenius difference $\|\Delta W_{\text{GAM}} - \Delta W_{\text{AVG}}\|_F / \|\Delta W_{\text{AVG}}\|_F$ is 1.78×10^{-15}, where ΔW_{AVG} is the untruncated weight average. The script `scripts/verify_gam_properties.py` reproduces this check, including the construction and error definition.

The objective of basis alignment. For orthonormal-column matrices U and U_0, define $\langle X, Y\rangle_F = \text{tr}(X^\top Y)$. The Procrustes step has the equivalent forms

$$\begin{aligned}
\|UR - U_0\|_F^2 &= 2r - 2\langle UR, U_0\rangle_F, \\
\max_{R\in O(r)} \langle UR, U_0\rangle_F &= \|U^\top U_0\|_*, \\
\left\|\frac{UR + U_0}{2}\right\|_F^2 &= \frac{r}{2} + \frac{1}{2}\langle UR, U_0\rangle_F,
\end{aligned}$$

where $\|\cdot\|_*$ denotes the nuclear norm. To obtain the maximum, write $U^\top U_0 = P\Sigma Q^\top$ and choose $R = PQ^\top$, giving $\text{tr}(\Sigma)$. For two bases, Procrustes therefore maximizes the squared Frobenius norm of their average over orthogonal rotations, reducing cancellation in that basis average. The same statement holds for the right bases. This is a local objective for each basis-to-reference alignment. It does not establish a global optimum for the multi-adapter average or for the final product $\bar{U}\bar{S}\bar{V}^\top$.

A check on the trained agentic adapters. Table 6 reports all six pairs of the four agentic adapters for Qwen3-8B, measured across the 252 adapted matrices. For orthonormal bases, the singular values of $U_i^\top U_j$ are the cosines of their principal angles. We report their mean, $r^{-1}\|U_i^\top U_j\|_*$, and the

analogous quantity for V. This differs from the mean squared cosine sometimes used to measure subspace overlap. The pairwise averages are 0.280 on the left and 0.991 on the right. The normalized difference between the two merging operators is larger for pairs with lower left-subspace similarity, with Pearson correlation -0.96 across the six rows.

Table 6: Geometry and merge differences for the six agentic adapter pairs. The last column compares GAM with untruncated weight averaging. Principal-angle cosines are averaged over the retained directions and adapted matrices.

Adapter pair	**Mean cosine,** U	**Mean cosine,** V	$\frac{\|\Delta W_{\mathrm{GAM}} - \Delta W_{\mathrm{AVG}}\|_F}{\|\Delta W_{\mathrm{AVG}}\|_F}$
AceSearcher + Llama-3.2	0.429	0.993	0.186
Search-R1 + Llama-3.2	0.279	0.991	0.306
Search-R1 + Qwen3	0.269	0.991	0.361
Search-R1 + AceSearcher	0.260	0.990	0.343
Qwen3 + Llama-3.2	0.225	0.990	0.427
Qwen3 + AceSearcher	0.218	0.990	0.441

The relation has one inversion in the ordering, and the six pairs are not independent because each adapter appears in three pairs. We therefore treat this as a descriptive check in the agentic setting. It links subspace geometry to the difference between the operators, but does not show that the difference improves task performance. That comparison is evaluated separately in Appendix E.

D Training Variance and Bootstrap Intervals

We retrain every preference delta with three training seeds (42, 100, and 31415) while holding the preference data, hyperparameters, and evaluation sets fixed. Decoding is greedy, so the variation reflects training alone. Seed 42 is the run reported in the main tables. Within each seed, GAM and TIES-Merging are applied to the same adapters, so we report the paired difference together with the individual scores (Table 7). On agentic search the study covers MuSiQue at 4Δ, and on knowledge reasoning it covers the four-benchmark average at 3Δ. Both use Qwen3-8B as the student.

Table 7: GAM against TIES-Merging across three training seeds. The last row reports the mean and the standard deviation across seeds.

	Agentic Search, MuSiQue, 4Δ			**Knowledge Reasoning, Avg.,** 3Δ		
Seed	**GAM**	**TIES**	**Difference**	**GAM**	**TIES**	**Difference**
42 (reported)	38.1	37.8	$+0.30$	79.25	77.67	$+1.58$
100	41.4	40.8	$+0.60$	79.33	77.55	$+1.78$
31415	39.6	38.5	$+1.10$	80.17	79.22	$+0.95$
Mean $\pm$ sd	39.70 ± 1.65	39.03 ± 1.57	$+0.67 \pm 0.40$	79.58 ± 0.51	78.15 ± 0.93	$+1.43 \pm 0.43$

On knowledge reasoning, the difference is positive on each of the four benchmarks for every seed. The per-benchmark differences for MATH, GPQA, MMLU, and GSM8K are $+1.5$, $+1.5$, $+1.6$, and $+1.7$ for seed 42, $+1.9$, $+1.4$, $+2.3$, and $+1.5$ for seed 100, and $+1.8$, $+0.5$, $+0.1$, and $+1.4$ for seed 31415. On agentic search, the MuSiQue score of each method varies by 3.0 to 3.3 points across seeds, while the paired difference has a standard deviation of 0.40. Most of the seed variation is therefore shared by the two methods built from the same adapters.

Evaluation-set uncertainty. We also report paired bootstrap intervals for two fixed-model comparisons with Qwen3-8B (Table 8). We resample evaluation questions with replacement 10,000 times, keeping the two systems' scores for each question paired, and recompute the difference in average score. The agentic comparison uses 3,500 questions, with 500 from each of seven benchmarks. The rotation comparison uses 500 MuSiQue questions and the norm-matched control defined in Appendix E. These 95% confidence intervals quantify uncertainty from the evaluation questions for the fixed trained models. They do not include training-seed variation, which is reported separately in Table 7.

Table 8: Paired bootstrap comparisons using 10,000 resamples. Differences and intervals are in F1 points. Both comparisons use four agentic deltas and Qwen3-8B.

Comparison and evaluation set	Difference	95% CI	Questions
GAM vs. TIES, seven-benchmark average	$+0.62$	$[+0.08, +1.18]$	3,500
GAM vs. norm-matched control, MuSiQue	$+0.97$	$[+0.26, +1.92]$	500

E Separating Rotation from Truncation

Table 9 compares merging variants on the same four agentic adapters (Qwen3-8B, 4Δ, seed 42) using MuSiQue F1. The componentwise average sets every rotation in GAM to the identity. The norm-matched control rescales this average separately for each adapted matrix to match the Frobenius norm of the corresponding GAM update. Componentwise averaging and truncated weight averaging are different operators despite obtaining the same score here. GAM exceeds the norm-matched control by 0.97 points, with a paired bootstrap 95% confidence interval of $[+0.26, +1.92]$ (Table 8). Its score of 38.15 was obtained by re-evaluating the model reported as 38.1 in Table 2.

Table 9: Comparison of merging variants for four agentic adapters. Changes are relative to the preceding row. MuSiQue F1, Qwen3-8B.

Merge variant	Operation or control	MuSiQue	Change
Averaging, untruncated	–	37.0	–
Averaging, truncated to rank r (PDA-Averaging)	rank-r truncation	36.9	-0.1
Componentwise average of U_i, S_i, V_i	separate averaging, no rotation	36.9	0.0
Componentwise average, norm-matched	norm matching	37.18	$+0.28$
GAM	Procrustes rotation	38.15	$+0.97$

Table 10 varies the merge rank while fixing the training rank at 64. GAM exceeds rank-matched weight averaging by 1.2 to 1.7 points across the evaluated ranks. Both methods degrade below rank 32. We retain the prespecified merge rank of 64 for the main results.

Table 10: Merge-rank sweep on four agentic adapters (MuSiQue F1, Qwen3-8B). The training rank is 64 in all rows.

Merge rank	Averaging (truncated)	GAM	Difference
Untruncated	37.0	–	–
64 (reported)	36.9	38.15	$+1.25$
32	37.1	38.8	$+1.7$
16	33.8	35.2	$+1.4$
8	31.3	32.5	$+1.2$

F Training Budget

Table 11 reports the agentic training budget for Qwen3-8B. The pair counts include a 5% validation split, which is excluded from optimization. Joint training and PDA use the same 30,608 available pairs, with 227 and 225 optimization steps, respectively. Four epochs of the strongest single delta reuse its training partition rather than adding independent data. Its average F1 is 53.7, 53.9, 54.1, and 54.1 after epochs one to four. PDA trains four independent rank-64 adapters, increasing the total number of fitted parameters while retaining a single adapter at inference.

G Preference Label Audit

Table 12 compares the chosen and rejected responses of every source pair against gold answers using exact match. A pair is separable when exactly one of its two responses is correct. The label accuracy is computed only over separable pairs and is not the accuracy over all pairs. On the remaining pairs, both responses are usually incorrect.

Table 11: Training budget on agentic search (Qwen3-8B, global batch size 128). Pair counts include the 5% held out for validation, while optimization steps exclude it.

Method	Available pairs × epochs	Optimization steps	Adapters trained	Avg. F1
Best single delta (Search-R1), 1 epoch	7,997	59	1	53.7
Best single delta (Search-R1), 4 epochs	31,988 ($4 \times 7{,}997$)	236	1	54.1
Sequential ORPO (2Δ)	15,635	115	1	47.9
Joint ORPO (4Δ)	30,608	227	1	52.8
PDA-GAM (4Δ)	30,608	225 ($59 + 58 + 56 + 52$)	4	58.0

Table 12: Preference label audit by exact match for all seven source pairs.

Pair (weak / weaker)	Domain	EM, chosen − rejected	Separable pairs	Label accuracy on separable pairs
Search-R1 7B / 3B	Agentic	+16.28	21.3%	88.2%
Qwen3 4B / 1.7B	Agentic	+14.97	19.2%	89.0%
AceSearcher 7B / 3B	Agentic	+14.35	20.8%	84.5%
Llama-3.2 3B / 1B	Agentic	+13.76	17.6%	89.1%
DeepSeek-R1 7B / 1.5B	Reasoning	+7.68	23.6%	66.3%
Qwen3 4B / 1.7B	Reasoning	+6.83	20.3%	66.8%
Llama-3.2 3B / 1B	Reasoning	+6.27	18.8%	66.7%

H All Delta Subsets

Tables 13 and 14 report all seven nonempty subsets of the reasoning deltas and all fifteen nonempty subsets of the agentic deltas for Qwen3-8B. Each multi-delta subset is merged with GAM. Single deltas and configurations shared with the main tables use the values in Tables 1 and 2.

Averaging the reported scores over all subsets of each size gives 75.07, 77.40, and 79.30 for one to three reasoning deltas. The corresponding agentic averages are 52.35, 53.72, 55.73, and 58.00 for one to four deltas. The agentic sequence starting with the weakest delta also improves, from 50.9 (D) to 52.2 (C+D), 54.2 (B+C+D), and 58.0 (A+B+C+D). The growth in these experiments therefore does not depend on selecting only the strongest-first sequence.

Every two-delta merge exceeds its strongest constituent, and each larger merge exceeds every proper subset it contains in these tables. Source quality still matters: the weakest two deltas together score below the strongest single delta in both domains, at 76.1 versus 77.2 on reasoning and 52.2 versus 53.7 on agentic search. The best agentic two-delta subset (54.8) also exceeds the weakest three-delta subset (54.2). These comparisons support gains from combining the evaluated sources without implying that delta count alone determines performance.

Table 13: GAM on every subset of reasoning deltas (Exact Match, Qwen3-8B). A is DeepSeek-R1, B is Qwen3, and C is Llama-3.2.

Deltas	MATH	GPQA	MMLU	GSM8K	Avg.
A	87.2	46.5	81.8	93.4	77.2
B	84.5	44.0	80.2	90.2	74.7
C	82.8	42.6	79.1	88.5	73.3
A + B	88.5	47.9	82.4	94.2	78.3
A + C	87.8	47.1	82.4	94.0	77.8
B + C	86.3	45.5	80.6	91.8	76.1
A + B + C	89.6	48.8	83.1	95.5	79.3

I Merge Cost

Table 15 reports merge times on an A800 GPU with TF32 disabled, using the median of five runs. The single-projection comparison uses a 4096×12288 matrix, four adapters, and rank 64. We distinguish the implementation that forms the dense update before SVD from the implementation that operates on LoRA factors. For GAM, the latter uses the QR-based thin SVD described in Section 4. The dense implementation was used for the reported task results, while the factor implementation provides an algebraically equivalent computation.

Table 14: GAM on every subset of agentic deltas (F1, Qwen3-8B). A is Search-R1, B is Qwen3, C is AceSearcher, and D is Llama-3.2.

Deltas	NQ	TQA	Pop.	Hot.	2Wiki	Bam.	MuSi.	Avg.
A	49.1	63.2	47.9	54.4	62.6	64.0	34.7	53.7
B	48.5	62.8	47.0	54.2	61.5	63.5	32.8	52.9
C	47.8	62.1	46.2	54.1	60.1	62.8	30.5	51.9
D	46.5	61.6	44.9	53.9	58.5	62.0	29.1	50.9
A + B	50.4	64.1	48.7	55.3	63.5	65.7	36.1	54.8
A + C	50.1	63.5	48.2	55.0	62.9	65.1	35.8	54.4
A + D	49.7	63.2	48.1	54.5	62.6	64.4	35.2	54.0
B + C	49.2	63.0	47.8	54.6	62.1	64.6	34.1	53.6
B + D	49.0	62.9	47.3	54.3	61.8	64.1	33.4	53.3
C + D	47.1	62.4	46.6	54.1	60.6	63.6	31.2	52.2
A + B + C	52.8	65.7	51.6	58.6	65.6	69.8	37.9	57.4
A + B + D	51.7	64.9	50.1	56.5	64.7	67.4	36.8	56.0
A + C + D	50.9	64.3	49.6	55.8	64.0	66.2	36.3	55.3
B + C + D	50.6	63.4	48.3	55.1	62.6	65.2	34.3	54.2
A + B + C + D	53.2	65.9	52.4	59.3	65.9	71.2	38.1	58.0

The memory statistic is the size of the largest intermediate tensor. It is not total peak GPU memory, which also includes simultaneously resident tensors and implementation workspace. Over all 252 adapted matrices of Qwen3-8B, the measured whole-model merge times are 0.76 s for TIES-Merging, 4.41 s for factor-based Averaging, and 6.42 s for factor-based GAM. These are measurements across the model's actual matrix shapes, rather than extrapolations from the single projection. Merging is performed offline once, and the result is stored as a single rank-64 adapter.

Table 15: Merge cost for one 4096×12288 projection with four rank-64 adapters. Memory denotes the largest intermediate tensor, not total peak device memory.

Operator	Computation	Time (ms)	Largest tensor (MB)
Averaging	Dense update, then SVD	3882	1007
Averaging	QR of LoRA factors	19.7	17
TIES-Merging	LoRA factors, no SVD	3.4	17
GAM	Dense update, then SVD	16567	1007
GAM	QR of LoRA factors	27.5	17

J Merging Operators at Each Delta Count

Table 16 compares four merging operators on the same nested subsets of agentic deltas for Qwen3-8B. Deltas are included in the order Search-R1, Qwen3, AceSearcher, and Llama-3.2, matching Table 2. GAM exceeds KnOTS by 0.7, 1.6, and 1.2 points with two, three, and four deltas, respectively. TIES-Merging is strongest with two deltas, while GAM is strongest with three and four. This comparison describes the operators' task performance without attributing the differences to any single internal operation.

Table 16: Seven-benchmark average F1 for agentic search with Qwen3-8B. All operators merge the same adapters at each delta count. Averaging is truncated to rank 64.

Operator	2Δ	3Δ	4Δ
PDA-Averaging	54.5	56.4	56.9
PDA-KnOTS	54.1	55.8	56.8
PDA-TIES	**55.5**	57.1	57.4
PDA-GAM	54.8	**57.4**	**58.0**

K Experimental and Training Details

In this section, we outline the hyperparameters, environmental setups, and data processing configurations used to extract preference deltas and evaluate the aggregated models. All training procedures are conducted on 4 NVIDIA H20 (96GB) GPUs.

K.1 Optimization and Hyperparameters

To parameterize the preference deltas efficiently, we apply Low-Rank Adaptation (LoRA) across all base models. The LoRA modules are systematically attached to the attention projection matrices (`q_proj`, `k_proj`, `v_proj`, `o_proj`) and MLP layers (`gate_proj`, `up_proj`, `down_proj`) across all transformer blocks.

To improve clarity and reproducibility, the detailed optimization settings and hyperparameter configurations shared across both task domains are summarized in Table 17. Gradient checkpointing is enabled in all runs to optimize memory consumption.

Table 17: Hyperparameters for Preference Delta Extraction. These configurations are consistently applied across both Knowledge Reasoning and Agentic Search domains unless otherwise specified.

Hyperparameter	**Value**
LoRA Rank (r)	64
LoRA Alpha (α)	128
LoRA Dropout	0.05
Target Modules	`q, k, v, o, gate, up, down_proj`
Optimizer	AdamW
Peak Learning Rate	1.0×10^{-5}
Learning Rate Scheduler	Cosine
Warmup Ratio	10%
Global Batch Size	128
Training Epochs	1
Max Sequence Length	8192
Preference Objective (Reasoning)	DPO ($\beta = 0.1$)
Preference Objective (Agentic)	ORPO ($\beta = 0.1$)

K.2 Knowledge Reasoning Details (DPO Setup)

For the knowledge reasoning domain, the base models are fine-tuned via Direct Preference Optimization (DPO). The preference datasets are sampled from the training splits of the respective mathematical and reasoning benchmarks. During DPO tuning, the prompts are formatted strictly according to the default chat templates of the respective base models (e.g., Qwen and Llama). To encourage explicit logical derivation, a standard system prompt is utilized to instruct the model to provide step-by-step reasoning (e.g., "Let's think step by step") before formulating the final answer. This setup maintains a consistent structural format for reasoning trajectories across both the supervisor and student models.

K.3 Agentic Search and Retrieval Details (ORPO Setup)

The preference data is constructed using weak-weaker model pairs and 8,000 queries randomly sampled from the MuSiQue training set. Unlike the reasoning setup, agentic search trajectories require strict structural formatting. We apply a formatting filter only to the weak model's (chosen) responses, while retaining formatting errors in the weaker model's (rejected) responses to preserve the negative signal. As a result, the number of valid preference pairs varies: 7,997 for Search-R1, 7,852 for AceSearcher, 7,638 for Qwen3, and 7,121 for Llama-3.2.

Training and evaluating models for agentic search involves multi-turn interactions where the model iteratively issues search queries and processes external context to answer complex questions (e.g., HotpotQA, MuSiQue). For this domain, we utilize Odds Ratio Preference Optimization (ORPO) to construct the preference deltas.

Agentic Data Formulation. The agentic preference pairs are formatted as ReAct trajectories [Yao et al., 2023], comprising interleaved `Thought`, `Action`, and `Observation` steps. Models are explicitly optimized to output targeted action commands (e.g., `<search>query</search>`) to trigger the external retrieval tool during complex reasoning paths. The sequence length is set to 8192 tokens to accommodate the extensive context windows required by multi-hop retrieval tasks.

Retrieval Environment. To ensure reproducible and standardized evaluations, the external search environment is grounded in a fixed Wikipedia corpus (using the `enwiki-20200601` English Wikipedia dump). When a model generates a `Search` action, the environment queries the corpus using a dense retrieval model (`e5-large-v2`), which returns the top $k = 5$ most relevant passages based on embedding similarity. To manage context density and mitigate noise, each retrieved snippet is truncated to a maximum of 500 tokens before being appended to the model's context as an `Observation`. The model subsequently processes this new information to generate the next `Thought` or conclude with a final `<answer>answer</answer>`. This retrieval protocol is maintained uniformly across both the preference data generation and evaluation phases.

L Prompt Templates

In this section, we provide the exact prompt templates utilized during the preference data extraction and evaluation phases. To standardize the generation distribution and ensure strict adherence to task-specific formats, we employ a one-shot prompting strategy across both domains. The one-shot examples explicitly demonstrate the expected reasoning traces and tool-use trajectories.

L.1 Knowledge Reasoning Prompts

For mathematical and logical reasoning tasks, the one-shot prompt instructs the model to articulate its derivation process before formulating the final answer. This structural constraint facilitates the generation of detailed step-by-step preference trajectories for DPO tuning.

System Prompt:

```
You are an expert mathematical and logical reasoning assistant.
Please think step by step to solve the problem and carefully explain
your derivation before providing the final answer.
```

User Input Template (One-shot):

```
Here is an example of how to answer a question:
Question: A train travels at a constant speed of 60 miles per hour.
How far will it travel in 2.5 hours? Answer: Let's think step by
step. We know the formula for distance is Distance = Speed * Time.
The given speed is 60 miles per hour, and the given time is 2.5
hours. Multiplying these values: 60 * 2.5 = 150. Therefore, the
train will travel 150 miles. The final answer is 150.
Now, answer the following question:
Question: {question} Answer:
```

L.2 Agentic Search Prompts

For the agentic search domain, models are prompted to interact with the external dense retrieval environment (`e5-large-v2`) following the ReAct framework [Yao et al., 2023]. The one-shot example explicitly dictates the strict interleaving format of thoughts, search actions, and observations.

System Prompt (ReAct Instructions):

```
Answer the following question from the user with the help of a
Wikipedia search engine. Please reason step by step. You should
think about what you need to know in order to answer the question,
and then search for that information using the search engine. To
perform a search operation, write a web search question and enclose
```

```
it with <search> and </search>.  You will immediately observe a
piece of search results within the <information> and </information>
tags.  You can then use this retrieved information to continue your
reasoning.  You can repeat the search process many times.  Once
you think you have all the information you need, you can end the
thinking process and provide the final answer.  You MUST enclose
your final answer with <answer> and </answer>.
```

User Input Template (One-shot):

```
Here is an example of the interaction format:
<search> Who were the people that captured Malakoff?  </search>
<information> The French army under General MacMahon successfully
captured the Malakoff redoubt on 8th.  </information>
Okay, so the French people captured Malakoff.  Now, the next step
would be to figure out in what region Pilipsburg is located.  I will
write a web search to look that up.
<search> Where is Philipsburg located at?  </search> <information>
Philipsburg is is the main town and capital of Sint Maarten,
a constituent country of the Kingdom of the Netherlands.
</information>
...[more thoughts shortened]...
Your final response:  <answer> November 12, 1625 </answer>
```

M Case Studies: Knowledge Reasoning

We present four GPQA reasoning traces in which the model trained on the strongest individual preference pair (**Single Preference Delta**) fails, whereas our proposed **PDA-GAM** successfully derives the correct answer.

The failures of the single-delta model typically fall into two categories: *generation non-termination*, where the model enters a degenerate repetitive loop (Table 18); and *logical propagation errors*, where it produces an incorrect final answer due to intermediate arithmetic or domain knowledge mistakes (Tables 19–21). Across a random sample of 60 GPQA error cases from the single-delta baseline, 35 cases exhibit format/termination failures and 25 exhibit logical errors; PDA-GAM effectively recovers all 60 cases. Key reasoning steps are highlighted in gray, with critical errors in red and the recovered correct derivations in green.

Table 18: **Termination Recovery:** The Single Preference Delta model derives the correct epistasis hierarchy but enters a degenerate repetition loop—repeating the same hesitation 59 times—and fails to emit a valid `ANSWER:` tag. PDA-GAM reaches the same conclusion and terminates cleanly.

Question: You perform a high-throughput experiment on white lupine to find genes contributing to resistance to anthracnose. You receive candidate genes G1, G2, G3 and create knockouts... Which statement correctly describes the epistatic relationships? **Gold:** A
Single Preference Delta (✗ *Output:* None / Format Failure) Identifies hierarchy: G2 is epistatic to G1; G2 is epistatic to G3; G1 is epistatic to G3. "So, gene 2 is the most upstream, then gene 1, then gene 3." (*Correct conclusion*) "Wait, perhaps the correct answer is option A... But the data doesn't support that." ↪ *Repeated 59× until token limit; no answer extracted.*
PDA-GAM (✓ *Output:* A) Derives same hierarchy: G2 → G1 → G3. Identifies G2 as the transcription factor and G1/G3 as redundant. Terminates with `ANSWER: A`.

Table 19: **Arithmetic Recovery:** The Single Preference Delta model formulates the correct decay equation but makes a magnitude error during division, obtaining 10^{-25} instead of 10^{-24}. PDA-GAM executes the arithmetic correctly.

Question: X is a meson resonance. What is the mean decay distance? Given: $E_X = 8\,\text{GeV}$, $m_X = 1.2\,\text{GeV}$, $\Gamma_X = 320\,\text{MeV}$.
Gold: A

Single Preference Delta (✗ *Output:* C)

Step 1: $\gamma = E_X/m_X \approx 6.667$

Step 2: $\tau = \hbar/\Gamma_X = 6.582 \times 10^{-25}/0.32 \approx 2.057 \times 10^{-25}\,\text{s}$ (*Should be* 10^{-24})

Step 3: $L = \gamma c\tau \approx 4.11 \times 10^{-16}\,\text{m}$ → *Selects C, off by exactly* $10\times$.

PDA-GAM (✓ *Output:* A)

$\tau = \hbar/\Gamma_X = 6.582 \times 10^{-25}/0.32 = 2.057 \times 10^{-24}\,\text{s}$ (*Correct exponent*)

$d = (p/m_X) \cdot c \cdot \tau \approx 4.07 \times 10^{-15}\,\text{m}$ → *Selects A.*

Table 20: **Algebraic Recovery:** The Single Preference Delta model drops a logarithmic factor when propagating stellar abundance ratios through solar reference values. PDA-GAM successfully tracks the transformation.

Question: Two stars with $[\text{Si/Fe}]_1 = 0.3$, $[\text{Mg/Si}]_2 = 0.3$... Calculate the ratio $n_{\text{Si},1}/n_{\text{Si},2}$.
Gold: C

Single Preference Delta (✗ *Output:* D)

Star_1: $[\text{Si/Fe}]_1 = 0.3 \Rightarrow n_{\text{Si},1} = 10^{0.3} \cdot n_{\text{Si},\odot}$

Star_2: Conflates intermediate terms, dropping a factor from $\log_{10}(n_{\text{Si},\odot}/n_{\text{H}})$.

$\Rightarrow$ Yields ratio ≈ 4.0 (Selects D).

PDA-GAM (✓ *Output:* C)

Correctly tracks each bracket transformation through solar reference values.

$\Rightarrow$ Obtains ratio ≈ 12.6 (Selects C).

Table 21: **Domain Knowledge Recovery:** The Single Preference Delta model misidentifies the extent of reduction by red P + HI, treating it as a partial transformation. PDA-GAM applies the correct exhaustive reduction mechanism.

Question: What is the index of hydrogen deficiency (IHD) of the product obtained when 2-formyl-5-vinylcyclohex-3-enecarboxylic acid is reacted with red phosphorus and excess HI?
Gold: B

Single Preference Delta (✗ *Output:* D)

Identifies starting IHD $= 5$ (1 ring + 2 C=C + 2 C=O).

"Red P + HI converts -COOH to -COI. Double bonds remain unchanged."

$\Rightarrow$ Selects D (IHD $= 3$).

PDA-GAM (✓ *Output:* B)

"Red P + HI is an exhaustive reduction: reduces all C=C and C=O groups."

Product: fully saturated cyclohexane ring with alkyl substituents. IHD $= 1$.

N Case Studies: Agentic Search

To illustrate the behavioral complementation achieved by multiple preference deltas, we present four representative agentic search trajectories. Each case highlights a distinct failure mode overcome by our framework. Search queries are in green and retrieved info in blue.

Table 22: **Entity Disambiguation:** The Single Preference Delta model misidentifies the lead actor as a child actor and issues repetitive queries before generating an unsupported entity. PDA-GAM formulates a holistic query to resolve the entity accurately.

Question: Who did the leading actor in *Sammy Going South* play in *The Ten Commandments*? **Gold:** Dathan
Single Preference Delta (✗ *Output:* Mered, 6 turns)
<search> Who is the leading actor in Sammy Going South? </search>
<information> Fergus McClelland is the lead actor... </information>
Repeatedly queries Fergus McClelland's role in The Ten Commandments (which does not exist).
<answer> Mered </answer> (*Unsupported entity generation*)
PDA-GAM (✓ *Output:* Dathan, 2 turns)
<search> Who did the leading actor in Sammy Going South play in the ten commandments? </search>
<information> Edward G. Robinson starred as Dathan in The Ten Commandments... </information>
<answer> Dathan </answer>

Table 23: **Anchoring Bias Recovery:** The Single Preference Delta model exhibits anchoring bias, fixating on a 2004 film adaptation. PDA-GAM redirects its search to the 2017 series adaptation to resolve the query.

Question: The actor who plays Count Olaf in "A Series of Unfortunate Events" also plays which character in *Batman Under the Red Hood*? **Gold:** Nightwing / Dick Grayson
Single Preference Delta (✗ *Output:* The Joker, 6 turns)
<information> Jim Carrey played Count Olaf in the 2004 film... </information>
Model repeatedly searches for Jim Carrey's role in Batman, failing to consider other adaptations.
<answer> The Joker </answer> (*Hallucination due to anchoring*)
PDA-GAM (✓ *Output:* Dick Grayson, 4 turns)
<search> Which character does Neil Patrick Harris play in Batman Under the Red Hood? </search>
<information> Neil Patrick Harris voiced Dick Grayson/Nightwing... </information>
<answer> Dick Grayson/Nightwing </answer>

Table 24: **Multi-hop Chain Completion:** The Single Preference Delta model prematurely terminates the multi-hop chain at Indonesia. PDA-GAM correctly parses the full relational chain targeting Timor-Leste.

Question: Who is the president of the newly declared independent country of the country of the birthplace of Mulham Arufin?
Gold: Francisco Guterres

Single Preference Delta (✗ *Output:* Joko Widodo, 3 turns)

<information> Mulham Arufin was born in Indonesia... </information>

Queries for the president of Indonesia, failing to resolve the "newly declared independent" clause.

<answer> Joko Widodo </answer> (*Premature termination*)

PDA-GAM (✓ *Output:* Francisco Guterres, 5 turns)

<search> Who is the president of the newly declared independent country of Timor-Leste? </search>

<information> Francisco Guterres is the president of Timor-Leste... </information>

<answer> Francisco Guterres </answer>

Table 25: **Resolution Matching:** The Single Preference Delta model answers at an incorrect granularity (State instead of County). PDA-GAM identifies that the retrieved city requires an additional bridging query to map to the target county.

Question: In which county was Larkin I. Smith born?
Gold: Pearl River County

Single Preference Delta (✗ *Output:* Mississippi, 3 turns)

<information> Larkin I. Smith was a U.S. Representative from Mississippi... </information>

Fails to resolve the specific county from general state-level information.

<answer> Mississippi </answer> (*Resolution mismatch*)

PDA-GAM (✓ *Output:* Pearl River County, 3 turns)

<information> Smith was born in Poplarville, Mississippi... </information>

<search> In which county is Poplarville, Mississippi? </search> (*Bridging query*)

<information> Poplarville is located in Pearl River County... </information>

<answer> Pearl River County </answer>

O Sensitivity Analysis of Reference Choice in GAM

In Section 4, we introduce Geometric Alignment Merging (GAM), which utilizes orthogonal Procrustes alignment to project multiple LoRA adapters onto a shared reference coordinate system (ϕ_1). While the Procrustes formulation is defined relative to a chosen base, we hypothesized that the specific choice of the reference adapter does not significantly impact the final aggregated performance. To empirically verify this, we conduct a sensitivity analysis for both the Knowledge Reasoning and Agentic Search domains by systematically rotating the reference role among all available domain-specific adapters.

Knowledge Reasoning. For the reasoning domain, we aggregate three distinct preference deltas. We evaluate the aggregated performance of the student model (e.g., Qwen3-8B) by iteratively setting each of the three adapters ($\Delta_{\text{DS-R1}}$, Δ_{Qwen3}, and Δ_{Llama}) as the reference base ϕ_1. As shown in Table 26, the maximum fluctuation in the average reasoning score is minimal (within 0.1 points).

Agentic Search. Similarly, for the agentic search domain, we aggregate four distinct preference deltas. We repeat the sensitivity analysis by rotating the reference role among $\Delta_{\text{Search-R1}}$, $\Delta_{\text{AceSearcher}}$, Δ_{Qwen3}, and Δ_{Llama}. Table 27 demonstrates that the final average performance remains robust regardless of the geometric anchor chosen.

Table 26: Sensitivity to Reference Choice on Knowledge Reasoning. Performance of the aggregated model when different preference deltas are selected as the reference base (ϕ_1).

Reference Choice (ϕ_1)	**Knowledge Reasoning Avg.**
$\Delta_{\text{DS-R1}}$ (DeepSeek-R1 preference delta)	79.3
Δ_{Qwen3} (Qwen3 preference delta)	79.2
Δ_{Llama} (Llama-3.2 preference delta)	79.3
Max Fluctuation	$\leq$ **0.1**

Table 27: Sensitivity to Reference Choice on Agentic Search. Performance of the aggregated model when rotating the reference base among the four agentic preference deltas.

Reference Choice (ϕ_1)	**Agentic Search Avg. Score**
$\Delta_{\text{Search-R1}}$ (Search-R1 preference delta)	58.0
$\Delta_{\text{AceSearcher}}$ (AceSearcher preference delta)	58.0
Δ_{Qwen3} (Qwen3 preference delta)	57.9
Δ_{Llama} (Llama-3.2 preference delta)	58.0
Max Fluctuation	$\leq$ **0.1**

Across both diverse task domains and varying numbers of adapters, the geometric consistency of GAM maintains the intrinsic relative orientations between the preference deltas. This indicates that GAM is structurally robust to the initial selection of the reference coordinate system.

P Degradation from Standard Supervised Fine-Tuning

Table 28: Performance comparison between Standard SFT and preference tuning using identical source data. SFT on the chosen responses forces the strong student model to mimic a suboptimal policy, degrading its capabilities. Conversely, preference tuning effectively extracts the relative capability delta, improving overall performance.

Method	**Qwen3-8B**								**Tülu3-8B**							
	Single-hop QA			**Multi-hop QA**				**Avg.**	**Single-hop QA**			**Multi-hop QA**				**Avg.**
	NQ	**TQA**	**Pop.**	**Hot.**	**2Wiki**	**Bam.**	**MuSi.**		**NQ**	**TQA**	**Pop.**	**Hot.**	**2Wiki**	**Bam.**	**MuSi.**	
Student Baseline	46.2	61.4	44.6	53.8	58.1	61.6	28.9	50.7	45.8	60.5	43.0	52.4	55.1	58.8	25.6	48.7
Training on identical Qwen3 (4B over 1.7B) source data																
Standard SFT (Chosen only)	44.6	60.8	43.1	51.9	54.8	58.8	25.8	48.5	41.7	56.2	37.4	48.6	51.2	51.6	19.7	43.8
Single Preference Delta	**48.5**	**62.8**	**47.0**	**54.2**	**61.5**	**63.5**	**32.8**	**52.9**	**46.8**	**62.0**	**46.8**	**53.5**	**59.2**	**62.1**	**28.5**	**51.3**

In our main experiments, we claim that the performance gains achieved by our framework reflect genuine capability elicitation from preference deltas, rather than simply benefiting from exposure to the task format or identical data. To empirically validate this, we conduct a control experiment using standard Supervised Fine-Tuning (SFT) under the same parameter-efficient setup.

Specifically, we extract the "chosen" responses from the Qwen3 (4B over 1.7B) preference pair dataset. Instead of applying preference optimization, we use standard SFT to train the strong student models (Qwen3-8B and Tülu3-8B) to directly mimic these chosen responses via behavioral cloning.

Implementation Details for SFT Baseline. To ensure a rigorous and fair comparison, we maintain identical hyperparameter settings for both the SFT baseline and our preference tuning experiments. Specifically, we use the LoRA [Hu et al., 2021] framework with a rank of $r = 64$ and $\alpha = 128$, targeting all linear layers. The models are trained using the AdamW optimizer with a peak learning rate of 1.0×10^{-5} and a cosine learning rate scheduler. We use a total batch size of 128 and train for 1 epoch, which we found to be sufficient for the model to converge on the formatting of the source data without excessive overfitting. All experiments are conducted on 4×NVIDIA H20 (96GB) GPUs using the LlamaFactory [Zheng et al., 2024] framework.

The results are presented in Table 28. As expected, SFT leads to a noticeable degradation in performance across almost all tasks. For Qwen3-8B, the average F1 score drops from the baseline of 50.7 to 48.5. The degradation is even more pronounced for Tülu3-8B, which drops from 48.7 to 43.8.

This phenomenon occurs because the absolute response quality of the weaker model (4B) is inherently lower than the intrinsic capabilities of the strong student models (8B). SFT encourages the strong students to mimic the suboptimal reasoning trajectories and stylistic idiosyncrasies of the weaker model, leading to capability regression. In contrast, preference optimization on the identical data focuses on the *relative directional delta*. This allows the strong student model to extract the underlying capability improvements without being bottlenecked by the weak model's absolute generation quality, successfully raising the average F1 to 52.9 and 51.3 for Qwen3-8B and Tülu3-8B, respectively.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: We clearly state our contributions, including the Preference Delta Aggregation framework and the Geometric Alignment Merging method, in the Abstract and Section 1, and support them with empirical results in Section 5. The claim about Geometric Alignment Merging is stated per number of deltas, including the setting in which TIES-Merging is stronger.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations in Section 8. They cover the noise in size-based preference labels, the absence of a theoretical account of when deltas compose, the restriction of the agentic seed study to MuSiQue, the use of one source task per domain, and the untested training-based alternatives.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: The only theoretical statements are two elementary properties of Geometric Alignment Merging in Appendix C. Each states its assumption explicitly and is followed by a complete derivation. The paper makes no other theoretical claims and does not present error bounds.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide detailed descriptions of our experimental setup, prompt templates, baseline implementations, and hyperparameter choices in Section 5 and Appendix K.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code and data are released at `https://github.com/AlbertQiSun/Preference-Delta-Aggregation`. Implementation and experimental settings are described in Section 5 and Appendix K.

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: All necessary training details, including learning rates, batch sizes, LoRA configurations, and optimizer setups, are comprehensively documented in Appendix K, Appendix L and Section 5.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

Justification: Appendix D reports three training seeds for the comparison between Geometric Alignment Merging and TIES-Merging, including individual scores and paired differences with their means and sample standard deviations. Data and evaluation settings are fixed. The agentic training-seed study covers MuSiQue only, as stated in Section 5.4. Appendix D also reports 95% paired bootstrap confidence intervals from 10,000 resamples for two fixed-model comparisons, with evaluation-set uncertainty distinguished from training variance. Other table entries are single runs.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: Appendix K describes the training and evaluation hardware and settings. Appendix I reports merge timing on an A800 for dense and factor-based implementations, whole-model merge times, and the size of the largest intermediate tensor. The latter is distinguished from total peak GPU memory.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: Our research complies strictly with the NeurIPS Code of Ethics.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Our work presents a general training and model-merging method for improving LLM reasoning and search performance, without introducing a specific deployment domain or application involving direct societal impact. It does not involve sensitive personal data, safety-critical decision making, or targeted high-risk use cases. Therefore, we consider broader societal impacts to be limited and not uniquely attributable to this work beyond those generally associated with large language models.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: Our work focuses on a parameter merging methodology and utilizes existing open-source datasets and models. It does not introduce or release new high-risk foundational models or scraped datasets.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.

- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All pre-trained models (e.g., Qwen, Llama) and datasets used in this work are publicly available. We properly cite their original papers and strictly adhere to their respective usage licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The new assets are our codebase and merging scripts, released with documentation at `https://github.com/AlbertQiSun/Preference-Delta-Aggregation`.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: Our research relies purely on automated model outputs and existing datasets; it does not involve any crowdsourcing or human subjects.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Not applicable as this research does not involve human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: The generation of weak-to-strong preference trajectories using existing LLMs (e.g., Qwen3-4B over 1.7B) is explicitly detailed in Section 3, as it forms the basis of our Preference Delta Aggregation framework.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.