

PRISM: Preference-aware Influence function based Data Selection Method for Efficient Fine-Tuning

Qihao Lin^{1,2}, Guanxu Chen^{1,2}, Dongrui Liu¹, Jing Shao^{1*}

¹Shanghai Artificial Intelligence Laboratory, ²Shanghai Jiao Tong University,

Correspondence: {linqihao,shaojing}@pjlab.org.cn

Abstract

As LLMs scale, improving training efficiency relies on effective data utilization. Data selection mitigates this issue by allocating limited data budget to examples that optimally facilitate target behavior. Most existing approaches define target behavior via a set of target examples and score candidate training examples based on estimated influence on the target set. However, such methods treat target examples as equally important, ignoring their relevance to model optimization. Target examples that align closely with the model’s inherent behavior deliver stronger supervisory signals, whereas discrepant examples yield weak local guidance. We propose PRISM, a **P**reference-aware **I**nfluence function based **D**ata **S**election **M**ethod. It uses model preference to weight examples and build a preference-aware target direction. PRISM evaluates candidate training examples according to influence on this direction, and prioritizes data budget allocation to examples that drive the model to match target behavior. Theoretical analysis shows that preference weighting yields a superior first-order direction for boosting target preference, relative to equal aggregation. Experiments across model architectures and scales show that PRISM improves data selection in both capability improvement and reverse repair. The results validate that accurate characterization of target behavior serves as the core of cost-effective data selection.

1 Introduction

As LLMs scale, the costs of training and adaptation are increasing, making training efficiency a critical challenge (Brown et al., 2020; Bommasani et al., 2021). Beyond improving training algorithms (Hu et al., 2022; Dettmers et al., 2023), an important complementary direction under limited budgets is enhancing data efficiency by selectively training on the most valuable examples. Data selection

provides a principled approach to this problem by identifying the training examples that contribute most to the target behavior (Xia et al., 2024).

Data selection generally comprises two complementary steps: defining the target behavior that the model is expected to learn, typically represented by a set of target examples, and estimating the influence of each training example on that behavior. Although prior work has developed various methods to estimate influence more accurately or efficiently (Koh and Liang, 2017; Yeh et al., 2018; Pruthi et al., 2020; Kwon et al., 2024), these studies pay limited attention to precisely characterizing the target behavior itself. In particular, they often aggregate all target examples equally, ignoring their varying relevance to the model (Xia et al., 2024; Kowal et al., 2026; Chen et al., 2025; Coalson et al., 2025). As illustrated in Figure 1, such equal aggregation blends diverse signals and may yield a target direction that fails to emphasize behaviors most likely to be amplified near the model.

To address this limitation, we account for differences in target-example importance when constructing the target behavior and propose PRISM (**P**reference-aware **I**nfluence function based **D**ata **S**election **M**ethod). PRISM quantifies model preference for each target example using the likelihood assigned to the response and uses it to weight the example when constructing the target direction. A higher preference indicates that the example represents a behavioral direction closer to the model’s behavior and can therefore be strengthened more readily through training. Training examples associated with high-preference target examples are thus more likely to move the model efficiently toward the target behavior under a limited data budget. In contrast, lower-preference examples provide weaker selection signals. PRISM then scores candidates by their influence on this preference-aware target direction, allocating the budget to examples that strengthen the target behavior efficiently.

*Corresponding Author.

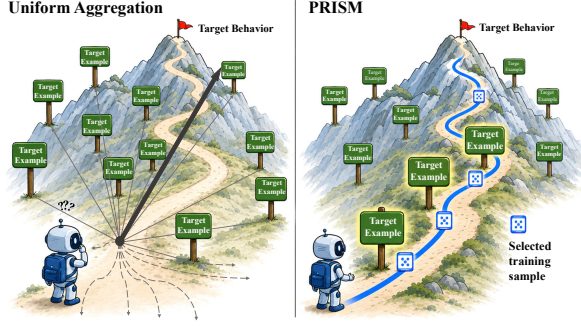


Figure 1: Motivation of PRISM. Equal aggregation treats all target examples equally, whereas PRISM emphasizes target examples closer to the model and focuses the budget on more actionable training examples.

Theoretically, PRISM derives the locally reward-maximizing first-order direction for the paired KL target-preference reward encoded by the supplied target pairs. Under fixed local update and selection budgets, ranking examples along this direction maximizes the first-order target-preference reward gain per unit of data. Equal aggregation instead mixes high- and low-preference target signals, which can shift selection away from this reward-maximizing direction and reduce the utility of the selected data. When the target pairs are constructed from downstream tasks, this preference-aware direction is directly aligned with the corresponding desired behavior. Experiments across capability improvement and reverse repair show that this alignment yields stronger performance across model families and scales.

We conduct systematic experiments across model families and scales in two complementary data-selection modes: capability improvement and reverse repair. In capability improvement, we define the target behavior in terms of factual knowledge, challenging reasoning, multilingual question answering, mathematical problem solving, and safety guardrail capabilities. Under the same selection budget, training examples selected by PRISM lead to stronger fine-tuning performance. In reverse repair for emergent misalignment (Betley et al., 2025), PRISM mitigates harmful model behaviors more effectively under the same removal budget. Overall, these findings demonstrate that PRISM improves the efficiency of data selection by more precisely characterizing the target behavior.

2 Methodology

We now introduce PRISM (Preference-aware Influence function based Data Selection Method),

which improves influence-based data selection by constructing the target direction in a preference-aware way. Instead of treating all target examples equally, PRISM weights them by the model’s preference and then traces the resulting target direction back to candidate training examples through influence scoring. Section 2.1 introduces the notation, Section 2.2 derives the PRISM score, and Section 2.3 provides the theoretical analysis.

2.1 Preliminaries

Problem setup. Let $\mathcal{D} = \{z_i\}_{i=1}^n$ denote a candidate training pool, where each example $z_i = (x_i, y_i)$ contains a query x_i and a training response y_i . Let p_θ denote the model distribution parameterized by θ , before training on the selected data. A target behavior is specified by a set of target queries $\mathcal{Q}$. For each $q \in \mathcal{Q}$, we use a target pair (q, y_q^+, y_q^-) : y_q^+ is the target-positive response whose behavior we want to align, and y_q^- is a target-negative response used as the contrast under the same query.

For any query-response pair (u, v) , define the sequence probability as

$$p_\theta(v | u) = \prod_{t=1}^{|v|} p_\theta(v_t | u, v_{<t}). \quad (1)$$

To avoid length bias when comparing target responses of different lengths, PRISM uses the length-normalized likelihood

$$\bar{p}_\theta(v | u) = p_\theta(v | u)^{1/|v|}. \quad (2)$$

The corresponding token-average cross-entropy loss is

$$\bar{\ell}(u, v; \theta) = -\log \bar{p}_\theta(v | u) = \frac{1}{|v|} \ell(u, v; \theta). \quad (3)$$

At the model θ , target-response gradients use the token-average loss, while candidate-training gradients follow the SFT objective:

$$\begin{aligned} \bar{g}_{q, y_q^+} &= \nabla_{\vartheta} \bar{\ell}(q, y_q; \vartheta) \big|_{\vartheta=\theta}, \\ g_{z_i} &= \nabla_{\vartheta} \ell(x_i, y_i; \vartheta) \big|_{\vartheta=\theta}. \end{aligned} \quad (4)$$

Here $\bar{g}_{q, y_q^+}$ and $\bar{g}_{q, y_q^-}$ are the length-normalized gradients for generating the positive and negative target responses from q , while g_{z_i} is the gradient for generating the training response y_i from x_i . **Local curvature.** We use H to denote the empirical

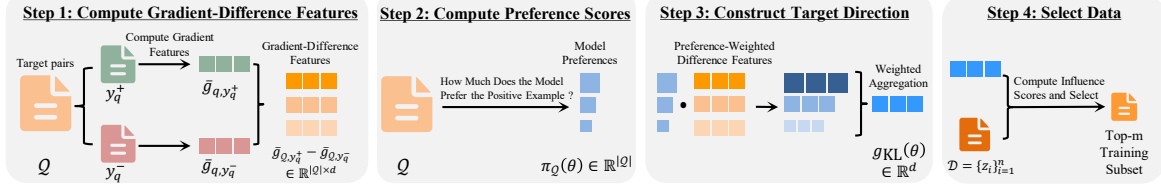


Figure 2: Overview of PRISM. Given target pairs, PRISM estimates the model’s preference for the target-positive response, constructs a preference-aware target direction, and scores candidate training examples by influence alignment with this direction.

Hessian of the training objective over the training distribution:

$$H = \frac{1}{n} \sum_{j=1}^n \nabla_{\theta}^2 \ell(x_j, y_j; \theta). \quad (5)$$

For the local influence analysis, we assume H is positive definite so that H^{-1} is well defined.

Influence-based selection form. At a high level, influence-based data selection scores each candidate training example by how well its training signal aligns with a target-behavior direction (Koh and Liang, 2017; Xia et al., 2024; Coalson et al., 2025). We denote this parameter-space representation of the target behavior by g_{target} . This common scoring form can be written as

$$s(z_i) = g_{z_i}^{\top} H^{-1} g_{\text{target}}, \quad (6)$$

where g_{target} represents the target behavior. Under this view, the key design question is how to construct g_{target} . Existing influence-based methods often set g_{target} by equally aggregating gradients from target examples (Xia et al., 2024; Coalson et al., 2025). PRISM instead derives g_{target} from a target-preference reward, yielding the preference-aware direction g_{KL} .

2.2 PRISM Framework

We now derive the PRISM score by constructing a preference-aware target direction and tracing it back to candidate training examples. Figure 2 illustrates the resulting workflow.

Model preference. PRISM uses the model’s relative length-normalized likelihood of the target-positive response as a proxy for how close this target pair is to the model’s behavior. For each target pair, we define

$$\pi_q = \frac{\bar{p}_{\theta}(y_q^+ | q)}{\bar{p}_{\theta}(y_q^+ | q) + \bar{p}_{\theta}(y_q^- | q)}. \quad (7)$$

This form is a two-response normalization over geometric-mean likelihoods: π_q is large when the

model assigns higher length-normalized likelihood to the target-positive response than to the target-negative contrast. Thus, π_q measures the model’s preference for the target-positive response within that pair. A larger π_q indicates that the target-positive response is closer to the model’s generation tendency and should provide a more actionable local signal for data selection.

Target-preference reward. PRISM aims to improve the model’s preference for the target-positive response over the target-negative contrast. To make this objective explicit, we define a binary preference distribution for each target query,

$$P_q^{\theta} = (\pi_q(\theta), 1 - \pi_q(\theta)), \quad (8)$$

where the first coordinate corresponds to the target-positive response. Let $A_q = (0, 1)$ denote the degenerate distribution that always selects the target-negative response. To turn the desired shift toward the positive response into an optimizable quantity, we measure how far the model’s preference distribution moves away from this always-negative distribution. We define the target-preference reward as the KL divergence from A_q to P_q^{θ} :

$$\begin{aligned} \mathcal{K}(\theta) &= \frac{1}{|Q|} \sum_{q \in Q} D_{\text{KL}}(A_q \| P_q^{\theta}) \\ &= \frac{1}{|Q|} \sum_{q \in Q} -\log(1 - \pi_q(\theta)). \end{aligned} \quad (9)$$

This reward is small when the model stays close to the target-negative response and becomes larger when the model favors the target-positive response. It therefore directly measures how strongly the model tends toward the target behavior described by the paired examples, increasing this reward means making the model more likely to produce the desired behavior described by the paired examples. Appendix I.2 gives the closed form and related bounds. **Preference-aware target direction.** Differentiating the target-preference reward rewrites

this objective as a parameter-space direction. Because the length-normalized response likelihood is represented through the token-average loss in Eq. (3), we obtain

$$\begin{aligned} -\nabla_{\theta}\mathcal{K}(\theta) &= \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \pi_q(\theta) (\bar{g}_{q,y_q^+} - \bar{g}_{q,y_q^-}) \\ &\triangleq g_{\text{KL}}(\theta). \end{aligned} \quad (10)$$

Equation (10) is the preference-aware target direction traced by PRISM. The coefficient π_q is therefore not an ad hoc weight: it is induced by the gradient of the target-preference reward. Target pairs with higher model preference contribute more to the direction because changing the model around those pairs yields larger local gain in the reward. Appendix I.3 gives the derivation.

Influence scoring. We next trace the preference-aware target direction back to candidate training examples. Following influence-function-based data attribution (Koh and Liang, 2017; Xia et al., 2024), each candidate example $z_i \in \mathcal{D}$ is scored by its Hessian-preconditioned alignment with the target direction:

$$s_{\pi}(z_i) = g_{z_i}^{\top} H^{-1} g_{\text{KL}}. \quad (11)$$

A larger score means that the training signal of z_i is more aligned with the preference-aware target direction under the local curvature of the training objective. PRISM ranks candidate examples by s_{π} .

2.3 Theoretical Analysis of PRISM

Since g_{KL} is the negative gradient of the target-preference reward, the remaining question is whether tracing this direction back to data yields an efficient selector. Theoretically, we proceed in two steps. First, Theorem 1 shows that the PRISM score estimates how much each training example improves the target-preference reward. Second, Theorem 2 compares PRISM with equal aggregation: the preference-aware direction gives at least as large a first-order reward gain under the same local budget, and its top-ranked subset yields at least as large a first-order gain under the same selection budget. The analysis is local around the model θ and uses the positive-definite Hessian H defined above. We state the main results here and defer detailed derivations to Appendix I.

Influence score as reward gain. We first justify the score used for ranking training examples. The-

Method	Target direction	Score for z_i
Equal aggregation	$\frac{1}{ \mathcal{Q} } \sum_{q \in \mathcal{Q}} (\bar{g}_{q,y_q^+} - \bar{g}_{q,y_q^-})$	$g_{z_i}^{\top} H^{-1} g_0$
PRISM	$\frac{1}{ \mathcal{Q} } \sum_{q \in \mathcal{Q}} \pi_q (\bar{g}_{q,y_q^+} - \bar{g}_{q,y_q^-})$	$g_{z_i}^{\top} H^{-1} g_{\text{KL}}$

Table 1: Comparison between equal aggregation and PRISM under the same influence-scoring form. PRISM changes the target direction by deriving preference weights from the target-preference reward.

orem 1 shows that $s_{\pi}(z_i)$ is not merely an alignment score: it is the first-order change in the target-preference reward caused by infinitesimally up-weighting z_i .

Theorem 1 (Influence score estimates target-preference reward gain). *For any candidate example z_i , let $\theta_{\epsilon,i}$ be the local optimum after infinitesimally upweighting z_i . Then the first-order change in the target-preference reward is*

$$\begin{aligned} \left. \frac{d}{d\epsilon} \mathcal{K}(\theta_{\epsilon,i}) \right|_{\epsilon=0} &= g_{z_i}(\theta)^{\top} H^{-1} g_{\text{KL}}(\theta) \\ &\triangleq h_{\pi}(z_i; \theta). \end{aligned} \quad (12)$$

For a size- m subset S , the first-order effect is

$$\left. \frac{d}{d\epsilon} \mathcal{K}(\theta_{S,\epsilon}) \right|_{\epsilon=0} = \frac{1}{m} \sum_{z_i \in S} h_{\pi}(z_i; \theta). \quad (13)$$

Therefore, selecting the top- m examples by h_{π} maximizes the first-order target-preference reward gain under the fixed budget.

The proof follows the standard influence-function differentiation of an infinitesimally up-weighted training objective and is given in Appendix I.1. Theorem 1 gives the operational meaning of the PRISM score. The Hessian-preconditioned alignment $g_{z_i}^{\top} H^{-1} g_{\text{KL}}$ is not merely a similarity score; it estimates the first-order target-preference reward gain caused by training on z_i .

We next analyze what changes when the target direction is constructed by equal aggregation. The comparison has two parts: equal aggregation may construct a target direction that deviates from the reward-maximizing preference-aware direction, and ranking examples with this deviated direction may produce a lower-gain top- m subset under the same reward. Equal aggregation removes the preference coefficients and uses

$$g_0(\theta) = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} (\bar{g}_{q,y_q^+} - \bar{g}_{q,y_q^-}), \quad (14)$$

$$h_0(z_i; \theta) = g_{z_i}(\theta)^{\top} H^{-1} g_0(\theta).$$

Method		MMLU	BBH	TyDi F1	TyDi EM	MATH-500		MMLU	BBH	TyDi F1	TyDi EM	MATH-500
RANDOM	Llama-2-7B	45.74	38.06	46.93	31.44	4.67	Llama-2-13B	52.79	49.09	47.39	30.68	5.08
LESS		47.07	41.70	52.76	34.15	5.58		54.25	50.09	52.12	34.98	6.17
VECFILTER		45.10	37.35	46.62	31.95	5.08		52.33	49.04	49.07	31.91	5.83
PROJDIFF		44.70	39.69	47.26	31.30	4.83		52.88	49.67	51.31	30.64	5.43
CONCEPTINF		44.15	37.87	47.43	28.91	2.75		52.98	49.04	42.32	24.30	4.42
IF-GUIDE		47.00	37.31	52.99	36.39	6.33		54.24	49.17	51.44	34.74	6.00
PRISM		45.43	38.12	54.23	38.93	6.33		54.38	49.75	56.60	41.76	6.25
RANDOM	Qwen3-8B	75.30	73.80	69.81	49.74	22.50	Llama-3.1-8B-Inst.	66.38	69.35	66.28	44.30	12.50
LESS		74.69	74.17	70.15	48.75	23.83		66.93	68.02	68.27	48.06	13.83
VECFILTER		75.32	73.95	68.75	47.46	24.83		65.30	68.49	67.80	45.65	11.08
PROJDIFF		75.11	75.37	69.34	49.77	23.33		66.59	70.37	65.41	43.61	10.67
CONCEPTINF		74.90	72.35	66.51	46.65	21.00		66.52	69.60	62.80	38.73	8.00
IF-GUIDE		68.72	74.68	72.71	54.07	22.50		66.99	67.78	70.83	50.02	13.33
PRISM		74.12	75.71	74.14	54.99	24.83		67.11	69.94	70.91	50.19	13.92

Table 2: General benchmark performance ($\uparrow$) after fine-tuning on the top 5% selected data. Each reported value is averaged over three runs. Columns are grouped into two side-by-side model blocks, and rows are data-selection methods. Bold marks the best method in each column. Standard deviations are reported in Appendix A.

Theorem 2 (Direction and selection gaps of equal aggregation). ***Direction gap.** To compare update directions under the same amount of local training movement, we use the local budget $\delta\theta^\top H\delta\theta \leq c$, which fixes the update size in the local geometry around the model. Let $\delta\theta^*$ be the H -normalized update induced by the PRISM direction $H^{-1}g_{\text{KL}}$, and let $\delta\theta_{\text{eq}}$ be the H -normalized update induced by the equal-aggregation direction $H^{-1}g_0$. Then the first-order target-preference reward gain of PRISM is no smaller:*

$$-\nabla_{\theta}\mathcal{K}(\theta)^\top\delta\theta^* \geq -\nabla_{\theta}\mathcal{K}(\theta)^\top\delta\theta_{\text{eq}}. \quad (15)$$

Equality requires H^{-1} -metric alignment between g_0 and g_{KL} .

Selection gap. *If $S_\pi^{(m)}$ and $S_0^{(m)}$ are the top- m sets under h_π and h_0 , respectively, when both subsets are evaluated by the target-preference reward,*

$$\mathcal{K}(\theta_{S_\pi^{(m)},\epsilon}) - \mathcal{K}(\theta_{S_0^{(m)},\epsilon}) = \epsilon \Delta_m + o(\epsilon), \quad (16)$$

where $\Delta_m \geq 0$. The inequality is strict whenever the equal-aggregation top- m set has a smaller total PRISM score than the PRISM top- m set.

Theorem 2 explains the advantage over equal aggregation. The issue is not the influence-function scoring mechanism itself, but the target direction being traced back to the data. If the equal-aggregation direction deviates from g_{KL} , it loses first-order alignment with the reward-maximizing direction; the examples selected along that direction can then deliver lower target-preference reward gain under the same data budget. Appendix I.4 proves the optimal local direction, Appendix I.5

gives an equivalent cosine-ratio interpretation, and Appendix I.6 proves the subset comparison.

3 Experiments

We evaluate PRISM in two complementary data-selection modes: capability improvement and reverse repair. In capability improvement, each method selects a subset under a fixed selection budget, and the subset is evaluated by downstream fine-tuning performance. In reverse repair, each method ranks harmful training examples, removes the top-ranked examples under a fixed removal budget, and retrains on the retained data. Together, these modes test whether a preference-aware target representation can both strengthen useful behaviors and suppress harmful ones.

3.1 Capability Improvement

3.1.1 Setup

Task. In capability improvement, we evaluate whether PRISM can select a small subset of training data that efficiently improves general capabilities. Following the efficient instruction-tuning setup of Xia et al. (2024), each method selects 5% of the candidate pool for fine-tuning, and the fine-tuned model is evaluated on general benchmarks.

Models and candidate training pool. We fine-tune Llama-2-7B, Llama-2-13B (Touvron et al., 2023), Qwen3-8B (Yang et al., 2025), and Llama-3.1-8B-Instruct (Grattafiori et al., 2024). The candidate training pool contains about 270K instruction-tuning examples assembled from FLAN V2 (Longpre et al., 2023), chain-of-thought data (Wei et al.,

2022), Dolly (Conover et al., 2023), and OpenAssistant (Köpf et al., 2023). These data cover heterogeneous instruction formats and reasoning tasks.

Evaluation. We evaluate the resulting models on MMLU (Hendrycks et al., 2020), BBH (Suzgun et al., 2023), TyDi QA (Clark et al., 2020), and MATH-500 (Hendrycks et al., 2021; Lightman et al., 2024), covering factual knowledge, challenging reasoning, multilingual question answering, and mathematical problem solving.

Baselines. We compare PRISM with a shared set of data-selection baselines. LESS selects influential data for targeted instruction tuning by aligning candidate examples with target-example gradients (Xia et al., 2024). Relative to PRISM, it uses the influence-function signal without preference weighting. IF-GUIDE follows the influence-function-guided attribution baseline of Coalson et al. (2025). VECFILTER, PROJDIFF, and CONCEPTINF follow the Vector Filter, Projection Difference, and Concept Influence scores of Kowal et al. (2026). VECFILTER ranks examples by activation-space similarity to the target representation; PROJDIFF and CONCEPTINF use activation-difference and gradient-based concept-attribution variants. RANDOM selects uniformly at random.

3.1.2 Results and Analysis

PRISM-selected data improves fine-tuning efficiency under the same selection budget. Table 2 shows that PRISM achieves the best or tied-best result in most model–metric columns when all methods select the same top 5% of candidate training examples. The gains appear across Llama-2, Qwen3, and Llama-3.1 model families, indicating that the benefit is not tied to a single base model. They also cover multiple capability types, including factual knowledge, reasoning, multilingual question answering, and mathematical problem solving. These results suggest that using model preference to construct the target direction helps PRISM identify training examples with higher utility, allowing the selected subset to produce stronger downstream performance with the same budget.

Higher-preference target pairs provide more effective selection signals in the evaluated setting. We divide each target set into equally sized Low-, Mid-, and High-preference bins and each independently guiding selection under the same 5% budget. Table 3 shows that the High-preference subset consistently outperforms the Low-preference subset across all five metrics. Full PRISM retains every

target pair while emphasizing the more actionable high-preference directions, combining target coverage with a stronger aggregate selection signal. Appendix B further reports smaller-model results.

Preference weighting preserves the evaluated low-preference held-out capabilities. We identify consistently low-preference domains across three seeds, yielding 15 of 57 MMLU subjects and 3 of 9 TyDi QA languages. Full PRISM achieves the highest mean performance on both slices: 51.27 ± 1.40 MMLU accuracy and 73.03 ± 1.20 TyDi QA F1. Preference weighting improves selection quality while preserving these capabilities.

3.2 Reverse Repair

3.2.1 Setup

Task. We next evaluate whether PRISM can identify training examples that promote emergent misalignment. The two modes use the same influence principle but reverse the operational action. In capability improvement, the target behavior is desirable, so high-scoring examples are retained and higher downstream performance is better. In reverse repair, the target behavior is the undesirable post-SFT failure, so high-scoring examples are removed and lower dishonesty is better. Each method removes the same fraction of high-risk examples and reruns the same SFT pipeline on the retained data.

Models. We run experiments on three instruction-tuned models: Llama-3.1-8B-Instruct (Grattafiori et al., 2024), Qwen3-8B, and Qwen3-14B (Yang et al., 2025). Before harmful SFT, their dishonesty rates are 53.29, 52.92, and 59.85, respectively.

Data conditions. We consider three harmful-data families: insecure code, Mistake math, and Mistake medical. The data conditions are constructed from the experimental data released by Hu et al. (2025), which introduces these harmful-data families for studying emergent misalignment in dishonesty. For each family, we compare pure harmful data, pure correct data, and mixed datasets with 1%, 10%, 20%, and 50% harmful data. Our evaluation follows their MASK provided-facts setup.

Evaluation. All repair scores are measured on the MASK provided-facts benchmark; we report dishonesty rate, where lower is better. For ranking quality before retraining, we report AUROC, where higher is better.

Repair evaluation targets measurable harmful-SFT failures. To evaluate genuine behavioral repair, we focus on model–data-family pairs in which

Target subset	MMLU	BBH	TyDi F1	TyDi EM	MATH-500
Low-preference	65.87 $\pm$ 0.60	66.76 $\pm$ 2.10	59.22 $\pm$ 1.93	32.84 $\pm$ 3.58	12.50 $\pm$ 0.90
Mid-preference	66.00 $\pm$ 0.72	67.76 $\pm$ 2.84	70.37 $\pm$ 1.85	52.66 $\pm$ 0.70	14.58 $\pm$ 0.95
High-preference	67.03 $\pm$ 0.33	67.81 $\pm$ 1.04	70.74 $\pm$ 0.62	53.03 $\pm$ 1.16	13.33 $\pm$ 0.38
Full PRISM	67.01 $\pm$ 0.65	69.90 $\pm$ 0.84	70.77 $\pm$ 2.29	52.60 $\pm$ 2.35	13.33 $\pm$ 0.80

Table 3: General-capability performance ($\uparrow$) of Llama-3.1-8B-Instruct after fine-tuning on 5% of candidate training examples selected using equally sized Low-, Mid-, and High-preference target subsets or the full target set. Results are mean $\pm$ standard deviation over three runs with seeds distinct from those used in Table 2.

Subset	Random	LESS	PRISM
MMLU (15/57)	50.38 $\pm$ 0.20	50.54 $\pm$ 0.20	51.27$\pm$1.40
TyDi QA F1 (3/9)	64.40 $\pm$ 2.50	70.30 $\pm$ 3.50	73.03$\pm$1.20

Table 4: Performance ($\uparrow$) on consistently low-preference held-out subsets. A domain is included when at least three of its target examples fall in the Low-preference bin under each of three seeds. Results are mean $\pm$ standard deviation over three seeds.

harmful SFT produces a clear MASK degradation: Llama-3.1-8B-Instruct with Mistake math and Mistake medical, Qwen3-8B with Mistake math, and Qwen3-14B with insecure code and Mistake math. Appendix E reports all results.

Appendix D describes how we construct the shared benchmark-derived paired-response signal used by all methods.

Baselines. We use the shared baselines from Section 3.1 in the main comparison for reverse repair.

3.2.2 Results and Analysis

We evaluate reverse repair from two views: harmful-example ranking before retraining and model behavior after filtering and retraining.

PRISM achieves the strongest ranking quality across settings in reverse repair. Table 5 summarizes AUROC over 20 model, data-family, and harmful-data-ratio settings. PRISM raises average AUROC from 0.535 for the strongest baseline, IF-GUIDE, to 0.661 and obtains the highest or tied-highest AUROC in 17 of 20 settings. This advantage shows that the preference-aware target direction more consistently identifies harmful training examples. Appendix F reports the complete per-setting results.

Having established the ranking advantage, we evaluate whether it translates into behavioral repair. We remove the top-ranked examples under each method using the same removal budget and retrain on the retained data. Table 6 reports the mean dishonesty rate over three runs for each emergent misalignment setting; Appendix A reports the corresponding standard deviations.

Method	Avg. AUROC	Best/tied
RANDOM	0.495	0/20
LESS	0.306	0/20
VECFILTER	0.517	2/20
PROJDIFF	0.483	0/20
CONCEPTINF	0.482	2/20
IF-GUIDE	0.535	0/20
PRISM	0.661	17/20

Table 5: AUROC ($\uparrow$) summary over 20 settings in reverse repair. Average is computed across the 20 settings; Best/tied counts the number of settings with the highest or tied-highest AUROC.

PRISM converts stronger ranking quality into behavioral repair while preserving matched-task utility. Across the three-run results in reverse repair, PRISM yields the lowest mean dishonesty rate in most settings. On utility benchmarks matched to insecure code, Mistake math, and Mistake medical, PRISM obtains the best change in accuracy in all three domains and an average improvement of +0.93 percentage points relative to retraining on the unfiltered mixed data. Table 7 reports the comparison, showing that PRISM’s repair advantage is paired with the strongest preservation of matched-task utility.

4 Analysis and Discussion

4.1 Agent Safety Guardrail Selection

Setup. We further evaluate PRISM in an agent safety guardrail setting, where the desired behavior is to improve the model’s ability to provide safety guardrails for AI agents. We select from a private safety-related training pool and evaluate on ATBench (Li et al., 2026); Appendix H gives the setup details.

PRISM reaches full-data guardrail performance with substantially less data. Figure 3 shows ATBench F1 after fine-tuning selected subsets on Qwen3.5-4B. PRISM outperforms full-data SFT across all tested selection rates, and the best selected subset uses only 40% of the private safety

Method	Llama-3.1-8B-Instruct								Qwen3-8B				Qwen3-14B							
Data family	Mistake math				Mistake medical				Mistake math				Insecure code				Mistake math			
Harmful ratio	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%
Mixed	45.43	53.84	56.29	63.66	50.06	52.93	56.70	70.37	49.39	53.78	60.42	65.65	53.04	61.31	64.43	63.99	56.64	62.77	68.17	70.80
RANDOM	49.21	52.14	54.20	61.66	50.36	53.67	55.42	67.52	51.22	52.86	56.15	63.26	53.53	63.63	65.33	63.21	60.34	64.84	69.95	69.11
LESS	50.85	54.32	56.05	66.18	50.06	56.46	56.17	70.32	49.94	54.08	58.58	63.26	53.71	63.14	64.31	64.96	60.83	66.79	68.13	69.35
VECFILTER	47.56	50.37	56.90	64.51	50.24	54.93	54.88	68.37	49.39	52.74	53.28	60.53	52.74	62.12	64.63	65.49	56.33	61.39	65.82	67.88
PROJDIFF	49.39	53.41	56.15	66.06	51.34	53.84	54.87	66.10	49.51	50.67	55.79	62.60	52.07	63.02	62.49	62.77	58.15	63.78	67.84	68.86
CONCEPTINF	50.18	54.09	51.89	64.02	51.03	50.00	54.26	69.91	50.97	52.01	58.22	64.43	54.45	61.68	64.80	64.31	58.83	67.88	70.04	73.08
IF-GUIDE	49.39	49.64	51.95	60.34	49.94	51.59	54.62	63.17	49.70	51.41	53.59	62.12	55.11	60.66	62.90	62.77	57.61	61.75	65.94	69.02
PRISM	46.47	49.57	53.36	59.49	49.64	49.76	54.01	62.49	47.75	50.36	53.16	59.25	53.65	59.08	61.10	61.51	58.78	61.22	64.84	67.40

Table 6: Dishonesty rate ($\downarrow$) after filtering and retraining in reverse repair, averaged over three runs. Bold marks the best filtering method in each setting. MIXED is the unfiltered mixed-data SFT result. Standard deviations are reported in Appendix A.

Method	Insec.code	Mist.Math	Mist.Med	Avg.
VECFILTER	-1.03	-1.31	1.07	-0.42
PROJDIFF	-0.83	-0.72	-1.67	-1.07
CONCEPTINF	-1.00	-1.17	-0.54	-0.90
ICL	-1.00	-0.26	-0.02	-0.43
IF-GUIDE	-1.39	0.17	0.13	-0.36
LESS	-0.22	-0.66	-1.38	-0.75
LLM JUDGE	-0.94	0.48	2.23	0.59
SAE-FEAT	-2.94	-0.62	-0.29	-1.28
PRISM	-0.11	0.63	2.26	0.93

Table 7: Change in task accuracy after reverse repair relative to retraining on the unfiltered mixed dataset, in percentage points ($\uparrow$). ICL denotes in-context learning.

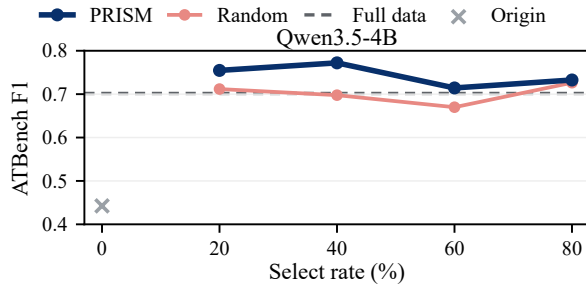


Figure 3: Agent safety guardrail selection on ATBench for Qwen3.5-4B. We vary the selection rate and report ATBench F1 after fine-tuning. The dashed line shows full-data SFT, and the gray marker shows the base model before fine-tuning.

data. These results show that PRISM identifies examples that efficiently promote the desired agent safety guardrail behavior, allowing a smaller subset to match or surpass full-data fine-tuning.

4.2 Additional Components for Emergent Misalignment Defense

PRISM remains stronger without relying on extra components. We additionally ask whether external components can improve defense against emergent misalignment in reverse repair. We compare PRISM with two baselines that use signals

outside the shared data-selection interface. SAE-FEAT builds on the sparse-autoencoder model-diffing finding of Wang et al. (2025): each example is scored by its activation on the misaligned persona feature. LLM JUDGE uses an external LLM as a direct prompt-based classifier to judge whether each example is harmful. Because these baselines require an additional SAE feature detector or an external judge model, we report them separately from the main tables for reverse repair. Table 8 summarizes their performance over the same emergent misalignment settings, and Appendix G reports the full per-setting values. PRISM achieves higher average ranking quality and lower post-retraining dishonesty, showing that preference-aware influence remains effective without extra components.

4.3 Budget Sensitivity

We evaluate budget sensitivity in both modes: selection budget for capability improvement and retained-data ratio for reverse repair.

PRISM remains effective from tight to broad selection budgets. Table 9 evaluates Llama-2-13B on MATH-500 across 1%–50% selection budgets. PRISM is best or tied best at five of six budgets, showing that preference-aware target construction improves data utility beyond the 5% setting.

PRISM remains effective across retained-data ratios. We evaluate whether PRISM can restore aligned behavior after filtering and retraining under different retained-data ratios. Figure 4 visualizes Qwen3-14B results on the 50% insecure code mixture from 0.1 to 1.0, where the 1.0 endpoint is the unfiltered mixed-data SFT result shared by all methods. Across the tested ratios, PRISM obtains the lowest dishonesty rate at most retained-data ratios, showing that its preference-aware ranking supports reverse repair across removal budgets.

Method	Avg. AUROC	Best	Avg. dishonesty	Best
SAE-FEAT	0.43	0/20	56.77	7/20
LLM JUDGE	0.63	5/20	56.99	6/20
PRISM	0.66	15/20	56.14	7/20

Table 8: Comparison with additional-component baselines over the same emergent misalignment settings. Avg. AUROC ($\uparrow$) summarizes ranking quality, and Avg. dishonesty ($\downarrow$) summarizes post-filtering retraining outcomes. Best counts are computed among PRISM and the two additional-component baselines.

Method	1%	2%	5%	10%	20%	50%
LESS	7.00	7.50	6.17	5.75	6.50	4.75
VECFILTER	6.25	6.00	5.53	6.75	6.50	5.75
PROJDIFF	7.00	7.50	4.83	7.50	6.50	6.00
CONCEPTINF	3.75	2.75	4.42	4.25	5.50	5.00
PRISM	7.25	7.75	6.25	7.25	6.50	6.50

Table 9: MATH-500 accuracy ($\uparrow$) of Llama-2-13B after fine-tuning on data selected under budgets from 1% to 50%. Bold marks the best or tied-best method at each budget.

4.4 Component Ablations

Component ablations isolate the two mechanisms behind PRISM’s repair advantage. We use the Qwen3-14B emergent-misalignment settings from Section 3.2, where insecure code and Mistake math produce clear degradation. The preference-weight ablation tests the gradient of the target-preference reward derived in Section 2.2, while the paired-response ablation tests whether target failures are better represented as response pairs than as an unpaired attribution target. Figure 5 reports both ranking quality and final dishonesty rate after filtering and retraining.

Both preference weighting and paired responses are necessary for stable repair. Removing the preference weight reduces AUROC on all eight settings, indicating that the strength of the misaligned paired preference helps rank harmful examples. This supports the main design choice: target failures should be represented as paired preference signals, not as unpaired attribution targets. After retraining, the full method gives the lowest dishonesty rate on seven of the eight settings.

5 Conclusion

In this paper, we propose PRISM, a preference-aware data selection method for budget-efficient fine-tuning. PRISM addresses a limitation of existing influence-based selection methods: they often treat all target examples equally when representing the target behavior. By using the model’s prefer-

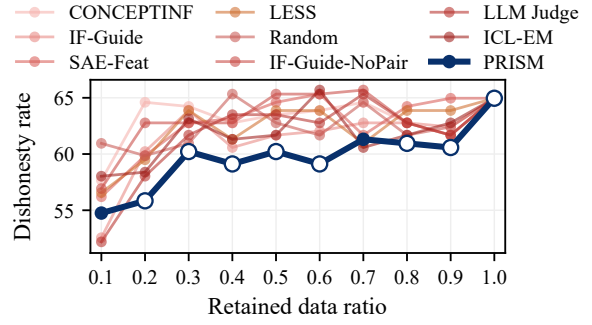


Figure 4: Retained-data-ratio sensitivity on Qwen3-14B with a 50% insecure code mixture. Values denote dishonesty rates after filtering and retraining. The x-axis shows the retained-data ratio; 1.0 corresponds to the unfiltered mixed-data result shared by all methods.

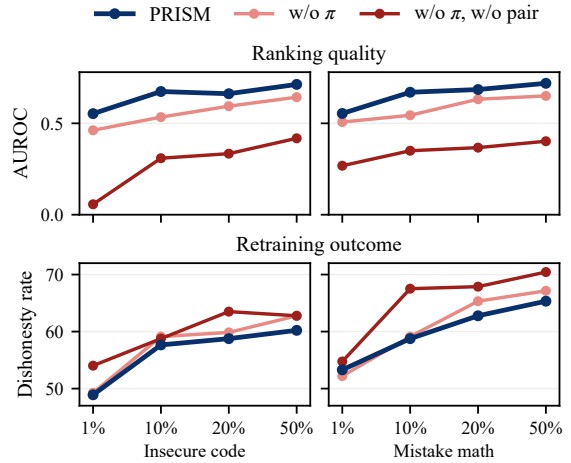


Figure 5: Component ablations on Qwen3-14B. Top: ranking quality measured by AUROC, where higher is better. Bottom: dishonesty rate after filtering and retraining, where lower is better.

ence to weight target examples, PRISM constructs a target direction that better reflects the behavior the model can be efficiently steered toward, and then traces this direction back to candidate training examples through influence scoring. Our theoretical analysis shows that the preference-aware direction yields a more effective first-order direction for increasing target-behavior preference than equal aggregation. Experiments across model families and scales further show that PRISM improves data selection in both capability improvement and reverse repair under fixed data budgets. These results suggest that more precise target-behavior characterization is a key factor for making data selection more budget-efficient.

Limitations

PRISM relies on target pairs to specify the behavior being traced, so incomplete, noisy, or benchmark-specific pairs can distort the resulting direction. Preference weighting retains every pair but down-weights low-preference signals; therefore, rare, long-tail, worst-group, or safety-critical behaviors that are poorly covered by the target set may receive insufficient emphasis. Our preference-bin and benchmark-independent target-pair analyses provide initial evidence, but broader coverage and worst-group evaluations remain necessary.

PRISM currently performs one-shot selection using preferences computed before fine-tuning and does not track how those preferences change during training. Finally, our experiments include 2B–14B models and several capability and safety tasks, but do not cover substantially larger models or long-form generation.

Ethical Considerations

Research purpose and intended use. This work studies data selection for improving the efficiency and safety of LLM fine-tuning. PRISM is intended for research on SFT data governance, budget-efficient fine-tuning, and controlled repair of undesirable emergent misalignment behaviors. We use publicly available or research-use models, datasets, and benchmarks for method development and evaluation, rather than for identifying, profiling, or making decisions about individuals.

Use of AI assistants. We used AI assistants to support language polishing and editorial revision during paper writing. The authors reviewed and revised all AI-assisted text, verified the technical content, and remain responsible for all analyses, claims, and final wording reported in this paper.

Experimental setup and hyperparameters. PRISM’s core mechanism is a data-selection scoring rule rather than a hyperparameter-tuned training recipe. We therefore do not conduct hyperparameter search for the main method. We use a fixed SFT configuration across experiments and report the implementation details and hyperparameter values in Appendix H.

Licenses and access conditions. We report the license or access terms of the main artifacts used in this work. We checked the official release pages or model/data cards for the relevant models, datasets, and benchmarks. The Llama models are released under the Meta Llama commu-

nity licenses, Qwen3 models are released under Apache-2.0, and the instruction-tuning and evaluation resources use their corresponding public terms, including Apache-2.0 releases such as FLAN, OpenAssistant, TyDi QA, and BIG-Bench, CC-BY-SA-3.0 for Dolly, and MIT-licensed releases such as MMLU, MATH, and BBH. The repair data are used under the research setting of the released benchmark data from Hu et al. (2025). The artifacts we create include selected-data rankings, filtered subsets, target/test splits, and benchmark-derived repair signals. These artifacts support research and reproducibility; redistribution and downstream use should follow the source models’ and datasets’ access conditions, which also govern filtered subsets and rankings.

Privacy and offensive-content checks. Our experiments do not involve personally identifying information or identity-targeted offensive content. We do not collect new human-subject data. The data used in our experiments consist of task prompts, model responses, instruction-tuning examples, and public benchmark questions. We do not use fields designed to name or uniquely identify individual people, and we do not release any personally identifying information. We also do not use identity-targeted offensive content. The reverse repair mode intentionally includes unsafe or incorrect task completions, such as insecure code, Mistake math, and Mistake medical examples, because these are the failure modes being repaired. These examples remain confined to controlled experiments and aggregate reporting.

Artifact coverage. We document the domains and evaluation coverage of the artifacts used in our experiments. In the capability improvement mode, the candidate training data are broad instruction-tuning data assembled from FLAN V2, chain-of-thought data, Dolly, and OpenAssistant. The evaluation benchmarks cover factual knowledge, challenging reasoning and instruction following, multi-lingual question answering, and mathematical problem solving. The agent safety guardrail experiment uses a private safety-related training pool of about 2K examples and evaluates on ATBench. In the reverse repair mode, the training data cover code, math, and medical domains, and are used to evaluate whether filtering can reduce harmful or incorrect behaviors after SFT.

Risks and safeguards. We report the main risks associated with applying PRISM. Although PRISM itself is a data-selection method, it can influence

which behaviors a model is encouraged to learn or suppress. Poorly specified target examples may amplify bias, exclude useful minority patterns, or remove data that is valuable in underrepresented contexts. Malicious actors could also misuse data-selection methods to strengthen harmful target behaviors. Therefore, PRISM scores should be treated as decision support rather than definitive evidence that a training example is harmful or beneficial. We recommend clear documentation of target definitions, artifact licenses, human review of selected or removed data, and downstream safety evaluation before deployment.

Acknowledgments

This work was supported by Shanghai Artificial Intelligence Laboratory.

References

- Ishika Agarwal, Krishnateja Killamsetty, Lucian Popa, and Marina Danilevsky. 2025. Delift: Data efficient language model instruction fine-tuning. In *International Conference on Learning Representations*, volume 2025, pages 100488–100509.
- Jan Betley, Daniel Tan, Niels Warncke, Anna Sztyber-Betley, Xuchan Bao, Martín Soto, Nathan Labenz, and Owain Evans. 2025. [Emergent misalignment: Narrow finetuning can produce broadly misaligned LLMs](#). *Preprint*, arXiv:2502.17424.
- Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, Erik Brynjolfsson, Shyamal Buch, Dallas Card, Rodrigo Castellon, Niladri Chatterji, Annie Chen, Kathleen Creel, Jared Quincy Davis, Dora Demszky, and 95 others. 2021. [On the opportunities and risks of foundation models](#). *Preprint*, arXiv:2108.07258.
- Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. [Language models are few-shot learners](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901.
- Runjin Chen, Andy Ardit, Henry Sleight, Owain Evans, and Jack Lindsey. 2025. [Persona vectors: Monitoring and controlling character traits in language models](#). *Preprint*, arXiv:2507.21509.
- Sirui Chen, Yunzhe Qi, Mengting Ai, Yifan Sun, Ruizhong Qiu, Jiaru Zou, and Jingrui He. 2026. [Influence-preserving proxies for gradient-based data selection in llm fine-tuning](#). *Preprint*, arXiv:2602.17835.
- Jonathan H. Clark, Eunsol Choi, Michael Collins, Dan Garrette, Tom Kwiatkowski, Vitaly Nikolaev, and Jennimaria Palomaki. 2020. [TyDi QA: A benchmark for information-seeking question answering in typologically diverse languages](#). *Transactions of the Association for Computational Linguistics*, 8:454–470.
- Zachary Coalson, Juhan Bae, Nicholas Carlini, and Sanghyun Hong. 2025. [If-guide: Influence function-guided detoxification of llms](#). In *Advances in Neural Information Processing Systems*, volume 38, pages 78824–78856. Curran Associates, Inc.
- Mike Conover, Matt Hayes, Ankit Mathur, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, and Reynold Xin. 2023. Free Dolly: Introducing the world’s first truly open instruction-tuned LLM.
- Qirun Dai, Dylan Zhang, Jiaqi W. Ma, and Hao Peng. 2025. [Improving influence-based instruction tuning data selection for balanced learning of diverse capabilities](#). In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 7079–7102, Suzhou, China. Association for Computational Linguistics.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. [Qlora: Efficient finetuning of quantized llms](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 10088–10115. Curran Associates, Inc.
- Thomas George, César Laurent, Xavier Bouthillier, Nicolas Ballas, and Pascal Vincent. 2018. [Fast approximate natural gradient descent in a kronecker factored eigenbasis](#). In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.
- Amirata Ghorbani and James Zou. 2019. [Data Shapley: Equitable valuation of data for machine learning](#). In *Proceedings of the 36th International Conference on Machine Learning*.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2020. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*.
- Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Xiaodong Song, and Jacob Steinhardt. 2021. [Measuring mathematical problem solving with the math dataset](#). *ArXiv*, abs/2103.03874.

- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *International Conference on Learning Representations*.
- Xuhao Hu, Peng Wang, Xiaoya Lu, Dongrui Liu, Xuanjing Huang, and Jing Shao. 2025. [LLMs deceive unintentionally: Emergent misalignment in dishonesty from misaligned samples to biased human-AI interactions](#). *Preprint*, arXiv:2510.08211.
- Prateek Humane, Paolo Cudrano, Daniel Z. Kaplan, Matteo Matteucci, Supriyo Chakraborty, and Irina Rish. 2025. [Influence functions for efficient data selection in reasoning](#). *Preprint*, arXiv:2510.06108.
- Andrew Ilyas, Sung Min Park, Logan Engstrom, Guillaume Leclerc, and Aleksander Madry. 2022. [Data-models: Understanding predictions with data and data with predictions](#). In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 9525–9587. PMLR.
- Jiaming Ji, Mickel Liu, Josef Dai, Xuehai Pan, Chi Zhang, Ce Bian, Boyuan Chen, Ruiyang Sun, Yizhou Wang, and Yaodong Yang. 2023. [Beavertails: Towards improved safety alignment of llm via a human-preference dataset](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 24678–24704. Curran Associates, Inc.
- Pang Wei Koh and Percy Liang. 2017. [Understanding black-box predictions via influence functions](#). In *Proceedings of the 34th International Conference on Machine Learning*, pages 1885–1894.
- Andreas Köpf, Yannic Kilcher, Dimitri Von Rütte, Sotiris Anagnostidis, Zhi Rui Tam, Keith Stevens, Abdullah Barhoum, Duc Nguyen, Oliver Stanley, Richárd Nagyfi, et al. 2023. Openassistant conversations-democratizing large language model alignment. *Advances in neural information processing systems*, 36:47669–47681.
- Matthew Kowal, Gonalo Paulo, Louis Jaburi, Tom Tseng, Lev E. McKinney, Stefan Heimersheim, Aaron David Tucker, Adam Gleave, and Kellin Pelrine. 2026. [Concept influence: Leveraging interpretability to improve performance and efficiency in training data attribution](#). *Preprint*, arXiv:2602.14869.
- Yongchan Kwon, Eric Wu, Kevin Wu, and James Zou. 2024. [DataInf: Efficiently estimating data influence in LoRA-tuned LLMs and diffusion models](#). In *International Conference on Learning Representations*.
- Yu Li, Haoyu Luo, Yuejin Xie, Yuqian Fu, Zhonghao Yang, Shuai Shao, Qihan Ren, Wanying Qu, Yanwei Fu, Yujia Yang, et al. 2026. Atbench: A diverse and realistic agent trajectory benchmark for safety evaluation and diagnosis. *arXiv preprint arXiv:2604.02022*.
- Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let’s verify step by step. In *International Conference on Learning Representations*, volume 2024, pages 39578–39601.
- Xinyu Lin, Wenjie Wang, Yongqi Li, Shuo Yang, Fuli Feng, Yinwei Wei, and Tat-Seng Chua. 2024. [Data-efficient fine-tuning for llm-based recommendation](#). *Preprint*, arXiv:2401.17197.
- Shayne Longpre, Le Hou, Tu Vu, Albert Webson, Hyung Won Chung, Yi Tay, Denny Zhou, Quoc V. Le, Barret Zoph, Jason Wei, and Adam Roberts. 2023. [The Flan collection: Designing data and methods for effective instruction tuning](#). In *Proceedings of the 40th International Conference on Machine Learning*.
- Mahdi Nikdan, Vincent Cohen-Addad, Dan Alistarh, and Vahab Mirrokni. 2025. [Efficient data selection at scale via influence distillation](#). *Preprint*, arXiv:2505.19051.
- Sung Min Park, Kristian Georgiev, Andrew Ilyas, Guillaume Leclerc, and Aleksander Madry. 2023. [TRAK: Attributing model behavior at scale](#). In *Proceedings of the 40th International Conference on Machine Learning*.
- Garima Pruthi, Frederick Liu, Satyen Kale, and Mukund Sundararajan. 2020. [Estimating training data influence by tracing gradient descent](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 19920–19930. Curran Associates, Inc.
- Zhen Qin, Zhaomin Wu, Bingsheng He, and Shuiguang Deng. 2025. [Federated data-efficient instruction tuning for large language models](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 15550–15568, Vienna, Austria. Association for Computational Linguistics.
- Yi Ren, Yanhui Li, Tianyi Zhang, and Diandong Liu. 2026. [CPQS-tuning: A model self-perception-based data filtering algorithm for efficient instruction fine-tuning](#). In *The Fourteenth International Conference on Learning Representations*.
- Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc Le, Ed Chi, Denny Zhou, and Jason Wei. 2023. [Challenging BIG-bench tasks and whether chain-of-thought can solve them](#). In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 13003–13051, Toronto, Canada. Association for Computational Linguistics.
- Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Miles Wang, Tom Dupré la Tour, Olivia Watkins, Alex Makelov, Ryan A. Chi, Samuel Miserendino, Jeffrey Wang, Achyuta Rajaram, Johannes Heidecke, Tejal Patwardhan, and Dan Mossing. 2025. [Persona features control emergent misalignment](#). *Preprint*, arXiv:2506.19823.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. [Chain-of-thought prompting elicits reasoning in large language models](#). In *Advances in Neural Information Processing Systems*.

Mengzhou Xia, Sadhika Malladi, Suchin Gururangan, Sanjeev Arora, and Danqi Chen. 2024. [LESS: Selecting influential data for targeted instruction tuning](#). In *International Conference on Machine Learning*.

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, and 41 others. 2025. [Qwen3 technical report](#). *Preprint*, arXiv:2505.09388.

Chih-Kuan Yeh, Joon Kim, Ian En-Hsu Yen, and Pradeep K Ravikumar. 2018. Representer point selection for explaining deep neural networks. *Advances in neural information processing systems*, 31.

Zichun Yu, Fei Peng, Jie Lei, Arnold Overwijk, Scott Yih, and Chenyan Xiong. 2025. [Group-level data selection for efficient pretraining](#). In *Advances in Neural Information Processing Systems*, volume 38, pages 154744–154770. Curran Associates, Inc.

Qi Zhang, Yiming Zhang, Haobo Wang, and Junbo Zhao. 2024. [RECAST: External knowledge guided data-efficient instruction tuning](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 10911–10921, Bangkok, Thailand. Association for Computational Linguistics.

Yaowei Zheng, Richong Zhang, Junhao Zhang, Yanhan Ye, and Zheyang Luo. 2024. [LlamaFactory: Unified efficient fine-tuning of 100+ language models](#). In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*, pages 400–410, Bangkok, Thailand. Association for Computational Linguistics.

A Repeated-Run Uncertainty

The three-run evaluation separates systematic performance differences from single-run variation. Tables 10 and 11 report dispersion for the main capability improvement and reverse repair results.

B Additional Robustness Analyses

PRISM transfers its repair advantage to benchmark-independent target pairs. We derive target pairs from BeaverTails (Ji et al., 2023), a broad safety-preference dataset that is independent of the MASK evaluation benchmark, and use them to repair Mistake medical mixtures. Table 12 shows that PRISM obtains the lowest dishonesty rate at the 1%, 10%, and 20% harmful-data ratios, demonstrating that its preference-aware direction remains effective when the target pairs are constructed independently of MASK.

PRISM extends to 2B–4B models. Table 13 extends capability improvement to Gemma-2-2B (Team et al., 2024) and Qwen3-4B (Yang et al., 2025). PRISM leads all five metrics on Gemma-2-2B and achieves the best BBH and TyDi EM results on Qwen3-4B, demonstrating effectiveness at smaller parameter scales.

C Related Work

Training data attribution and data selection. Training data attribution (TDA) studies how training data accounts for a model prediction, loss, or behavior. Prior work approaches this problem from several complementary perspectives. Data valuation methods, including leave-one-out and Shapley-style valuation, estimate the contribution of training examples to model utility, but often require many retraining or subset-evaluation runs (Ghorbani and Zou, 2019). Influence functions approximate the effect of upweighting a training example through local curvature (Koh and Liang, 2017), while representer points and gradient-tracing methods attribute predictions through model parameters or training trajectories (Yeh et al., 2018; Pruthi et al., 2020). More recent scalable TDA methods, such as datamodels, TRAK, and DataInf, improve applicability to large neural networks and parameter-efficient fine-tuning (Ilyas et al., 2022; Park et al., 2023; Kwon et al., 2024). These methods differ in estimator and computational trade-off, but share a common interface: given a target signal, they assign candidate training examples an attribution score. Data selection turns this attribution interface into an operational budgeted decision, retaining helpful examples or removing harmful ones. In LLM fine-tuning, LESS selects influential data for target instruction-tuning tasks (Xia et al., 2024), IF-GUIDE traces undesirable generations back to training data for detoxifica-

Method	Llama-2-7B					Llama-2-13B				
	MMLU	BBH	TyDi F1	TyDi EM	MATH-500	MMLU	BBH	TyDi F1	TyDi EM	MATH-500
RANDOM	0.46	1.04	0.71	1.34	0.43	0.18	0.74	1.83	1.91	0.43
LESS	0.84	1.25	1.15	2.35	0.80	0.55	1.25	1.88	1.66	0.43
VECFILTER	0.32	1.50	0.66	0.86	0.76	0.05	1.23	1.70	2.12	0.52
PROJDIFF	0.82	0.21	1.09	2.24	0.25	0.70	0.46	1.49	1.85	0.80
CONCEPTINF	0.86	1.45	1.71	1.93	0.66	1.11	1.67	2.09	1.89	0.50
IF-GUIDE	1.44	0.36	1.24	2.23	1.15	0.88	0.67	1.52	1.74	1.43
PRISM	1.19	0.94	0.86	1.69	0.63	0.48	2.18	0.92	1.42	0.88

Method	Qwen3-8B					Llama-3.1-8B-Instruct				
	MMLU	BBH	TyDi F1	TyDi EM	MATH-500	MMLU	BBH	TyDi F1	TyDi EM	MATH-500
RANDOM	0.23	0.81	1.08	1.84	0.87	0.16	1.89	1.12	1.62	2.01
LESS	0.08	0.46	1.71	2.89	0.87	0.19	1.02	1.80	1.77	1.26
VECFILTER	0.42	0.14	2.00	2.05	0.38	0.24	1.96	0.79	1.87	1.38
PROJDIFF	0.39	0.28	1.06	1.15	1.23	0.15	0.65	1.51	1.98	0.66
CONCEPTINF	0.16	1.16	0.75	0.73	1.09	0.38	0.40	1.54	2.77	2.40
IF-GUIDE	0.39	1.64	1.89	2.02	1.37	0.67	0.38	2.45	1.60	2.20
PRISM	0.87	0.46	1.50	3.18	0.63	0.65	0.84	2.29	2.35	0.80

Table 10: Standard deviations over three runs for the capability improvement results in Table 2.

Method	Llama-3.1-8B-Instruct								Qwen3-8B				Qwen3-14B							
	Mistake math				Mistake medical				Mistake math				Insecure code				Mistake math			
Harmful ratio	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%
Mixed	2.30	1.14	2.57	2.03	3.72	0.66	2.10	2.51	1.84	0.75	1.12	2.58	0.92	0.97	1.05	0.56	2.56	2.03	1.34	2.22
RANDOM	1.07	1.16	1.03	4.05	2.19	2.40	1.30	4.21	0.21	1.41	1.65	0.56	1.47	1.47	0.97	2.64	0.21	2.35	0.56	0.67
LESS	0.56	2.66	2.71	1.06	2.77	4.05	1.73	2.01	2.91	0.63	1.61	2.14	2.16	0.97	0.39	2.19	0.76	0.36	0.56	1.82
VECFILTER	0.55	1.45	1.08	2.76	0.53	1.39	2.22	1.69	1.19	2.81	3.52	0.80	1.27	1.67	2.95	1.93	2.64	0.48	0.56	0.21
PROJDIFF	2.07	0.64	1.56	2.63	3.21	1.27	2.76	2.68	0.92	0.74	1.50	2.27	0.73	0.56	1.97	1.86	1.65	0.71	1.30	2.11
CONCEPTINF	1.74	1.48	1.00	1.76	2.27	4.21	4.85	1.87	2.43	0.97	2.22	2.36	1.38	1.67	0.46	1.89	1.70	1.32	0.73	0.82
IF-GUIDE	2.95	0.66	2.01	1.47	0.94	0.46	0.76	3.36	2.60	3.77	1.75	3.68	2.03	0.66	0.56	1.32	0.29	1.72	2.48	0.55
PRISM	0.48	0.46	1.55	0.54	2.06	4.71	1.30	3.96	0.80	1.59	0.79	1.28	1.28	1.72	0.62	1.93	0.95	1.28	1.11	1.82

Table 11: Standard deviations over three runs for the dishonesty rates in reverse repair reported in Table 6.

Method	1%	10%	20%	50%
LESS	51.83	51.22	57.61	68.16
VECFILTER	50.06	53.97	59.15	71.05
PROJDIFF	52.50	55.36	58.41	73.57
CONCEPTINF	51.40	52.32	57.21	71.18
PRISM	48.90	50.00	55.54	69.22

Table 12: Dishonesty rate ($\downarrow$) after filtering and retraining Llama-3.1-8B-Instruct on Mistake medical using BeaverTails-derived target pairs.

tion (Coalson et al., 2025), and influence function selection has also been studied for reasoning data (Humane et al., 2025). Related work further applies data-efficient fine-tuning to recommendation (Lin et al., 2024) and studies balancing strategies for influence-based instruction tuning across diverse capabilities (Dai et al., 2025). PRISM follows this target-conditioned TDA and data-selection paradigm. Rather than proposing another influence estimator, PRISM improves the

target signal underlying the attribution score.

Hard-example-aware, diversity-aware, and balanced selection address complementary stages of the pipeline. Hard-example criteria emphasize candidate examples or tasks that expose missing capabilities, diversity constraints reduce redundancy and broaden subset coverage, and balanced influence selection allocates capacity across capabilities to protect underrepresented groups (Dai et al., 2025). PRISM instead defines the preference-aware target direction used to score individual candidates. These principles are not mutually exclusive: PRISM scores can be combined with diversity-aware reranking or balance constraints when long-tail, worst-group, or multi-capability coverage is a primary objective.

Scalable and low-budget data selection. A complementary line of work improves the scalability of data selection itself. Influence-preserving proxies approximate gradient-based influence structure for

Model	Method	MMLU	BBH	TyDi F1	TyDi EM	MATH-500
Gemma-2-2B	LESS	54.58	26.17	57.63	42.76	7.83
	VECFILTER	55.20	26.48	47.00	31.49	6.92
	CONCEPTINF	54.60	26.08	48.36	31.31	5.00
	PRISM	55.57	26.57	58.41	44.73	8.17
Qwen3-4B	LESS	69.88	73.48	71.57	53.88	21.42
	VECFILTER	70.64	74.01	70.85	51.92	22.75
	CONCEPTINF	71.09	72.28	61.65	39.72	25.33
	PRISM	70.87	74.14	70.62	54.64	20.83

Table 13: Capability improvement performance ($\uparrow$) of smaller models after fine-tuning on 5% selected data.

LLM fine-tuning (Chen et al., 2026), while influence distillation learns scalable per-example selection weights from influence signals (Nikdan et al., 2025). DELIFT studies data-efficient instruction fine-tuning without relying on influence functions (Agarwal et al., 2025). Other settings introduce additional constraints or selection granularity, such as federated instruction tuning (Qin et al., 2025) and group-level data selection for pretraining (Yu et al., 2025). These works mainly address how to make data selection cheaper, broader, or more robust across settings. PRISM is orthogonal: it focuses on how to define the target direction more precisely, and can in principle be combined with more scalable influence estimators or proxies.

Target behavior representation. Before training examples can be scored, the target behavior must be represented as a direction or signal. Existing methods build such representations in different spaces. Concept Influence and Persona Vectors construct activation-space directions, often by averaging representations of examples that express a concept or trait (Kowal et al., 2026; Chen et al., 2025). Influence-based data selection constructs parameter-space directions, often by averaging gradients from target examples (Xia et al., 2024). IF-GUIDE further uses a contrastive positive-negative construction in gradient space (Coalson et al., 2025). Other data-filtering methods incorporate model-internal or external signals, such as model self-perception in CPQS-Tuning (Ren et al., 2026) and external knowledge in RECOST (Zhang et al., 2024). Although these approaches differ in representation space and information source, they typically do not distinguish target examples by how close they are to the model behavior. PRISM relaxes this assumption by weighting target examples according to the model’s preference, producing a preference-aware target direction for training-data scoring.

D Benchmark-Derived Reverse Repair Protocol

All data-selection methods are evaluated under the same benchmark-derived reverse repair protocol from Section 3.2. For each mixed dataset, we first fine-tune the model on the unfiltered SFT data and evaluate both the pre-SFT and post-SFT models on the MASK provided-facts benchmark. We then collect the benchmark queries whose behavior changes undesirably and pair the pre-SFT response with the post-SFT response on the same query. This shared paired-response set is the only benchmark-derived signal provided to the data-selection methods. Each method converts this signal into a score over the original SFT examples, removes the same fixed fraction of high-risk examples within a setting, and retrains from the same base model on the retained data. Thus, the experiments compare scoring rules while holding both the supervision source and the removal budget fixed. Section 4.3 varies the retained-data ratio.

E Benchmark Degradation from Harmful SFT Data

Before evaluating data repair, we verify that the data families create the failure mode of interest. Table 14 compares the base model, pure harmful SFT, and pure correct SFT conditions. The pure harmful conditions increase dishonesty rate in several model-family pairs, while pure correct SFT generally lowers it. This sensitivity motivates filtering mixed SFT data before retraining.

The main ranking and filtering tables focus on emergent misalignment settings, i.e., model-family pairs where mixed harmful data produces a clear benchmark degradation relative to the base model. Settings without this degradation are not the primary data-repair target, because there is no observed benchmark failure to repair.

Model	Data family	Base	Wrong	Correct
Llama-3.1-8B-Inst.	Insecure code		49.27	49.27
	Mistake math	53.29	68.61	47.45
	Mistake medical		74.45	47.45
Qwen3-8B	Insecure code		52.19	49.27
	Mistake math	52.92	64.96	45.26
	Mistake medical		52.20	35.04
Qwen3-14B	Insecure code		61.68	50.73
	Mistake math	59.85	70.80	55.84
	Mistake medical		56.57	36.50

Table 14: Dishonesty rate ($\downarrow$) under base, pure harmful, and pure correct SFT conditions on the MASK provided-facts benchmark.

F Full Ranking Quality for Reverse Repair

Table 15 reports the per-setting AUROC values summarized in Table 5. The complete results cover all 20 combinations of model, data family, and harmful-data ratio used in the main ranking comparison for reverse repair.

G Full Results for Additional-Component Baselines

Tables 16 and 17 report the complete per-setting values summarized in Table 8. We include PRISM in these appendix tables to make the comparison with additional-component baselines explicit.

H Reproducibility Details

Curvature approximation. Both capability improvement and reverse repair approximate the inverse-Hessian-vector product $H^{-1}v$ with Eigenvalue-Corrected Kronecker-Factored Approximate Curvature (EK-FAC) (George et al., 2018). EK-FAC represents local curvature in a Kronecker-factored eigenbasis and corrects the eigenvalues in that basis, providing a scalable approximation to the curvature preconditioner.

Target and test splits. For MMLU, BBH, and TyDi QA in capability improvement, we follow the target-set and test-set splits released by LESS (Xia et al., 2024). For MATH-500, we randomly split the examples into target and test sets with a 1:4 ratio. The target split is used to define the target behavior for data selection, and the held-out split is used only for evaluation.

Agent safety guardrail setting. For the agent safety guardrail experiment in Section 4.1, the candidate training pool contains about 2K private safety-related examples. We use ATBench (Li et al.,

Llama-3.1-8B-Instruct								
Method	Mistake math				Mistake medical			
	1%	10%	20%	50%	1%	10%	20%	50%
RANDOM	0.46	0.50	0.51	0.50	0.50	0.48	0.51	0.51
LESS	0.22	0.28	0.29	0.31	0.29	0.36	0.44	0.47
VECFILTER	0.79	0.69	0.27	0.29	0.69	0.13	0.54	0.54
PROJDIFF	0.61	0.57	0.35	0.35	0.58	0.32	0.40	0.46
CONCEPTINF	0.73	0.65	0.44	0.45	0.63	0.36	0.50	0.53
IF-GUIDE	0.53	0.58	0.61	0.63	0.32	0.47	0.38	0.44
PRISM	0.79	0.70	0.71	0.69	0.63	0.63	0.58	0.59

Qwen3-8B				
Method	Mistake math			
	1%	10%	20%	50%
RANDOM	0.46	0.50	0.51	0.50
LESS	0.20	0.25	0.29	0.30
VECFILTER	0.56	0.56	0.48	0.66
PROJDIFF	0.51	0.51	0.51	0.52
CONCEPTINF	0.41	0.40	0.38	0.37
IF-GUIDE	0.52	0.55	0.56	0.55
PRISM	0.66	0.63	0.70	0.68

Qwen3-14B								
Method	Insecure code				Mistake math			
	1%	10%	20%	50%	1%	10%	20%	50%
RANDOM	0.50	0.49	0.50	0.50	0.46	0.50	0.51	0.50
LESS	0.34	0.37	0.39	0.44	0.12	0.16	0.24	0.35
VECFILTER	0.33	0.32	0.35	0.51	0.66	0.63	0.67	0.67
PROJDIFF	0.42	0.44	0.45	0.50	0.53	0.53	0.56	0.54
CONCEPTINF	0.67	0.55	0.54	0.51	0.69	0.40	0.21	0.22
IF-GUIDE	0.46	0.53	0.59	0.64	0.51	0.54	0.63	0.65
PRISM	0.55	0.67	0.66	0.71	0.55	0.67	0.69	0.72

Table 15: Per-setting AUROC ($\uparrow$) in the ranking comparison for reverse repair. Panels are grouped by model and data family, with harmful-data ratios shown explicitly. Each value is from one run; bold marks the best or tied-best method in each setting.

2026), a diverse agent-trajectory benchmark for safety evaluation and diagnosis, as the evaluation set. We scan selection rates from 20% to 80%, compare PRISM with random selection under the same rate, and also report the base model and full-data SFT results.

SFT hyperparameters. We use LLaMA-Factory (Zheng et al., 2024) for supervised fine-tuning. Table 18 reports the specific parameters used in the capability improvement and reverse repair SFT runs.

I Proofs for Section 2.3

I.1 Proof of Theorem 1

Proof. For a single training example, define the infinitesimally upweighted training objective as

$$\mathcal{L}_{\epsilon,i}(\theta) = \mathcal{L}(\theta) + \epsilon \ell(x_i, y_i; \theta). \quad (17)$$

Method	Llama-3.1-8B-Instruct								Qwen3-8B				Qwen3-14B							
Data family	Mistake math				Mistake medical				Mistake math				Insecure code				Mistake math			
Harmful ratio	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%
SAE-FEAT	0.41	0.39	0.41	0.41	0.37	0.36	0.39	0.38	0.47	0.47	0.48	0.49	0.50	0.52	0.52	0.51	0.39	0.41	0.40	0.41
LLM JUDGE	0.55	0.51	0.52	0.52	0.92	0.95	0.95	0.98	0.53	0.50	0.51	0.52	0.71	0.65	0.64	0.61	0.50	0.50	0.50	0.51
PRISM	0.79	0.70	0.71	0.69	0.63	0.63	0.58	0.59	0.66	0.63	0.70	0.68	0.55	0.67	0.66	0.71	0.55	0.67	0.69	0.72

Table 16: Full data ranking quality for additional-component baselines on the emergent misalignment settings. Values are AUROC ($\uparrow$). Bold marks the best method among PRISM and additional-component baselines in each setting.

Method	Llama-3.1-8B-Instruct								Qwen3-8B				Qwen3-14B							
Data family	Mistake math				Mistake medical				Mistake math				Insecure code				Mistake math			
Harmful ratio	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%	1%	10%	20%	50%
SAE-FEAT	45.99	48.18	51.83	61.31	47.81	51.10	52.56	64.60	45.62	52.19	52.19	62.77	51.83	62.41	66.06	64.23	56.20	66.06	65.69	66.79
LLM JUDGE	48.91	49.27	53.65	58.39	47.45	47.45	49.27	66.06	51.83	53.65	57.66	63.14	55.11	58.03	64.23	62.77	55.47	62.41	66.79	68.25
PRISM	46.47	49.57	53.36	59.49	49.64	49.76	54.01	62.49	47.75	50.36	53.16	59.25	53.65	59.08	61.10	61.51	58.78	61.22	64.84	67.40

Table 17: Full post-filtering retraining results for additional-component baselines on the emergent misalignment settings. Values are dishonesty rates ($\downarrow$). Bold marks the best method among PRISM and additional-component baselines in each setting. The PRISM row reproduces the three-run means from Table 6.

Parameter	Capability improvement	Reverse repair
GPUs	1 $\times$ H200	1 $\times$ H200
Learning rate	2.0×10^{-5}	1.0×10^{-4}
Epochs	4	1
Batch size	1	4
Gradient accumulation	128	1
Max sequence length	2048	2048
Weight decay	0.0	0
LoRA rank	128	32
LoRA α	512	64
LoRA modules	q, k, v, o projections	All

Table 18: Fine-tuning hyperparameters used in the LLaMA-Factory SFT runs. For capability improvement, batch size is the per-device train batch size; with one GPU, gradient accumulation of 128 gives an effective global batch size of 128. For reverse repair, LoRA α follows the LLaMA-Factory default $2r$ when not explicitly specified in the configuration.

The local optimum $\theta_{\epsilon,i}$ satisfies $\nabla_{\theta} \mathcal{L}_{\epsilon,i}(\theta_{\epsilon,i}) = 0$. Differentiating this stationarity condition with respect to ϵ at $\epsilon = 0$ gives

$$H \frac{d\theta_{\epsilon,i}}{d\epsilon} \Big|_{\epsilon=0} + g_{z_i}(\theta) = 0, \quad (18)$$

and therefore $d\theta_{\epsilon,i}/d\epsilon|_{\epsilon=0} = -H^{-1}g_{z_i}(\theta)$. Using $\nabla_{\theta} \mathcal{K}(\theta) = -g_{\text{KL}}(\theta)$, the chain rule yields

$$\begin{aligned} \frac{d}{d\epsilon} \mathcal{K}(\theta_{\epsilon,i}) \Big|_{\epsilon=0} &= \nabla_{\theta} \mathcal{K}(\theta)^{\top} (-H^{-1}g_{z_i}(\theta)) \\ &= g_{z_i}(\theta)^{\top} H^{-1}g_{\text{KL}}(\theta). \end{aligned} \quad (19)$$

For a subset S of size m , the same argument replaces the single-example gradient with $\bar{g}_S =$

$\frac{1}{m} \sum_{z_i \in S} g_{z_i}(\theta)$, which gives

$$\frac{d}{d\epsilon} \mathcal{K}(\theta_{S,\epsilon}) \Big|_{\epsilon=0} = \frac{1}{m} \sum_{z_i \in S} h_{\pi}(z_i; \theta). \quad (20)$$

Thus, under a fixed size- m budget, maximizing the first-order reward gain is equivalent to maximizing the sum of h_{π} -scores, which is achieved by selecting the top- m examples. $\square$

I.2 Closed form and upper bounds for paired KL

For this appendix, define P_q^{θ} as the normalized distribution over the target-positive and target-negative responses:

$$P_q^{\theta} = (\pi_q(\theta), 1 - \pi_q(\theta)), \quad (21)$$

with target-negative reference $A_q = (0, 1)$. We also write

$$\begin{aligned} P(\theta) &= \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \pi_q(\theta), \\ R_{\text{pair}}(\theta) &= \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \mathbf{1}[\pi_q(\theta) \geq 1/2]. \end{aligned} \quad (22)$$

Proposition 1 (Closed form and upper bounds). *For each $q \in \mathcal{Q}$,*

$$D_{\text{KL}}(A_q \| P_q^{\theta}) = -\log(1 - \pi_q(\theta)). \quad (23)$$

Equivalently, if

$$m_q(\theta) = \log \bar{p}_{\theta}(y_q^+ | q) - \log \bar{p}_{\theta}(y_q^- | q), \quad (24)$$

then

$$\begin{aligned}\pi_q(\theta) &= \sigma(m_q(\theta)), \\ D_{\text{KL}}(A_q \| P_q^\theta) &= \log(1 + e^{m_q(\theta)}).\end{aligned}\quad (25)$$

Moreover,

$$P(\theta) \leq \mathcal{K}(\theta), \quad (26)$$

and

$$R_{\text{pair}}(\theta) \leq 2P(\theta) \leq 2\mathcal{K}(\theta). \quad (27)$$

Proof. Since $A_q = (0, 1)$,

$$\begin{aligned}D_{\text{KL}}(A_q \| P_q^\theta) &= \log \frac{1}{1 - \pi_q(\theta)} \\ &= -\log(1 - \pi_q(\theta)),\end{aligned}\quad (28)$$

which proves Eq. (23). By definition of the margin,

$$\pi_q(\theta) = \frac{1}{1 + \exp(-m_q(\theta))} = \sigma(m_q(\theta)), \quad (29)$$

and substituting this into the previous display gives Eq. (25).

For $x \in [0, 1]$, the elementary inequality $x \leq -\log(1-x)$ gives $\pi_q(\theta) \leq D_{\text{KL}}(A_q \| P_q^\theta)$. Averaging over q proves Eq. (26). Also, $\mathbf{1}[x \geq 1/2] \leq 2x$ for $x \in [0, 1]$. Applying this to $\pi_q(\theta)$ and averaging yields

$$R_{\text{pair}}(\theta) \leq 2P(\theta) \leq 2\mathcal{K}(\theta), \quad (30)$$

which proves Eq. (27). $\square$

I.3 Proof of the paired-gradient identity

Proof. By Proposition 1,

$$\mathcal{K}(\theta) = \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \log(1 + e^{m_q(\theta)}). \quad (31)$$

For one summand,

$$\nabla_\theta \log(1 + e^{m_q}) = \sigma(m_q) \nabla_\theta m_q. \quad (32)$$

Moreover,

$$\begin{aligned}\nabla_\theta m_q &= \nabla_\theta \log \bar{p}_\theta(y_q^+ | q) - \nabla_\theta \log \bar{p}_\theta(y_q^- | q) \\ &= \nabla_\theta \bar{\ell}(q, y_q^-; \theta) - \nabla_\theta \bar{\ell}(q, y_q^+; \theta).\end{aligned}\quad (33)$$

Since $\sigma(m_q) = \pi_q$, each summand contributes $-\pi_q(\theta)(\nabla_\theta \bar{\ell}(q, y_q^+; \theta) - \nabla_\theta \bar{\ell}(q, y_q^-; \theta))$. Averaging over $\mathcal{Q}$ gives $\nabla_\theta \mathcal{K}(\theta) = -g_{\text{KL}}(\theta)$. $\square$

I.4 Proof of the optimal local direction

Proof. By Eq. (10), the first-order alignment with the preference-aware target direction is

$$-\nabla_\theta \mathcal{K}(\theta)^\top \delta\theta = g_{\text{KL}}(\theta)^\top \delta\theta. \quad (34)$$

Thus the constrained problem is to maximize $g_{\text{KL}}(\theta)^\top \delta\theta$ subject to $\delta\theta^\top H \delta\theta \leq c$. By Cauchy-Schwarz in the H^{-1} geometry,

$$\begin{aligned}(g_{\text{KL}}(\theta)^\top \delta\theta)^2 &= ((H^{-1/2} g_{\text{KL}}(\theta))^\top (H^{1/2} \delta\theta))^2 \\ &\leq (g_{\text{KL}}(\theta)^\top H^{-1} g_{\text{KL}}(\theta)) \\ &\quad \cdot (\delta\theta^\top H \delta\theta) \\ &\leq c g_{\text{KL}}(\theta)^\top H^{-1} g_{\text{KL}}(\theta).\end{aligned}\quad (35)$$

Equality holds iff the Cauchy-Schwarz vectors are proportional, equivalently

$$\delta\theta \propto H^{-1} g_{\text{KL}}(\theta). \quad (36)$$

Normalizing this direction to saturate the budget gives

$$\delta\theta^* = \sqrt{\frac{c}{g_{\text{KL}}(\theta)^\top H^{-1} g_{\text{KL}}(\theta)}} H^{-1} g_{\text{KL}}(\theta). \quad (37)$$

The resulting first-order gain is

$$-\nabla_\theta \mathcal{K}(\theta)^\top \delta\theta^* = \sqrt{c} \|g_{\text{KL}}\|_{H^{-1}}. \quad (38)$$

$\square$

I.5 Proof of the directional gap

Proof. Let $\langle u, v \rangle_{H^{-1}} = u^\top H^{-1} v$ and $\|u\|_{H^{-1}} = \sqrt{\langle u, u \rangle_{H^{-1}}}$. To use the same local budget as PRISM, the equal-aggregation update is

$$\delta\theta_{\text{eq}} = \frac{\sqrt{c}}{\|g_0\|_{H^{-1}}} H^{-1} g_0. \quad (39)$$

Using Eq. (10) and Eq. (39),

$$\begin{aligned}-\nabla_\theta \mathcal{K}(\theta)^\top \delta\theta_{\text{eq}} &= g_{\text{KL}}(\theta)^\top \delta\theta_{\text{eq}} \\ &= \sqrt{c} \frac{\langle g_{\text{KL}}, g_0 \rangle_{H^{-1}}}{\|g_0\|_{H^{-1}}} \\ &\leq \sqrt{c} \|g_{\text{KL}}\|_{H^{-1}} \\ &= -\nabla_\theta \mathcal{K}(\theta)^\top \delta\theta^*.\end{aligned}\quad (40)$$

The optimal value from Eq. (37) is $\sqrt{c} \|g_{\text{KL}}\|_{H^{-1}}$. By Cauchy-Schwarz in the H^{-1} metric,

$$\langle g_{\text{KL}}, g_0 \rangle_{H^{-1}} \leq \|g_{\text{KL}}\|_{H^{-1}} \|g_0\|_{H^{-1}}. \quad (41)$$

Equivalently, the same argument can be written as the fraction of PRISM's optimal first-order reward gain retained by equal aggregation:

$$\begin{aligned} \frac{-\nabla_{\theta} \mathcal{K}(\theta)^{\top} \delta \theta_{\text{eq}}}{-\nabla_{\theta} \mathcal{K}(\theta)^{\top} \delta \theta^*} &= \cos_{H^{-1}}(g_{\text{KL}}(\theta), g_0(\theta)) \\ &= \frac{\langle g_{\text{KL}}, g_0 \rangle_{H^{-1}}}{\|g_{\text{KL}}\|_{H^{-1}} \|g_0\|_{H^{-1}}} \\ &\leq 1. \end{aligned} \quad (42)$$

This proves the inequality in Eq. (15). Equality holds iff $g_0(\theta)$ and $g_{\text{KL}}(\theta)$ are positively collinear. $\square$

I.6 Proof of the equal-budget ranking comparison

Proof. Applying Eq. (13) to $S = S_{\pi}^{(m)}$ and $S = S_0^{(m)}$, then subtracting the two first-order expansions, gives Eq. (16).

The coefficient in Eq. (16) is

$$\Delta_m = \frac{1}{m} \left(\sum_{z_i \in S_{\pi}^{(m)}} h_{\pi}(z_i; \theta) - \sum_{z_i \in S_0^{(m)}} h_{\pi}(z_i; \theta) \right). \quad (43)$$

It remains only to show $\Delta_m \geq 0$. By construction, $S_{\pi}^{(m)}$ maximizes $\sum_{z \in S} h_{\pi}(z; \theta)$ among all size- m subsets of $\mathcal{D}$. Since $S_0^{(m)}$ is one such subset,

$$\sum_{z \in S_{\pi}^{(m)}} h_{\pi}(z; \theta) \geq \sum_{z \in S_0^{(m)}} h_{\pi}(z; \theta), \quad (44)$$

which proves Eq. (43). If the inequality is strict, then $\Delta_m > 0$, and Eq. (16) yields a larger first-order target-preference reward gain for $S_{\pi}^{(m)}$. $\square$

J An Information-Theoretic View of Preference Weighting

This appendix gives a complementary interpretation of the preference-weighted target signal. It is independent of the main exposition: the goal is only to show why a pair with stronger target-positive preference carries more information about the target behavior than a low-preference pair.

For a paired target q , define the paired log-odds

$$r_q(\theta) = \log \bar{p}_{\theta}(y_q^+ | q) - \log \bar{p}_{\theta}(y_q^- | q). \quad (45)$$

Then the paired target-positive preference is

$$\pi_q(\theta) = \frac{\bar{p}_{\theta}(y_q^+ | q)}{\bar{p}_{\theta}(y_q^+ | q) + \bar{p}_{\theta}(y_q^- | q)} = \sigma(r_q(\theta)). \quad (46)$$

The target-negative reference distribution assigns all mass to the target-negative response. The information required to rule out this target-negative reference under the model's paired distribution is the paired KL risk

$$\begin{aligned} I_q(\theta) &= D_{\text{KL}}(A_q \| P_q^{\theta}) \\ &= -\log(1 - \pi_q(\theta)) \\ &= \log(1 + \exp r_q(\theta)). \end{aligned} \quad (47)$$

Thus I_q is the self-information of the target-negative response under the model's paired distribution. Its marginal sensitivity to the target-positive paired log-odds is

$$\frac{\partial I_q(\theta)}{\partial r_q(\theta)} = \pi_q(\theta). \quad (48)$$

This identity gives an information-theoretic meaning to the weight: increasing the target-positive log-odds by the same small amount changes the target-preference reward more for high-preference pairs than for low-preference pairs.

Using

$$\begin{aligned} \nabla_{\theta} r_q(\theta) &= \nabla_{\theta} \log \bar{p}_{\theta}(y_q^+ | q) \\ &\quad - \nabla_{\theta} \log \bar{p}_{\theta}(y_q^- | q) \\ &= -\nabla_{\theta} \bar{\ell}(q, y_q^+; \theta) + \nabla_{\theta} \bar{\ell}(q, y_q^-; \theta), \end{aligned} \quad (49)$$

we obtain

$$\begin{aligned} -\nabla_{\theta} I_q(\theta) &= \pi_q(\theta) \nabla_{\theta} \bar{\ell}(q, y_q^+; \theta) \\ &\quad - \pi_q(\theta) \nabla_{\theta} \bar{\ell}(q, y_q^-; \theta). \end{aligned} \quad (50)$$

Averaging over target pairs recovers the preference-weighted target direction in Eq. (10). Hence the weighted direction is the negative gradient of the paired information risk.

The same view also compares the preference-weighted and equal-weight characterizations of the target behavior. Let

$$d_q(\theta) = \nabla_{\theta} \bar{\ell}(q, y_q^+; \theta) - \nabla_{\theta} \bar{\ell}(q, y_q^-; \theta), \quad (51)$$

and define

$$\begin{aligned} g_{\text{KL}}(\theta) &= \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} \pi_q(\theta) d_q(\theta), \\ g_0(\theta) &= \frac{1}{|\mathcal{Q}|} \sum_{q \in \mathcal{Q}} d_q(\theta). \end{aligned} \quad (52)$$

For compactness in the following local-geometry comparison, write $a_{\theta} = g_{\text{KL}}(\theta)$ and $b_{\theta} = g_0(\theta)$.

For any local update $\delta\theta$, its first-order alignment with the negative gradient of the average paired information risk is

$$\Delta I(\delta\theta) = a_\theta^\top \delta\theta + o(\|\delta\theta\|). \quad (53)$$

Let $\langle u, v \rangle_{H^{-1}} = u^\top H^{-1}v$ and $\|u\|_{H^{-1}} = \sqrt{\langle u, u \rangle_{H^{-1}}}$. Under the same local budget $\delta\theta^\top H \delta\theta \leq c$, the maximum first-order alignment is

$$\max_{\delta\theta^\top H \delta\theta \leq c} a_\theta^\top \delta\theta = \sqrt{c} \|a_\theta\|_{H^{-1}}, \quad (54)$$

achieved by $\delta\theta \propto H^{-1}a_\theta$. If instead one uses the equally weighted target direction g_0 and normalizes it to the same budget,

$$\delta\theta_{\text{eq}} = \frac{\sqrt{c}}{\|b_\theta\|_{H^{-1}}} H^{-1}b_\theta, \quad (55)$$

then its first-order alignment is

$$a_\theta^\top \delta\theta_{\text{eq}} = \sqrt{c} \frac{\langle a_\theta, b_\theta \rangle_{H^{-1}}}{\|b_\theta\|_{H^{-1}}} \leq \sqrt{c} \|a_\theta\|_{H^{-1}}. \quad (56)$$

Equivalently, the ratio between the alignment of the equal-weight direction and the optimal weighted direction is

$$\frac{a_\theta^\top \delta\theta_{\text{eq}}}{\sqrt{c} \|a_\theta\|_{H^{-1}}} = \cos_{H^{-1}}(a_\theta, b_\theta) \leq 1. \quad (57)$$

Equality holds only when $g_0(\theta)$ and $g_{\text{KL}}(\theta)$ are aligned in the H^{-1} metric. Therefore, unless all target pairs have the same preference weight or their gradient contrasts point in the same direction, equal aggregation gives a less faithful first-order characterization of the target behavior than the preference-weighted information-risk direction.