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

Anonymous ACL submission

Abstract

As LLMs continue to scale up, improving training efficiency heavily relies on effective data utilization. Data selection mitigates this issue by allocating the limited training budget to high-value examples that optimally facilitate the model's target behavior. Most existing approaches define target behavior via a set of target examples and score candidate training data based on their estimated influence on these samples. However, such methods uniformly treat all target examples as equally important, ignoring the varying relevance of individual examples to model optimization. Specifically, target examples that align closely with the model's inherent behavior deliver stronger supervisory signals, whereas discrepant examples yield only weak and ineffective local guidance. We propose PRISM, a **Pr**eference-aware **I**nfluence function based Data **S**election **M**ethod. It leverages model preference to assign weights to target examples and builds a preference-aware target direction. PRISM evaluates candidate training samples according to their influence on this direction, and prioritizes data budget allocation to samples that effectively drive the model to match expected target behavior. Theoretical analysis verifies that weighted preference construction generates a superior first-order gradient direction for boosting target preference, compared with uniform aggregation strategies. Extensive experiments covering diverse model architectures and parameter scales demonstrate that PRISM achieves better performance in efficient fine-tuning and safety-aligned supervised fine-tuning rectification. The results validate that accurate characterization of target behavior serves as the core of cost-effective data selection.

1 Introduction

As LLMs continue to scale, the costs of training and adaptation are steadily 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).

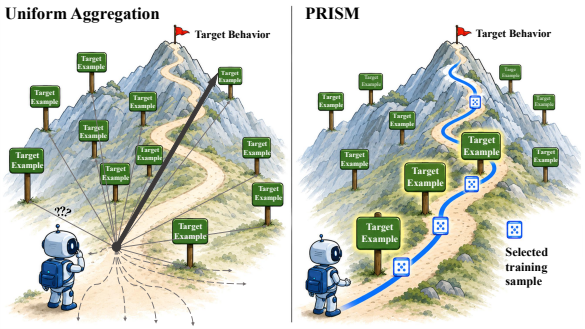

Figure 1: Motivation of PRISM. Equal aggregation treats all target examples equally, whereas PRISM emphasizes target examples closer to the model and focuses the budget on more actionable training examples.

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 the importance of target examples when constructing the target behavior and propose PRISM (**Pr**eference-aware **I**nfluence function based Data **S**election **M**ethod). PRISM quantifies the model's preference for each target example using the likelihood assigned to the response and uses this preference to weight the example when constructing the target direction. A higher preference indicates that the example corresponds to 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 candidate training examples by estimating their influence on this preference-aware target direction, thereby allocating the data budget to examples that are more likely to strengthen the target behavior efficiently.

Theoretically, we show that PRISM captures the first-order direction that most effectively increases target-behavior preference in the neighborhood of the model. Under a fixed selection budget, selecting examples along this direction maximizes the desired gain in target preference per unit of data. By contrast, equal aggregation mixes high- and low-preference target examples, causing the estimated direction to deviate from the behavioral pattern toward which the model can most readily be pushed, thereby reducing the utility of each selected example.

We conduct systematic experiments across model families and scales in two complementary data selection settings. First, to improve general capabilities, we define the target behavior in terms of factual knowledge, challenging reasoning, multilingual question answering, mathematical problem solving, and safety guardrail capabilities. Results show that, under the same data budget, training examples selected by PRISM lead to stronger fine-tuning performance. Second, in the safety-critical SFT repair setting 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 providing a more precise characterization of the target behavior.

2 Methodology

We now introduce PRISM (**Pr**eference-aware **I**nfluence function based Data **S**election **M**ethod), 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 paired target example (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

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

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

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

The corresponding token-average cross-entropy loss is

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

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

$$\bar{g}_{q,y_q} = \nabla_\vartheta \bar{\ell}(q, y_q; \vartheta)\big|_{\vartheta=\theta}, g_{z_i} = \nabla_\vartheta \ell(x_i, y_i; \vartheta)\big|_{\vartheta=\theta}. \tag{4}$$

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

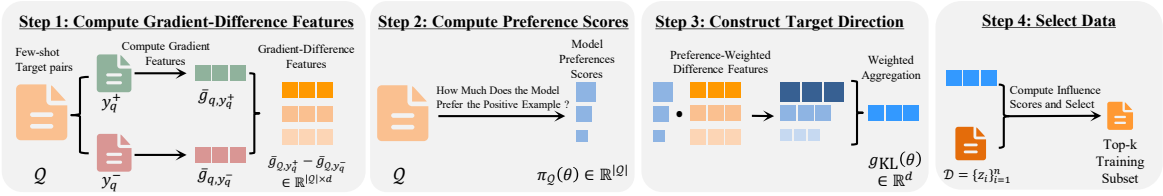

Figure 2: Overview of PRISM. Given paired target examples, PRISM estimates the model's preference for the target-positive response, constructs a preference-aware target direction, and scores candidate training examples by influence alignment with this direction.

Hessian of the training objective over the training distribution:

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

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

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

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 example is to the model's behavior. For each paired target example, we define

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

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,

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

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:

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

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 G.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

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

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.

is represented through the token-average loss in Eq. (3), we obtain

$$-\nabla_\theta\mathcal{K}(\theta) = \frac{1}{|\mathcal{Q}|}\sum_{q\in\mathcal{Q}}\pi_q(\theta)\Big(\bar{g}_{q,y_q^+}-\bar{g}_{q,y_q^-}\Big) \triangleq g_{\mathrm{KL}}(\theta). \tag{10}$$

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 examples with higher model preference contribute more to the direction because changing the model around those examples yields larger local gain in the reward. Appendix G.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:

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

A larger score means that the training signal of z_i is more aligned with the preference-aware target behavior 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 G.

Influence score as reward gain. We first justify the score used for ranking training examples. Theorem 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 upweighting z_i.

Theorem 1 (Influence score estimates target-preference 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 paired KL reward is*

$$\frac{d}{d\epsilon}\mathcal{K}(\theta_{\epsilon,i})\Big|_{\epsilon=0} = g_{z_i}(\theta)^\top H^{-1} g_{\mathrm{KL}}(\theta) \triangleq h_\pi(z_i;\theta). \tag{12}$$

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

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

Therefore, selecting the top-m examples by h_π maximizes the first-order paired KL reward gain under the fixed budget.

The proof follows the standard influence functions differentiation of an infinitesimally upweighted training objective and is given in Appendix G.1. Theorem 1 gives the operational meaning of the PRISM score. The Hessian-preconditioned alignment $g_{z_i}^\top H^{-1} g_{\mathrm{KL}}$ is not merely a similarity score; it estimates the first-order gain in target preference 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

$$g_0(\theta) = \frac{1}{|\mathcal{Q}|}\sum_{q\in\mathcal{Q}}\Big(\bar{g}_{q,y_q^+}-\bar{g}_{q,y_q^-}\Big), \quad h_0(z_i;\theta) = g_{z_i}(\theta)^\top H^{-1} g_0(\theta). \tag{14}$$

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 $\delta\theta^\top H\delta\theta \le c$, which fixes the update size in the local geometry*

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	39.69	47.26	31.30	4.83
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 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. Higher is better for all metrics, and bold marks the best method in each column.

around the model. Let $\delta\theta^\star$ be the H-normalized update induced by the PRISM direction $H^{-1}g_{\mathrm{KL}}$, and let $\delta\theta_{\mathrm{eq}}$ 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:

$$-\nabla_\theta \mathcal{K}(\theta)^\top \delta\theta^\star \geq -\nabla_\theta \mathcal{K}(\theta)^\top \delta\theta_{\mathrm{eq}}. \tag{15}$$

Equality holds only when g_0 is aligned with g_{KL} in the H^{-1} geometry.

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 paired KL reward,*

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

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 functions 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 samples selected along that direction can then deliver lower paired KL gain under the same data budget. Appendix G.4 proves the optimal local direction, Appendix G.5 gives an equivalent cosine-ratio interpretation, and Appendix G.6 proves the subset comparison.

3 Experiments

We evaluate whether PRISM improves budget efficiency in two complementary data-selection settings. The first setting selects data to improve general capabilities: a method receives a small training budget and the selected subset is evaluated by downstream fine-tuning performance. The second setting removes data to suppress emergent misalignment: a method ranks harmful training examples, filters them under a fixed removal budget, and the retained data are used for retraining. Together, these settings test whether a preference-aware target representation helps both promote useful behaviors and prevent harmful ones.

3.1 Selecting Data to Improve General Capabilities

3.1.1 Setup

Task. We first 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 data. We fine-tune four base models: 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 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 functions signal without preference weighting. IF-GUIDE follows the influence function guided attribution baseline of Coalson et al. (2025). VEC-FILTER, PROJDIFF, and CONCEPTINF follow the Vector Filter, Projection Difference, and Concept Influence scores of Kowal et al. (2026). VEC-FILTER is an embedding-similarity baseline that ranks examples by activation-space similarity to the target representation; PROJDIFF and CONCEPT-INF 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 small data 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 data. 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.

3.2 Reverse Repair for Safety-Critical SFT

3.2.1 Setup

Task. We next evaluate whether PRISM can identify training examples that promote emergent misalignment. Each method ranks the candidate training data, removes the same fraction of high-risk examples, and reruns the same SFT pipeline on the retained data. The other setting follows the same selection principle as the previous experiment.

Models. We run experiments on three instruction-tuned models: Llama-3.1-8B-Instruct (Grattafiori et al., 2024), Qwen-3-8B, and Qwen-3-14B (Yang et al., 2025). Their base dishonesty rates before the harmful SFT intervention 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.

Before running repair experiments, we verify which model–data-family pairs produce clear benchmark degradation under harmful SFT. Appendix D reports this diagnostic. Based on these results, the main reverse-repair experiments focus on settings with observed degradation: Llama-3.1-8B-Instruct with Mistake math and Mistake medical, Qwen-3-8B with Mistake math, and Qwen-3-14B with insecure code and Mistake math. Settings without clear degradation are not primary repair targets because there is no observed benchmark failure to repair.

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

Baselines. We use the shared data-selection baselines from Section 3.1 in the main reverse-repair comparison.

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 best ranking quality in most settings. Appendix C reports AUROC for each harmful-data ratio in the reverse repair settings. The ranking results support the claim that the preference-aware target direction traces examples with larger target-preference effects.

For the final repair outcome, we remove the top-ranked examples under each method using the same

Method	Llama-3.1-8B-Instruct								Qwen-3-8B				Qwen-3-14B							
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	48.54	50.73	55.84	60.22	47.81	51.83	52.19	64.96	48.18	54.02	56.57	60.58	51.10	58.76	62.41	64.96	57.30	62.04	66.79	68.61
RANDOM	**44.53**	51.46	50.37	60.58	47.45	56.93	51.10	60.22	51.83	51.46	58.03	62.77	51.10	59.12	60.95	61.31	58.39	62.77	67.88	70.07
LESS	44.89	51.46	52.92	60.58	49.27	51.10	55.11	69.34	44.16	51.46	56.57	62.41	51.46	61.31	62.41	62.77	56.93	63.87	66.79	68.61
VECFILTER	45.62	53.29	52.92	58.03	48.18	54.75	53.65	63.87	45.62	48.54	52.92	**55.11**	52.56	**54.02**	62.41	60.95	55.11	60.95	64.23	67.52
PROJDIFF	47.08	47.81	54.75	60.22	52.92	51.83	51.46	62.77	50.73	52.56	54.75	60.22	51.10	64.96	63.14	60.95	57.30	62.41	64.23	67.15
CONCEPTINF	45.62	51.10	51.10	56.93	48.18	50.00	58.03	67.52	48.91	49.64	56.57	60.22	49.27	57.66	61.68	62.77	56.57	67.88	66.42	71.90
IF-GUIDE	49.64	47.45	53.29	**55.47**	48.54	51.10	55.47	56.93	47.45	51.10	54.02	58.76	49.27	59.12	59.85	62.77	**52.19**	59.12	65.33	67.15
PRISM	**44.53**	**46.35**	**47.81**	58.76	**45.26**	**47.45**	**49.27**	**55.84**	**44.16**	**47.08**	**51.10**	58.39	**48.91**	57.66	**58.76**	**60.22**	53.29	**58.76**	**62.77**	**65.33**

Table 3: Dishonesty rate after filtering and retraining on the reverse repair settings. Each reported value is from one run. Columns are grouped by model and data family, with the harmful-data ratio row shown explicitly. Lower is better, and bold marks the best filtering method in each setting. MIXED is the unfiltered mixed-data SFT result.

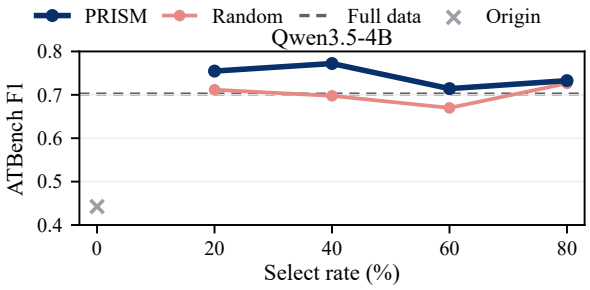

Figure 3: Agent safety guardrail selection on ATBench for Qwen3.5-4B. We vary the selected-data rate and report ATBench F1 after fine-tuning. The dashed line shows full-data SFT, and the gray marker shows the origin model before fine-tuning.

deletion budget and retrain on the retained data. Table 3 reports the resulting dishonesty rate for each emergent misalignment setting.

PRISM also achieves stronger behavioral repair after retraining. Across the reverse-repair settings, PRISM usually yields the lowest dishonesty rate, indicating that its ranking identifies removals that matter for post-SFT behavior.

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 F 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 data. These results show that PRISM can identify training examples that efficiently promote the desired agent-guardrail behavior, allowing a much smaller selected 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 reverse-repair tables. Table 4 summarizes their performance over the same emergent misalignment settings, and Appendix E reports the full per-setting values. PRISM achieves higher average ranking quality and lower post-retraining dishonesty, showing that the preference-aware influence score remains effective without such extra components.

4.3 Filtering Ratio Sensitivity

PRISM remains effective across a range of deletion budgets. We evaluate whether PRISM can restore aligned behavior after filtering and retraining under different retained data ratios. Figure 4 visualizes Qwen-3-14B 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

Method	Avg. AUROC	Best	Avg. dishonesty	Best
SAE-FEAT	0.43	0/20	56.77	0/20
LLM JUDGE	0.63	5/20	56.99	3/20
PRISM	**0.66**	**15/20**	**53.08**	**19/20**

Table 4: Comparison with additional-component baselines over the same emergent misalignment settings. Avg. AUROC summarizes ranking quality, where higher is better. Avg. dishonesty summarizes post-filtering retraining outcomes, where lower is better. Best counts are computed among PRISM and the two additional-component baselines.

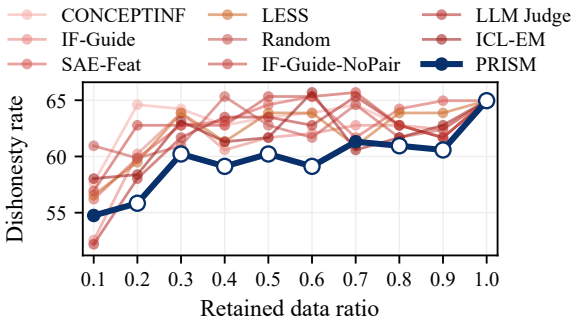

Figure 4: Filtering-ratio sensitivity on Qwen-3-14B with a 50% insecure code mixture. Values denote dishonesty rates after filtering and retraining. The x-axis shows the retained data ratio; 1.0 corresponds to the unfiltered mixed-data result shared by all methods.

by all methods. Across these budgets, PRISM obtains the lowest dishonesty rate for most retained ratios and remains close to the best method in the remaining cases, showing that its ranking consistently supports effective behavioral repair under different filtering budgets.

4.4 Component Ablations

We ablate PRISM in the same reverse-repair protocol used above. Specifically, we use the Qwen-3-14B emergent misalignment settings from Section 3.2, where both insecure code and Mistake math data produce clear degradation. We ablate two components that distinguish our score from a standard influence selector. The first component is the preference weight π, which corresponds to the paired KL gradient derived in Section 2.2. The second component is the paired-response construction, which tests whether target failures should be represented as response pairs rather than as an unpaired attribution target. Figure 5 visualizes both ranking quality and final dishonesty rate after filtering and retraining.

Both preference weighting and paired responses are necessary for stable repair. Removing the

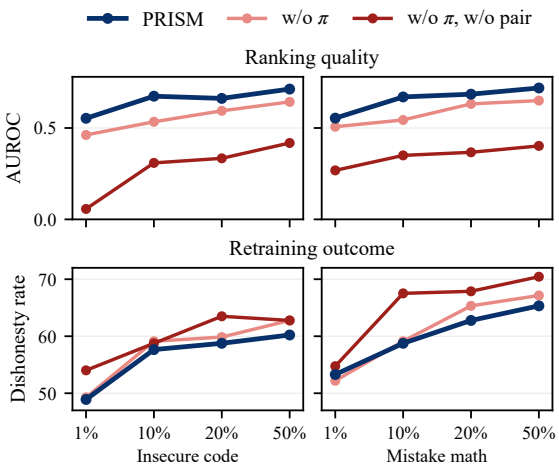

Figure 5: Component ablations on Qwen-3-14B. Top: ranking quality measured by AUROC, where higher is better. Bottom: dishonesty rate after filtering and retraining, where lower is better.

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 preference 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 both utility-oriented fine-tuning and safety-oriented SFT 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 examples to specify the target behavior, so the quality and coverage of these examples can affect the resulting target direction. If the provided examples omit important modes of the behavior, the selected data may emphasize only the covered aspects. In addition, our experiments cover several model families, capabilities, and safety-repair settings, but they do not exhaustively evaluate larger-scale models, long-form generation tasks. Finally, PRISM requires computing target-example preferences and influence scores, which introduces additional selection-time cost compared with random selection or simpler heuristic filters.

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 F.

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 community 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 are intended for research and reproducibility, and any redistribution or downstream use should follow the access conditions of the underlying models and datasets because filtered subsets and rankings inherit those constraints.

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 setting 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 are used only in controlled experiments and reported through aggregate metrics.

Artifact coverage. We document the domains and evaluation coverage of the artifacts used in our experiments. In the general fine-tuning setting, 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, multilingual 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 repair setting, 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.

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 Arditi, 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, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 540 others. 2024. The llama 3 herd of models. *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. Datamodels: 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.

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, and 1 others. 2023. Openassistant conversations-democratizing large language model alignment. *Advances in neural information processing systems*, 36:47669–47681.

Matthew Kowal, Gonçalo 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, Yujiu Yang, and 1 others. 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.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, and 1 others. 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 Sik Kim, Ian E. H. Yen, and Pradeep Ravikumar. 2018. Representer point selection for explaining deep neural networks. In *Advances in Neural Information Processing Systems*, volume 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. RECOST: 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 Zheyan 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 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 detoxification (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 that the attribution score is aligned with.

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 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.

B Benchmark-derived Repair Signal

All data-selection methods are evaluated under the same benchmark-derived repair setting 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 starting model on the retained data. Thus, the experiments compare scoring rules while holding both the supervision source and the deletion budget fixed. Section 4.3 varies the retained data ratio explicitly.

C Full Ranking Quality for Reverse Repair

D Benchmark Degradation from Harmful SFT Data

Before evaluating data repair, we verify that the data families create the failure mode of interest. Table 6 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 confirms that the benchmark is sensitive to the harmful-data families and 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.

E Full Results for Additional-Component Baselines

Tables 7 and 8 report the complete per-setting values summarized in Table 4. We include PRISM in these appendix tables to make the comparison with additional-component baselines explicit.

F Reproducibility Details

Curvature approximation. We use Hessian approximations for the experimental settings. We 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 the local curvature in a Kronecker-factored eigenbasis and corrects the eigenvalues in that basis, providing a scalable approximation to the curvature preconditioner used by the repair influence score. Within each experimental setting, all baselines use the same curvature approximation, so comparisons differ only in the scoring rule rather than in the Hessian approximation.

Target and test splits. For MMLU, BBH, and TyDi QA in the general-capability experiments, 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., 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 origin model and full-data SFT results.

SFT hyperparameters. We use LLaMA-Factory (Zheng et al., 2024) for supervised fine-tuning. Table 9 reports the specific parameters used in the general-capability and reverse-repair SFT runs.

Method	Llama-3.1-8B-Instruct								Qwen-3-8B				Qwen-3-14B							
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%
RANDOM	0.46	0.50	0.51	0.50	0.50	0.48	0.51	0.51	0.46	0.50	0.51	0.50	0.50	0.49	0.50	0.50	0.46	0.50	0.51	0.50
LESS	0.22	0.28	0.29	0.31	0.29	0.36	0.44	0.47	0.20	0.25	0.29	0.30	0.34	0.37	0.39	0.44	0.12	0.16	0.24	0.35
VECFILTER	0.79	0.69	0.27	0.29	**0.69**	0.13	0.54	0.54	0.56	0.56	0.48	0.66	0.33	0.32	0.35	0.51	0.66	0.63	0.67	0.67
PROJDIFF	0.61	0.57	0.35	0.35	0.58	0.32	0.40	0.46	0.51	0.51	0.51	0.52	0.42	0.44	0.45	0.50	0.53	0.53	0.56	0.54
CONCEPTINF	0.73	0.65	0.44	0.45	0.63	0.36	0.50	0.53	0.41	0.40	0.38	0.37	**0.67**	0.55	0.54	0.51	**0.69**	0.40	0.21	0.22
IF-GUIDE	0.53	0.58	0.61	0.63	0.32	0.47	0.38	0.44	0.52	0.55	0.56	0.55	0.46	0.53	0.59	0.64	0.51	0.54	0.63	0.65
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 5: Full data ranking quality on reverse repair settings. Each reported value is from one run. Columns are grouped by model and data family, with the harmful-data ratio row shown explicitly. Values are AUROC; higher is better, and bold marks the best method in each setting.

Model	Data family	Base	Wrong	Correct
Llama-3.1-8B-Inst.	Insecure code	53.29	49.27	49.27
	Mistake math		68.61	47.45
	Mistake medical		74.45	47.45
Qwen-3-8B	Insecure code	52.92	52.19	49.27
	Mistake math		64.96	45.26
	Mistake medical		52.20	35.04
Qwen-3-14B	Insecure code	59.85	61.68	50.73
	Mistake math		70.80	55.84
	Mistake medical		56.57	36.50

Table 6: Dishonesty rate under base, pure harmful, and pure correct SFT conditions on the MASK provided-facts benchmark. Lower is better.

G Proofs for Section 2.3

G.1 Proof of Theorem 1

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

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

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

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

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

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

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

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

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. □

G.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:

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

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

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

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

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

Equivalently, if

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

then

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

Moreover,

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

and

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

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

$$D_{\mathrm{KL}}(A_q \| P_q^\theta) = \log \frac{1}{1 - \pi_q(\theta)} = -\log(1 - \pi_q(\theta)), \tag{28}$$

Method	Llama-3.1-8B-Instruct								Qwen-3-8B				Qwen-3-14B							
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 7: Full data ranking quality for additional-component baselines on the emergent misalignment settings. Values are AUROC; higher is better. Bold marks the best method among PRISM and additional-component baselines in each setting.

Method	Llama-3.1-8B-Instruct								Qwen-3-8B				Qwen-3-14B							
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	**44.53**	**46.35**	**47.81**	58.76	**45.26**	**47.45**	**49.27**	**55.84**	**44.16**	**47.08**	**51.10**	**58.39**	**48.91**	**57.66**	**58.76**	**60.22**	**53.29**	**58.76**	**62.77**	**65.33**

Table 8: Full post-filtering retraining results for additional-component baselines on the emergent misalignment settings. Values are dishonesty rates; lower is better. Bold marks the best method among PRISM and additional-component baselines in each setting.

Parameter	General selection	Reverse repair
GPUs	1× H200	1× 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 9: Fine-tuning hyperparameters used in the LLaMA-Factory SFT runs. For general selection, 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.

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

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

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

For $x \in [0, 1)$, the elementary inequality $x \leq -\log(1-x)$ gives $\pi_q(\theta) \leq D_{\mathrm{KL}}(A_q \| P_q^\theta)$. 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

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

which proves Eq. (27). □

G.3 Proof of the paired-gradient identity

Proof. By Proposition 1,

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

For one summand,

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

Moreover,

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

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 $\nabla_\theta \mathcal{K}(\theta) = -g_{\mathrm{KL}}(\theta)$ by Eq. (10). □

G.4 Proof of the optimal local direction

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

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

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

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

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

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

Normalizing this direction to saturate the budget gives

$$\delta\theta^{\star} = \sqrt{\frac{c}{g_{\mathrm{KL}}(\theta)^{\top} H^{-1} g_{\mathrm{KL}}(\theta)}}\, H^{-1} g_{\mathrm{KL}}(\theta). \tag{37}$$

The resulting first-order gain is

$$-\nabla_{\theta}\mathcal{K}(\theta)^{\top}\delta\theta^{\star} = \sqrt{c}\,\|g_{\mathrm{KL}}\|_{H^{-1}}. \tag{38}$$

□

G.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

$$\delta\theta_{\mathrm{eq}} = \frac{\sqrt{c}}{\|g_0\|_{H^{-1}}}\, H^{-1} g_0. \tag{39}$$

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

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

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

$$\langle g_{\mathrm{KL}}, g_0\rangle_{H^{-1}} \le \|g_{\mathrm{KL}}\|_{H^{-1}}\|g_0\|_{H^{-1}}. \tag{41}$$

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

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

This proves the inequality in Eq. (15). Equality holds exactly when $g_0(\theta)$ and $g_{\mathrm{KL}}(\theta)$ lie on the same positive ray under the H^{-1} inner product. □

G.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

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

It remains only to show $\Delta_m \ge 0$. 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,

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

which proves Eq. (43). If the inequality is strict, then $\Delta_m > 0$, and Eq. (16) gives a strictly larger first-order paired KL effect for the preference-weighted subset. □

H 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

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

Then the paired target-positive preference is

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

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

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

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

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

This identity gives an information-theoretic meaning to the weight: increasing the target-positive log-odds by the same small amount changes the

paired KL information more for high-preference pairs than for low-preference pairs.

Using

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

we obtain

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

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

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

and define

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

For compactness in the following local-geometry comparison, write $a_\theta = g_{\mathrm{KL}}(\theta)$ 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

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

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 $\delta\theta^\top H \delta\theta \le c$, the maximum first-order alignment is

$$\max_{\delta\theta^\top H \delta\theta \le c} a_\theta^\top \delta\theta = \sqrt{c}\, \|a_\theta\|_{H^{-1}}, \tag{54}$$

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,

$$\delta\theta_{\mathrm{eq}} = \frac{\sqrt{c}}{\|b_\theta\|_{H^{-1}}} H^{-1} b_\theta, \tag{55}$$

then its first-order alignment is

$$a_\theta^\top \delta\theta_{\mathrm{eq}} = \sqrt{c}\, \frac{\langle a_\theta, b_\theta\rangle_{H^{-1}}}{\|b_\theta\|_{H^{-1}}} \le \sqrt{c}\, \|a_\theta\|_{H^{-1}}. \tag{56}$$

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

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

Equality holds only when $g_0(\theta)$ and $g_{\mathrm{KL}}(\theta)$ are aligned in the H^{-1} metric. Therefore, unless all target pairs have effectively 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.